Intraoperative Endoscopic Augmented Reality in Third Ventriculostomy

Zur Erlangung des akademischen Grades eines

Doktors der Ingenieurwissenschaften

von der Fakultät für Informatik
der Universität Fridericiana zu Karlsruhe (TH)

genehmigte

Dissertation

von

Matteo Ciucci
aus Monza

Tag der mündlichen Prüfung:	23.10.2009
Erster Gutachter:	Prof. Dr.-Ing. H. Wörn
Zweiter Gutachter:	Prof. Dr. med. Christian Rainer Wirtz

Kurzfassung

Endoskope wurden seit jeher als passive Inspektionssysteme verwendet. Sie werden sehr häufig in der minimalinvasiven Chirurgie (MIC) eingesetzt, um Aufnahmen vom Inneren zu erhalten, zu operieren und um Tumore zu entfernen. Es ist vorauszusehen, dass die nächste Generation an Endoskopen die Möglichkeit bieten wird, über die Technik der Erweiterten Realität (ER) präoperative 3D-Modelle dem Video des Endoskops zu überlagern.

ER ist eine bereits ausgereifte Technik, die jedoch im Moment in der Chirurgie noch nicht vollständig ausgenutzt wird. In Operationsmikroskopen wird nur eine 2D-Kontur des präoperativ aufgenommenen Tumors dargestellt und Endoskope mit ER-Unterstützung kommen noch gar nicht zum Einsatz. Neurochirurgen verlassen sich immer noch lieber auf ihre visuellen Eindrücke und die Navigation der Endoskope wird nur als Unterstützung angesehen.

Obwohl die ER den Anforderungen der Chirurgie sehr gut gerecht wird, ist es dennoch eine rein passive Technik: Präoperative Daten werden intraoperativen Bildern überlagert, um Regionen sichtbar zu machen, die ansonsten nur schwer zu erkennen wären, sprich krankes von gesundem Gewebe zu unterscheiden. Dies ist zum Beispiel bei Grad I-II Gliomen in der Neurochirurgie der Fall. Während der Operation wird keine quantitative Messung durchgeführt. Die eingeblendete Information stammt rein nur aus der präoperativ bekannten Information.

Fakt ist, dass sich die beobachteten Gebiete während der Operation verändern. Dies kann vielerlei Gründe haben: Änderung des intrakranialen Drucks, elastische Deformationen von Weichgeweben nach der Kraniotomie (in Richtung der Anziehungskraft) und natürlich die Entfernung von Gewebe. Als Konsequenz verschieben sich nun kritische Strukturen von der präoperativ geplanten Position zu neuen intraoperativen Positionen. Die Abweichung kann dabei bis zu einigen Zentimetern betragen. Diese Verschiebung macht die präoperativen Modelle sehr schnell obsolet.

Die Notwendigkeit von aktiven Endoskopen, die in der Lage sind im Dreidimensionalen den optischen Fluss und die 3D-Rekonstruktion zu vereinen liegt auf der Hand. Als mögliches Szenario kann eine Oberfläche intraoperativ gescannt und das Patientenmodell upgedated werden. Die möglichen Anwendungen reichen dabei von Modellerweiterung, intraoperativer Registrierung bis hin zur Korrektur von Modell- und Tumorverschiebungen.

Das NEAR-Projekt (Neuroendoskopie towards Augmented Reality) bietet einen Ansatz um aktive Endoskope zu entwickeln, das heißt ein Endoskop ausgestattet mit Navigation, Erweiterter Realität und Triangulationsmodulen mit dem Ziel, intraoperativ 3D-Punktwolken aufzunehmen und diese mit den präoperativen Daten zu verknüpfen.

Abstract

Endoscopy has grown since its early birth to become the main visualization technique in minimally invasive procedures. Minimally invasive surgery, unlike open surgery, involves small incisions and the use of endoscopes to indirectly inspect the surgical field. This procedure offers reduced trauma, smaller scars and faster recovery time and usually results in shorter hospital stays.

However, smaller incisions also mean a tighter working volume, a limited freedom of movements and the loss of the surgeon's depth perception upon the observed area. Because of these drawbacks, significant skills are required from surgeons in performing minimally invasive procedures, the problem being only partially relieved by the introduction of the surgical navigation.

Even if endoscopy it's today the preferred way to perform keyhole surgery, it remaines in the surgical practice a passive technology, with endoscopes used only as inspecting devices. If surgical navigation allows endoscopes to be tracked, its main goal doesn't go beyond a mere locatization of the instruments in the patient frame. Endoscopic camera poses and their relative views, which might be combined to recover the real-time geometrical information of the observed patient, remain in the current surgical practice two separate sources of information.

Medical images acquired through surgical microscopes are usually augmented with the position of critical structures like veins, vessels or nerves or with the profile of the tumoured area to be removed. Low-grade gliomas, brain tumours in their early stage, cannot in neurosurgery be distinguished from healthy tissues. Surgeons use diagnostic data for their identification and augmented reality for their direct visualization onto the observed image.

Diagnostic information unfortunately describes the preoperative geometry of the patient's anatomy only. In neurosurgery, as a result of a change in intracranial pressure or because of any elastic relaxion happening after the craniotomy, the intra-operative geometry used as a local reference changes. Preoperative models, useful to locate tumours, become intraoperatively obsolete. More generally, any cut or tissue removal which alters the preoperative patient geometry raises issues about the drop of significance of preoperative virtual models against the timeline of the operation. The need for a model update becomes then obvious.

The NEAR project, *Neuroendoscopy towards Augmented Reality*, is an attempt to

build an active endoscope, i.e., an endoscopic device equipped with navigation, augmented reality, and triangulation modules, with the goal of extracting intraoperatively 3D point clouds to allow intraoperative patient registration, further referencing and surface model reconstruction.

Preface

Some of the work in this thesis has already been published partially as a short paper or as a full contribution in the proceedings of the following conferences: CURAC 2007, Deutsche Gesellschaft für Computer-und Roboter-Assistierte Chirurgie; workshop MIMOS 2007, Movimento Italiano Modellazione e Simulazione; CARS 2008, Computer Assisted Radiology and Surgery Conference; international IEEE conference VECIMS 2009, Virtual Environments, Human-Computer Interfaces and Measurement Systems; HCII 2009, Human Computer Interface.

The work has been mainly done at the Institute for Process Control and Robotics of Karlsruhe, Germany, with the cooperation and the medical supervision of the Neurosurgical Departments of the Universities of Heidelberg, Ulm and Günzburg, Germany.

The COMPU-SURGE/NEAR project is part of the MARIE CURIE ACTIONS, EU-funded Host fellowships for Early Stage Research Training under the Sixth Framework Programme.

Acknowledgements

This work wouldn't have been possible without the help of many people. First of all I would like to thank my supervisor Prof. H. Wörn for his guidance and insight during all my time spent at the IPR, Institute for Process Control and Robotics. I also thank my co-supervisor Prof. C. R. Wirtz for his patience, valuable input and constant medical supervision during my frequent visits in the operation room at the University Clinics of Heidelberg, Ulm and Günzburg.

I also would like to thank my group leader Dr. Jörg Raczkowsky, who showed me how to effectively lead a small group of young researchers and gave me the possibility of pursuing my own research line, supporting me with his strong competence and valuable leadership.

Big thanks to the whole medical group, MEGI, for the great atmosphere shared during my stay: Jessica Burgner, Alessandro De Mauro, Lüder Alexander Kahrs, Gavin Kane, Markus Mehrwald, Matthias Riechmann, Daniel Stein, Holger Mönnich. Oliver Weede.

A special thank to the members of the COMPU-SURGE project: Vitor Vieira, Gavin Kane and Horia Ionescu: discussions with them have been always stimulating and fruitful. Another thank to the medical project supervisors, Dr. R. Marmulla and Dr. G. Eggers for their valuable inputs during our regular project meetings; thanks to Dr. R. Bösecke for his impeccable financial management and valuable suggestions.

I'm grateful for the support the Klinik und Poliklinik für Mund-, Kiefer- und Gesichtschirurgie of the University of Heidelberg and the Neurosurgical Departemnts of the clinics of Ulm and Günzburg gave me. I thank in particular the Privat-Dozent Dr. Halatsch for answering to all my medical questions with great competence and Mr. Heckele of the R. Wolf Company for the interest shown in my project.

Many thanks also to the whole IPR group who warmly welcomed me since the very beginning, in particular to the phd students: Michael Mende, Stefan Zimmermann, Simon Notheis, Alexander Steig, Davide Lanieri; coffee-time together was always a good time.

A warm hug to all my friends from all over the world who shared the same life experience in Germany: Marta, Fernando, Hernan, Ivana, Fréderic, Ana, Bruno,

Ernesto, Mattia, Carlo, Belen, Michel.

Last but not least, I would like to express my gratitude to my close friends Vittorio, Andrea, Paolo, Giorgio and to my family for their constant support during all these years abroad. A very special thank goes to my wife Simona, who assisted and helped me during these years with her patience, devotion and love.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Endoscopes and tracking systems are becoming ubiquitous in any modern operating room. Image-guided surgery (IGS), the general term used to describe any surgical procedure where the surgeon uses indirect visualization to operate, makes a wide use of internal video cameras, fiber optic guides and flexible or rigid endoscopes. Originally developed for the treatment of brain tumours, IGS is today mostly performed as minimally invasive surgery (MIS) procedure.

A MIS procedure, compared to the traditional open surgery, typically involves the use of laparoscopic devices and remote-control manipulated instruments. The indirect observation of the surgical field with an endoscope is carried out through the skin, a body cavity or an anatomical opening. MIS procedures offer to the patient a lot of benefits, such as less scars and trauma, shorter hospital stays, reduced recovery time and outpatient treatment but bring severe difficulties to surgeons and hospitals: MIS operations may indeed last longer and be more complex than traditional operations. To minimize the number of errors and new possible complications [1] deriving from their applications, surgeons performing MIS procedures must be both well-trained and experienced.

During the MIS procedures, the indirect vision through the endoscope and the indirect manipulation of the tissues are the main causes of the surgeon's perception problems. These can be classified into disturbed hand-eye coordination problems, reduced depth perception problems, and reduced haptics problems. Nonetheless, MIS procedures, like the removal of a gallbladder or an appendix, are known to be as safe as their correspondent open surgery operations. To allow the tracking of the surgical instruments into the surgical scene, the surgical navigation attaches some artificial markers like retro-reflecting spheres or special print-outs to the tools. The support of the surgical navigation allows to represent the surgical tools in the preoperative virtual models of the patient [2], providing a reference during the operation which is perceived from the surgeons as a valid technological support during most MIS operations.

The current state of the art in medical technology research can be described by highlighting its main investigation lines: robot-assisted surgery and tele-manipulation offer to improve the surgeon's accuracy in fine positioning and maneuvring tools by removing the high-frequency hand tremors [3]; haptics research investigates the tactile and kinestetic perception to provide the surgeon with a reliable haptic feedback during remote operations [4]; the current medical visualization research tries to to avoid the hand-eye coordination and depth perception problems [5] resulting from an indirect view of the operation field by proposing new visualization and rendering algorithms.

With the growth of computer vision [6, 7], the research in medical visualization focused his attention on the whole spectrum of opportunities offered by the introduction of real-time image processing methods on endoscope devices. One of the most promising visualization techniques developed at the very beginning for industrial applications [8] is Augmented Reality (AR). Medical AR renders virtual organs on intraoperative endoscopic videos and draws currently the interest of the surgical community with more and more systems tested on surgical phantoms, cadavers and real patients [9, 10, 11, 12, 13, 14, 15].

The main advantage offered by the medical AR is the direct and intuitive visual introduction in the intraoperative context of the preoperative information which integrates the planning information in the intraoperative context; its main limitation is its lack of data update during the operation. Since the preoperative information becomes, as the surgeon operates, rapidly obsolete, its usefulness along the operation timeline decreases.

Research in MIS devices is also pushed forward by computer vision techniques. Active research fields in endoscopy propose to support surgeons with a broad spectrum of computer vision features: D. Dey describes in [16, 17, 18] an automated fusion of freehand endoscopic brain images to create stereoscopic panoramas; U. Bockholt uses augmented reality in [11] to import the surgical planning during image-guided surgery; Konen introduces in [19] virtual intraoperative views to inspect otherwise inaccessible regions; Scholz shows in [20] how to build live databases of endoscopic images by saving couples of endoscope positions and views to offer stored views in case of profuse bleeding; Bartz, Neubauer and Fischer show in [21, 22, 23] various possible applications of virtual endoscopy as a diagnostic and intraoperative tool; Fossati shows in [24] marker-less tracking and navigation, avoiding the implantation of fiducials into the patient.

At the other extreme of the medical visualization, scene recontruction techniques are also becoming more and more prominent. These methods, which reconstruct from multiple views the geometry of any observed object, are complementary respect to pure visualization techniques. Instead of introducing the preoperative information into the observed scene like AR does, scene reconstruction techniques aim at extracting a quantitative information by measuring the observed scene from multiple views. Quantitative endoscopy is therefore a technique based on scene reconstruction methods [25, 26, 27] that allows to extract and use the 3D geometrical surface

information of an inspected scene. This technique, which adapt 3D reconstruction methods to MIS contraints, opens to the medical technology new exciting possibilities: intraoperative registration, update and creation of virtual models would become then available techniques in a near future.

This work is an attempt to describe both the advantages and the possibilities offered by a single endoscopic tool which integrates together AR and scene reconstruction techniques.

1.1 The main scenario

The following paragraph introduces the concepts of neuroendoscopy, augmented reality, virtual and quantitative endoscopy required to understand the problem statement and the goal of the present work. The main ideas are illustrated through the more relevant keywords and their corresponding explanation.

Neuroendoscopy is a medical procedure whereby brain surgery can be undertaken with minimum disturbance to the patient. Today the practice of this very special branch of endoscopy is deeply entangled with the devices used during the operation and able to provide augmented reality and surgical navigation support. The neurosurgeon operates with either a surgical microscope or an endoscope: in the first modality, the brain tissue is examined via a couple of oculars on a computer-enhanced image; in the second modality, a neuroendoscopy intervention is performed with the aid of a navigated instrument showing on an external monitor the relative endoscope and tumor positions. It requires an expert surgeon to recognize the tumour boundary on an camera image. The visual detection of tumours like low-grade gliomas, for example, is impossible even to experienced surgeons which can locate them only by using radiological scans.

Augmented reality applied to endoscopy is a technique thought to enhance the surgeon's perception during a real endoscopy operation. Augmented reality is used today to highlight the relevant tumoured structures, overimposing bright shapes on the endoscope image camera whenever a critical anatomical structure enters in the field of view of the surgeon. Augmented reality requires the camera as well as the patient to be tracked by a standard navigation system and aims to enhance and extend the image contents with preoperative diagnostic or positional data. No effort is done to automatically extract any metrical or diagnostical information from the images, the principal aim of the technique being the direct visualization of the preoperative data on the intraoperative images.

Virtual endoscopy is a non-invasive technique meant to explore hollow organs and anatomical cavities using 3D medical imaging and computer graphics with no need for any real operation performed on a patient. Using a reconstruction of the

patient's 3D model, a surgeon can perform an analysis of the preoperative patient's anatomy, for example by carefully looking for polyps in the patient virtual model. The surgeon may also intraoperatively benefit of the 3D reconstructed environment for navigation purposes and referencing. Virtual endoscopy is not yet a well established diagnostic procedure. The trade-off between the dose of radiation absorbed by a patient and the benefit obtained by performing a non-invasive diagnostic procedure must still be evaluated. Virtual endoscopy remains a good candidate technique to improve, or even in some cases to replace, real diagnostic endoscopy.

Quantitative endoscopy is a research technique though to extract metric information from navigated endoscopic images. It brings into the medical field the efforts of 3D scene reconstruction techniques. As opposite to augmented reality, quantitative endoscopy aims at extracting geometric information from images, like the geometry of an area currently object of surgery or the depth of inspected structures. This new and up-to-date information source may be used to complete, extend, or replace the original preoperative and therefore obsolete augmented reality information or to create geometrical surface models on the base of the surgeon's needs.

1.2 Problem statement

Augmented reality has also been very influential in the development of new video endoscopic devices. The opportunity of extending the image information content with preoperative diagnostic and positional data has been welcomed enthusiastically since its very first birth. However, endoscopic images are augmented with virtual details even when the correspondence of preoperative models with the real scene is questioned by the elastic deformations occurring during an operation on the patient's anatomy as a consequence of the surgical intervention or of the relaxation of the soft tissues. Virtual models immersed in a real scene can actually provide reliable information only when two obvious conditions are satisfied: for the whole life of the operation, both their global registration with the real scene must remain above a certain level of accuracy and the preoperative 3D models used to augment the images must remain up-to-date. While the first condition usually happens in a lot of augmented reality systems, the second one is in surgical practice not always true. In neurosurgery, for instance, the brain-shift effect and the change in intracranial pressure due to cerebro-spinal fluid (CSF) losses do modify the patient's initial state. Patient's preoperative models become then rapidly obsolete and useless. In this particular case, a second intraoperative MRI scan performed after the craniotomy corrects the brain shift effect and is regarded as the main available solution to this problem. Enlarging the perspective, any tissue removal due to the standard surgical practice which modifies the actual patient state raises issues about the drop of significance of the virtual models against the timeline of the operation. Quantitative endoscopy can be seen as a possible bottom-up solution to the problem of

keeping the models up-to-date.

We will use in the paper the following terminology: we define **passive endoscopes** as devices able to inspect or to enhance the surgeon perception without being able to extract any quantitative information from the observed scene. Pure or even augmented reality equipped endoscopes fall therefore within this category because they introduce into the observed scene preoperative information only. We define on the other hand **active endoscopes** as devices able to intraoperatively 3D reconstruct the observed scene. This work describes another attempt to build an active endoscope with the goal of extracting intraoperatively 3D point clouds allowing intraoperative patient registration, further referencing, surface reconstruction and model correction.

1.3 Motivations of the work

The NEAR Project, *Neuroendoscopy towards Augmented Reality*, presents an active endoscope implementation realized using standard operating room equipment. The application is tailored to augmented reality and feature triangulation and assumes third ventriculostomy as its surgical scenario. In the following section the motivations to develop active endoscopes are listed and discussed.

Navigated endoscopes are standard tools

Any modern operation room (OR) is already equipped with optical or radio tracking systems. Tracked calibrated cameras are already used to perform augmented reality but not to perform any quantitative measurement. Cameras are defined calibrated if the correspondence between a 2D image pixel and its corresponding direction vector is, for the particular camera setup, known and can be described using a small set of parameters. Tracked calibrated cameras are therefore standard tools in any operating room.

The positional image information is wasted

Any navigated performed endoscopy is video recorded for safety purposes. On the other hand, any tracking information concerning the navigated endoscope position is only used intraoperatively to locate the tool at the time of the operation and subsequently discarded. The combination of both informations, which enables to describe both the camera pose and the inspected scene, even if useful to support the surgeon, is unfortunately available in its components at two different times and therefore not used.

The optical resolution overcomes the CT/MRI data resolution

A main motivation for building active endoscopes comes from a direct comparison of the endoscope optical spatial resolution with respect to the spatial resolution of a patient scan. A standard CT/MRI scan resolution is 1-2 pixel/mm; the optical resolution of an endoscope is, on the other hand, 10-40 pixel/mm. Higher resolution ratios can be achieved depending on the choice of the image detector and on the magnifying power of the endoscope lens. The average optical resolution of an endoscope is then already higher than the patient scan resolution. Even if not for diagnostic purposes, endoscopes may be used to intraoperatively acquire details of the currently inspected patient geometry which cannot be resolved during a CT/MRI scan.

The optical resolution overcomes the navigation resolution

The setup of the operation room is composed by tracking systems set at a far distance of 2-3 meters and by navigated endoscopes close to the patient. As explained in the previous paragraph, an endoscope is basically acting as a big magnifying glass close to the inspected object which has on this object a better spatial resolution than the average CT/MRI scan. Since navigation systems must have an average accuracy which is comparable or better than the CT/MRI data resolution, the optical endoscope resolution allows in principle a more accurate marker-less local navigation on the inspected area.

Local procedures require bounded information

An advantage of endoscopic surgical procedures versus open surgery is that most of them are minimally invasive local procedures. Surgeons usually look at what they are inspecting, making endoscopy an intrinsically local technique which gives information concerning the surgical region of interest only. A patient scan on the other hand contains a lot of global diagnosis and preoperative information whose only a small amount is used in the augmented reality applications. As an example, in the surgical microscopes the tumour representation uses bidimensional contours, the overlay of 3D virtual objects being still an active research field under evaluation. Most of the collected quantitative CT/MRI information is after the diagnosis discarded and the most intraoperatively relevant data concerns the region of endoscope navigation. It becomes then advisable to try to extract quantitative local information during navigation.

Inverting the 3D model generation workflow

Considered from a broader point of view, active endoscopes allow also to recast the classical surgical data workflow scheme. The classical flow of surgical data begins

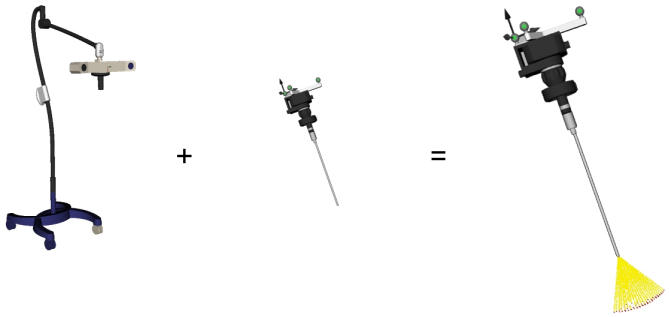


Figure 1: Graphical equation representing active endoscopes as the combination of tracking systems and calibrated cameras.

with CT/MRI scans, continues with the generation by segmentation of 3D virtual models and ends, after patient registration, using these models for the surgical navigation or the augmentation of the observed scene. The use of this data structure has been driven by two main reasons: the scanned data is used before to perform diagnoses and then reused to generate the patient virtual models for navigation and augmented reality. Active endoscopes would reverse the standard data workflow: real images with their high optical resolution may be used to extract local quantitative surface models of inspected areas. These would allow to perform intraoperative registration of preoperative models or intraoperative and up-to-date surface model generation.

1.4 Goal of the work

The NEAR project aims to design, develop, calibrate, test and enhance a fully augmented reality endoscopic system. The application must be able to integrate the augmented reality with the pure virtual environment, which can be used as a support but could also easily be excluded. The application has been designed, developed and evaluated in strong cooperation with the Neurosurgical Department of the University clinics of Heidelberg and Ulm, Germany.

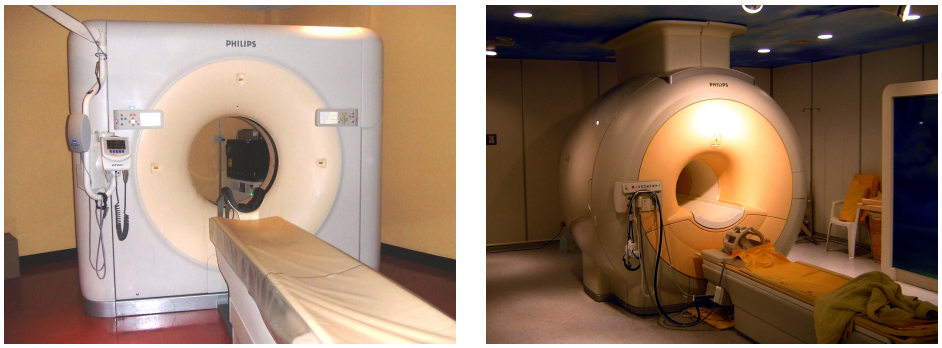
Chapter 2

Principles of medical imaging and visualization

The following chapter introduces the reader to the field of computer assisted surgery (CAS) and to the necessary concepts required to understand this work. The chapter describes the main modalities of medical imaging used throughout this work, introduces the topic of medical visualization and virtual model generation and ends describing the registration procedure of virtual models against the current patient's anatomy.

2.1 Medical imaging

Medical imaging is the collection of techniques and processes used to create images of the human body for clinical purposes. Medical imaging is primarily designed to produce positional data maps which are visualized as images. Images are acquired through different modalities: computer tomography, magnetic resonance imaging,



(a) (b)

Figure 2: A multislice CT scanner: Philips Brilliance 64-channel thin-slice (a); modern 3T clinical MRI scanner (b).

ultrasound, nuclear medical imaging are among the most popular. Since this work focuses on image visualisation more than on their production, more emphasis will be given to the main high resolution imaging techniques: radiographs, CT and MRI. Medical ultrasonografy (US) and nuclear imaging are mentioned as ones of the main 3D imaging techniques because of their relevance in the medical field, even if not used in this work.

2.1.1 Radiography

Two kind of radiographic images are used in medical imaging: projection radiography and fluoroscopy. Both utilize a wide beam of x rays for image acquisition.

Radiographs

Radiographs or x-rays are used to determine the type and the extent of a fracture as well as to detect pathological changes in the lungs. Medical x-rays are emitted through x-rays tubes, through x-ray fluorescence as spectral lines, or through bremsstrahlung as a continuous sprectrum, with a beam energy in the range of 5-150 keV. Only the hard part of the x-rays spectrum is in medical diagnostic applications kept, since low energy x-rays are totally absorbed by the body increasing the dose of radiation delivered to the patient. A single bone radiography delivers an average effective dose of 0.1 mSv.

Fluoroscopy

Fluoroscopy is an X-ray based technique to obtain real-time moving images of internal structures of the body. It employs a constant x-ray source at a lower dose rate as an input. The transmitted signal was previously collected on a fluorescent screen, while today hits a caesium iodide phosphor which is deposited directly on the photocathode of an x-ray intensifier whose output is sent to a CCD video camera. The Lubberts effect, which refers to the non-uniform response of an imaging system to x-rays that are absorbed at different depths within the input phosphor, limits the image resolution of this technique. Contrast media such as barium, iodine, and air are used to better visualize internal organs. Typical skin dose rates are 20-50 mSv/min.

2.1.2 CT

Computer tomography is a digital geometry process used to generate a three-dimensional image of the inside of an object from a large series of two-dimensional X-ray images taken around a single axis of rotation. X-ray slice data is generated using an X-ray source that rotates around the object; X-ray sensors are positioned

on the opposite side of the circle from the X-ray source. The data stream representing the varying radiographic intensity sensed at the detectors on the opposite side of the circle during each sweep is then computer processed to calculate cross-sectional estimations of the radiographic density, expressed in Hounsfield units. Detectors matrices have usually 256x256 or 512x512 square pixels, with pixel sides of 1,2,4,8 mm, according to the required resolution. A single CT scan, depending on the observed volume, delivers an effective average dose ranging from 1 to 10 mSv. Reducing the radiation dose during CT examinations without compromising the image quality is today the main radiological issue.

2.1.3 MRI

Magnetic resonance imaging, also known as nuclear magnetic resonance imaging, is a medical imaging technique used to visualize the internal structure and functions of the human body which uses a powerful magnetic field (0.1-3 T) to align the nuclear magnetization of hydrogen atoms in the body water. A radio frequency (RF) field is turned on causing hydrogen nuclei to absorb some of its energy and to alter the alignment of their magnetization along its longitudinal and transverse component respect to the external field. When the RF is turned off, protons realign with the external field and release their excess energy as a radiation detectable by the scanner. The recovery of the magnetization occurs exponentially with a time constant T and is called longitudinal or T_1 and transverse or T_2 relaxation. Small inhomogeneities make the observed transverse relaxation time T_2^* shorter than the theoretical T_2. In soft tissues, T_1 is about 1s and T_2^* about 10ms. To spatially locate single emitters, a magnetic gradient field (1-100 mT/m) is applied across the body so that different spatial locations become associated with different precession frequencies. MRI makes no use of any ionizing radiation.

2.1.4 Ultrasonography

Ultrasonography uses a piezoelectric transducer to produce an acoustic wave (1-18 MHz), measures the time it takes for the echo to travel back to the probe and uses it to calculate the depth of the tissue interface causing the echo. The main drawback of the technique is that even if the speed of the sound differs in different materials and is dependent on its acoustical impedance, the sonographic instrument assumes the acoustic velocity as constant at 1540 m/s. As a consequence, in a real body with non-uniform tissues the beam becomes de-focused and image resolution is reduced.

2.1.5 Nuclear imaging

In nuclear imaging energetic photons or gamma rays emitted from radioactive nuclei are used both for diagnosis by enhancing and viewing various pathologies, and for

treatment by irradiating tumours with high radiation doses.

Gamma cameras

Gamma cameras are devices used to image gamma radiation emitting radioisotopes introduced into the body. A short lived isotope is administered to the patient to be absorbed by biologically active regions of the body often associated with diseases, such as tumors or fracture points in bones.

PET

Positron emission tomography is an imaging technique which produces a three-dimensional image or picture of functional processes in the body. The system detects pairs of gamma rays emitted indirectly by a positron-emitting radionuclide or tracers introduced on a biologically active molecule in the body. The technique used to reconstruct the image is similar to CT/MRI.

2.1.6 Research techniques

Optical coherence tomography (OCT), synchrotron medical imaging, photoacustic imaging, elastography, being among the techniques currently object of active medical research, are only worth to be listed here.

2.2 Image reconstruction

In medical imaging devices, objects are usually scanned along different projections or sections (fig. 3). Even if the tomographic reconstruction problem deals with obtaining a 3D scan from its 2D sections, in the following we will reduce it to the problem of reconstructing a 2D image from their 1D projections as the mathematic tools involved are easily generalizable [28]. The Radon transform of a distribution function $f(x, y)$ is:

$$p(\xi, \theta) = \int f(x, y)\delta(x\cos(\phi) + y\sin(\phi) - \xi)dxdy \tag{1}$$

The function $p(\xi, \theta)$ is often referred to as a sinogram (fig. 2.2) because the Radon transform of an off-center point source is a sinusoid. The task of the tomographic reconstruction is to find $f(x, y)$ given the knowledge of $p(\xi, \theta)$. Mathematically, the backprojection operation is defined as:

$$f_{BP}(x, y) = \int_0^{\pi} p(x\cos(\phi) + y\sin(\phi), \phi)d\phi \tag{2}$$

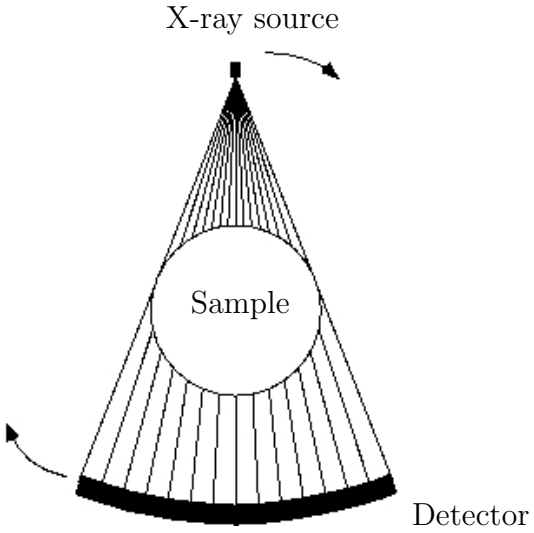


Figure 3: Third-generation rotate-only fan beam CT geometry

Geometrically, the backprojection operation simply propagates the measured sinogram back into the image space along the projection paths. As for a point source at the origin $\delta(x, y)$ the intensity of the backprojection image rolls off slowly as $1/r$, the relationship between $f(x, y)$ and $f_{BP}(x, y)$ is:

$$f_{BP}(x, y) = \frac{1}{r} \star f(x, y) \tag{3}$$

where the symbol $\star$ denotes the convolution operator. The full 2D image reconstruction from projections is based now on the projection slice theorem, which states that the 1D Fourier Transform (FT) of a projection at angle ϕ is a line on the 2D Fourier transform of the image at the same angle. The 2D original signal is then easily recovered: all the 1D FT projections are aligned along their corresponding lines and then interpolated. A 2D FT backprojection gives then the original 2D slice signal. To avoid interpolation in the frequency domain, a filtered or a convolved backprojection is used.

Back-projection

Since there are many tomographic reconstruction techniques, we will limit our description to only the two more relevant: the Direct Fourier (DF) reconstruction and the Filtered Back-projection (FB) reconstruction. In the DF reconstruction, once $F(\omega_x, \omega_y)$ is obtained from $p(\xi, \phi)$ using the PST, $f(x, y)$ can be obtained by applying inverse FT to $F(\omega_x, \omega_y)$. An artifact-prone interpolation in the Fourier space is however required: to utilize the fast Fourier transform algorithm, values of $F(\omega_x, \omega_y)$ should be available at a rectangular grid, while the values generated from the CST are available at a polar grid. The FB algorithm avoids this interpolation

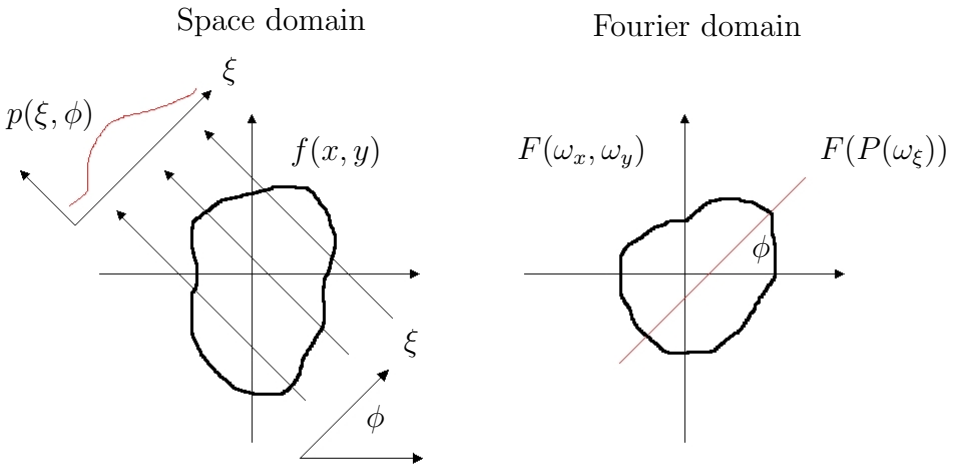


Figure 4: Projection slice theorem: the projection under an angle ϕ equals the slice under ϕ in Fourier domain

in the Fourier space; the filtered profile $p'(\xi; \phi)$ convoluted with the ramp filter $b(\xi)$ is used instead of the original profile $p(\xi, \phi)$:

$$p'(\xi; \phi) = p(\xi; \phi) \star b(\xi) = \int_{-\inf}^{\inf} |\nu| \, P(\nu, \phi) \exp(-2i\pi\nu\xi) d\nu \tag{4}$$

The importance in the choice of the back-projection filter is examined for example in [29]; the filtered back-projection profile algorithm is by far the most widely used algorithm in clinics.

2.2.1 Data reconstruction

Once acquired through CT or MRI, the process of 3D data reconstruction from a set of 2D slices is relatively easy. According to the multiplanar reconstruction method (fig. 2.2.1), the slices are stacked together according to their z-resolution along a regular grid. Each 3D position defines a single volume pixel or voxel whose brightness is defined as its corresponding scalar value. Voxels are rendered as 8-bit digital signals using gray value units, integer values ranging from 0 (black) to 255 (white). Reconstruction is also possible along oblique planes using data interpolation.

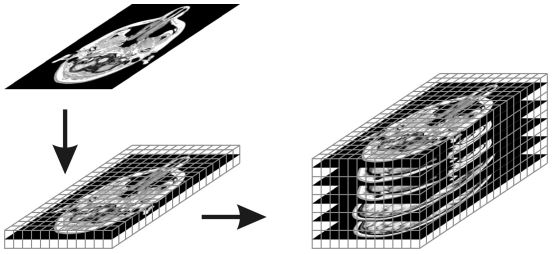


Figure 5: Multiplanar reconstruction method. Source: Dr. Hoppe doctoral thesis

2.2.2 Data visualization

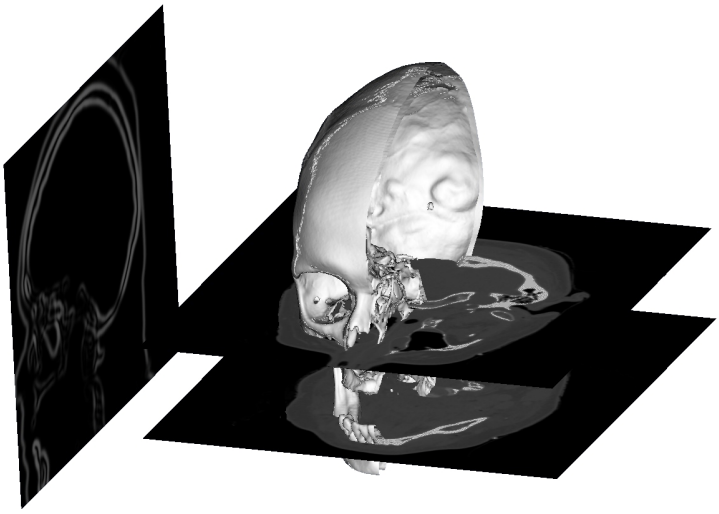
Data is visualized using different rendering techniques: in surface rendering, polygonal meshes of isosurfaces are extracted from a three-dimensional scalar field using the Marching Cubes algorithm. The algorithm proceeds through the scalar field, takes eight neighbour locations at a time which form an imaginary cube, and determines the polygons needed to represent the part of the isosurface that passes through this cube. The individual polygons are then fused into the desired surface [30]. In volume rendering, every single value is treated as a single block of data, converted to opacity and color (RGBA) and projected on the correspondent pixel on the frame buffer. This method maps elements directly into screen space and avoids using geometric primitives as an intermediate representation. Volume rendering is especially useful for representing amorphous features such as clouds, fluids, and gases but has the disadvantage that the entire dataset must be traversed for each rendered image.

2.3 Patient registration

The correspondence between the 3D virtual patient dataset and its corresponding real patient position and orientation or frame, is of primary importance in computer guided surgery. Patient registration is the process of aligning the model dataset with the current patient's frame in the operating room. The best rigid body transformation $[\mathbf{R}, \mathbf{t}]$ aligning or registering the two data sets is found by minimizing the mean squared error between the two sets of points. The problem of finding the best rigid transformation which maps one frame into another is common to image registration, patient registration and rigid body tracking and known as Orthogonal Procrustes Problem in statistics or Absolute Orientation Problem in photogrammetry. Given two sets of N points X_i and Y_i, the absolute orientation problem finds the best rigid body transformation $\mathbf{T} = [\mathbf{R}, \mathbf{t}]$ which minimize in the least squares sense the residuals:

$$\min_{\mathbf{T}} \|\mathbf{TX} - \mathbf{Y}\| \ , \quad \mathbf{R}\mathbf{R}^{\mathbf{t}} = \mathbf{1}$$

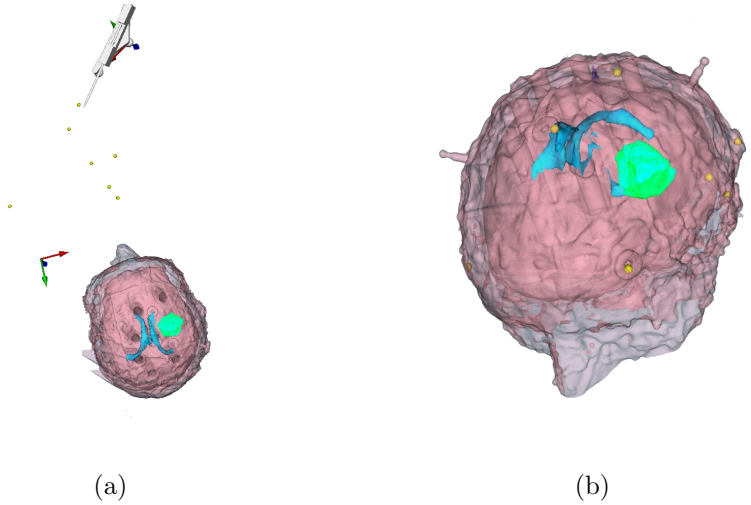
The very first closed solution to the problem was found by Schönemann in 1966 [31]; the gold-standard method was successively defined by other independently found



(a)

(b)

Figure 6: Visualization of medical CT/MRI data: surface rendering (a); volume rendering (b). Source: Kitware Inc.



(a) (b)

Figure 7: Point-based patient registration: the point cloud acquisition with a pointer in the vistual scene (a); the registration performed (b)

solutions using eigenvalue-eigenvectors decomposition of a matrix [32, 33], singular value decomposition [34] or unit quaternions [35]. For any method, the translation vector $\mathbf{t}$ is equal to the vector aligning the centroids of the two data sets. For the rotational part, the singular value decomposition solution is particularly easy: as any matrix $\mathbf{A}$ can be decomposed as a product of a diagonal matrix $\mathbf{\Lambda}$ and two orthogonal matrices $\mathbf{U}, \mathbf{V}$:

$$\mathbf{A} = \mathbf{U}^t\mathbf{\Lambda}\mathbf{V} \ , \quad \mathbf{U}\mathbf{U}^\mathbf{t} = \mathbf{V}\mathbf{V}^\mathbf{t} = \mathbf{1}$$

The solution to the absolute orientation problem is the rotation matrix $\mathbf{R}$ given by the projection on SO_3, the group of proper 3D rotations, of the original matrix $\mathbf{A}$ or:

$$\mathbf{R} = \mathbf{U}^t\mathbf{V}$$

To acquire the data sets, two main methods are known: point-based methods use sparse point clouds acquired semi-automatically or manually; surface-based methods use instead laser scanners or projectors to automatically scan dense point clouds; a comparison between both methods can be found in [36, 37, 38].

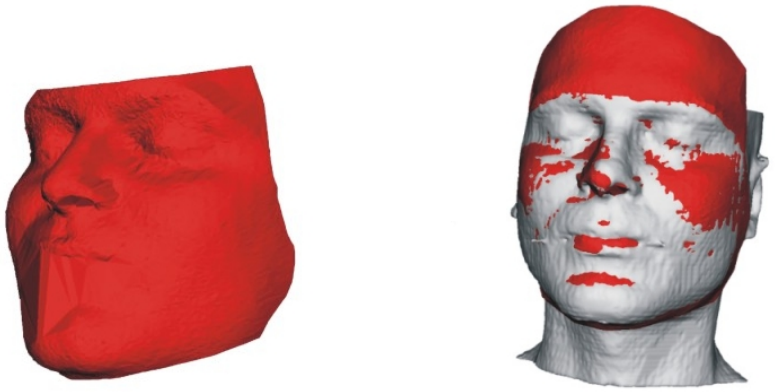
2.3.1 Stereotactic frame

A stereotactic frame is a mechanical device which allows the accurate positioning of instruments such as probes, electrodes and cannulas in three-dimensional space [36]. It realizes a manual registration of the preoperative data in the patient frame offering at the same time a guidance to an accurate navigation. It consists of three components: a planning system which includes an atlas, some image matching tools, and a coordinates calculator; the stereotactic device or stereotactic frame; a localization and placement procedure. Modern stereotactic planning systems are

computer based. The stereotactic atlas is composed by a series of cross sections of an anatomical structure: in neurosurgery is the case of the human brain represented respect to a two-coordinate frame. In this way it's possible to assign to each brain structure three coordinates used for positioning the stereotactic device. The stereotactic coordinates are usually defined in an orthogonal or in a polar coordinate system. The patient's head is put in its initial fixed position or origin by using some head-holding clamps and bars. In humans, the reference points are intracerebral structures which are clearly discernible in a radiograph or tomogram. Guide bars in the x, y and z directions allow the neurosurgeon to position the point of a probe inside the brain at the calculated coordinates.

2.3.2 Point-based registration

In a point-based registration, corresponding points are identified on raw CT/MRI data and on the real patient anatomy using bone-implanted titanium screws or skin surface fiducial markers (fig. 2.3). In the first case, points are called extrinsic since they are derived from artificially applied markers; in the second case, points are called intrinsic since they are derived from patient specific image properties. The geometric center of a fiducial marker defines a fiducial point and is acquired semi-automatically or manually in the model's frame. Before the beginning of the operation, in a procedure known as patient registration, the surgeon acquires, using a navigated pointer, the position of corresponding fiducial points in the navigation system's frame. The best rigid body transformation between the two sets of points is then computed together with its error; the surgeon usually rejects the outliers until the average registration error is, in a target area, smaller than 1.0 mm.



(a) (b)

Figure 8: Surface based patient registration: patient scan (a); patient registration (b). Source: Dr. Hoppe's doctoral thesis

2.3.3 Surface-based registration

In a surface-based registration, a surface scan of the patient's area is performed automatically using a laser or a projector with structured light and is aligned against the surface of the patient's 3D generated model (fig. 8). The best transformation is obtained using a surface registration algorithm, which iteratively computes the closest points and the best transformation between two different scans. Surface-based least-squares registration methods are susceptible to poor initial pose estimates and to error contamination during intraoperative data collection. Mostly surface registration methods are based on the Iterative Closest Point algorithm [39], which has been used in several laboratory and clinical tests [40, 41].

The ICP algorithm

The ICP Algorithm was developed by Besl and McKay [39] and is used to register two given point sets in a common coordinate system. The algorithm selects iteratively the closest points as correspondences and calculates both the best rigid body transformation and the corresponding weights to minimize the weighted residuals of the sum of squared distances between all possible couples of points. In each ICP iteration the transformation can be calculated by any of these four methods: SVD [34], a quaternion method [35], an orthonormal matrices method [33] and a dual quaternions method [42]. These four algorithms show similar performances and stability versus noisy data [43]. The specificity of ICP relies then in computing the closest points between two similar point clouds by considering all possible correspondences and by assigning them a suitable weight.

2.3.4 Registration accuracy taxonomy

Errors in patient registration are classified depending on their source and on their position.

FLE or fiducial localization error is the error in determining the fiducials before patient registration. It may be due to the discretized pixel scale of an image preventing a sharp selection of a pixel position on a CT/MRI image or to the surgeon's accuracy in manually picking the corresponding fiducial point on the patient's anatomy.

FRE or fiducial registration error is the distance between corresponding fiducials after patient registration. When the best transformation between two point sets has been determined, an own registration error can be assigned to each fiducial allowing the surgeon to reject it if judged as an outlier.

TRE or target registration error, is the distance between corresponding target points other than fiducials tipically in the area of surgical interest. Because

of this, it's the most relevant surgical error.

2.3.5 Registration accuracy in OR

In the daily operation room practice, surgeons mainly perform point-based patient registrations. In this way, they can identify the required anatomical landmarks on the base of their medical experience and, even if this procedure lacks of objectivity and standardization, in the surgical community it's still the preferred method. Surgeon are used to reduce the average registration error on fiducials by rejecting outliers until, according to their judgement, the average FRE, or better the TRE in a specific area, has become smaller than 1.0 mm. This procedure is done in the operation room before beginning any surgical procedure and requires two different point clouds to be matched: the one which was planned preoperatively by the surgeon on CT/MRI data and the one which is picked on the patient's anatomy in the operation room.

2.3.6 Prediction of the target registration error

The TRE for a specific point cloud costellation can even be predicted before any actual real patient registration has happened by considering the geometry of the planned point cloud and by making reasonable assumptions on the fiducial localization error distributions. If the fiducial localization errors behave as random variables, the relationship between FRE and FLE is given by:

$$\langle \mathrm{FRE}^2 \rangle = \left(1 - \frac{2}{N}\right) \langle \mathrm{FLE}^2 \rangle \tag{5}$$

where N is the number of fiducials. The previous relationship shows, maybe surprisingly, that the fiducial registration error is independent of the fiducial configuration and geometry. On a second though, this result can be explained with assumption of a pure random distribution for the FLEs, which establishes no average correlations or spatial dependencies among the fiducials. Under the same assumptions and at the second order of the perturbation theory, the predicted TRE at the point $\mathbf{r}$ from the center of mass of the constellation can be related to the average FLE through Fitzpatrick's formula [44]:

$$\langle \mathrm{TRE}^2(\mathbf{r}) \rangle = \frac{\langle \mathrm{FLE}^2 \rangle}{N} \left(1 + \frac{1}{3} \sum_{k=1}^{3} \frac{d_k^2}{f_k^2}\right) \tag{6}$$

where N is the number of fiducials, f_k is the rms distance of the fiducial from the principal axis k and d_k is the distance of the target point from the principal axis k. This formula can be used in the planning phase, before actually performing any patient registration to carefully tune the geometry of the point cloud in order to reduce the registration error at a specific target position.

Chapter 3

The surgical scenario

The following chapter introduces endoscopic third ventriculostomy as the main surgical scenario assumed in the NEAR project.

3.1 Endoscopic third ventriculostomy

The most frequently performed endoscopic procedure in neurosurgery is endoscopic third ventriculostomy (ETV) in patients with occlusive hydrocephalus [45, 46]: here a communication between the third ventricle and the prepontine cisterne through the floor of the third ventricles re-establishes physiological CSF pressure dynamics and enables a shunt-free life for the patient.

3.1.1 Indications

There is a general consensus that the best candidates for such a procedure are patients with early stage onset of a non-tumoural acquaeduct stenosis who have never undergone diversional spinal fluid procedures. If this strict criteria are followed, ETV has a success rate of approximately 90%. Based on individual anatomy analysis, ETV can be performed in selected patients as alternative treatment to ventricular catheter placement for obstructive hydrocephalus. Hydrocephalus affects one in every 1000 live births, making it one of the most common developmental disabilities, more common than Down syndrome or deafness. There is no cure for hydrocephalus but surgery.

3.1.2 Anesthesia and positioning

The procedure is performed typically under general endotracheal anesthesia. The patient is positioned supine in a horseshoe cerebellar headrest with a small roll placed under the shoulders to elevate the chest 10 to 15 degrees.

3.1.3 Detection of the third ventricle floor

Following anesthesia, a small area of the scalp is shaved clean of hair. The television monitor is placed opposite to the surgeon. After a standard preparation of the scalp and draping of the patient, a 3 cm vertical incision based on the coronal suture is made 2.5 cm from the midline. A 1 cm burr hole is opened slightly anterior to the coronal suture. The dura is incised and coagulated to permit the entry of the introducer, a 12.5 French peel-away sheath introducer used to cannulate lateral ventricles. The rigid or flexible endoscope is inserted through the cannula into the lateral ventricle. The surgeon should inspect it and identify the Foramen of Monro and the choroid plexus. The endoscope is advanced into the third ventricle: mammillary bodies, infundibulum and optic chiasm should come into view. The floor of the third ventricle pulses freely with each heartbeat. Many of the structures may be visualized beneath the attenuated floor including the clivus, dorsum sellae and basilar artery. The site for fenestration is selected. If the floor is transparent, the fenestration should be performed between the clivus and mammillary bodies, slightly posterior to the infundibulum. If the floor is translucent or opaque, the inexperienced surgeon should consider abandoning the procedure to avoid inadvertent serious injury of the basilar artery. In such a circumstances, an experienced surgeon will select the stained area of the floor of the third ventricle that is immediately posterior to the infundibular recess.

3.1.4 Perform the fenestration

The right positioning of the fenestration at the base of the third ventricle is of extreme importance to avoid brain and vascular damages. The perforation of the base must be done halfway of the line which ideally connects the infundibular recess and the mammillary bodies, just behind the dorsum sellae. To perform the fenestration, various techniques are possible. If the floor is attenuated, it is easiest to bring the scope into direct contact with the floor and gently advance the entire scope through the floor into the interpeduncular cistern. If the floor is not attenuated or translucent, monopolar cautery may be used to create a slight tuft in the floor just posterior to the infundibulum. Rapid irrigation then creates a pathway to the translucent firm floor. The scope is then advanced further into the cistern. Alternatively, a Fogarty balloon may be advanced through the working channel of the scope. The balloon should be inflated when the epicenter of the balloon is aligned with the fenestration. The balloon is then deflated and then withdrawn to allow ouflow of cerebrospinal fluid. The scope should be withdrawn slowsly into the lateral ventricle and then through the entry tract. The tract should be inspected for any bleeding vessels as the scope is withdrawn from the brain. A small circular pledget of gelfoam is placed into the burr hole, followed by a small titanium burr hole plate. The scalp is closed in an anatomical fashion.

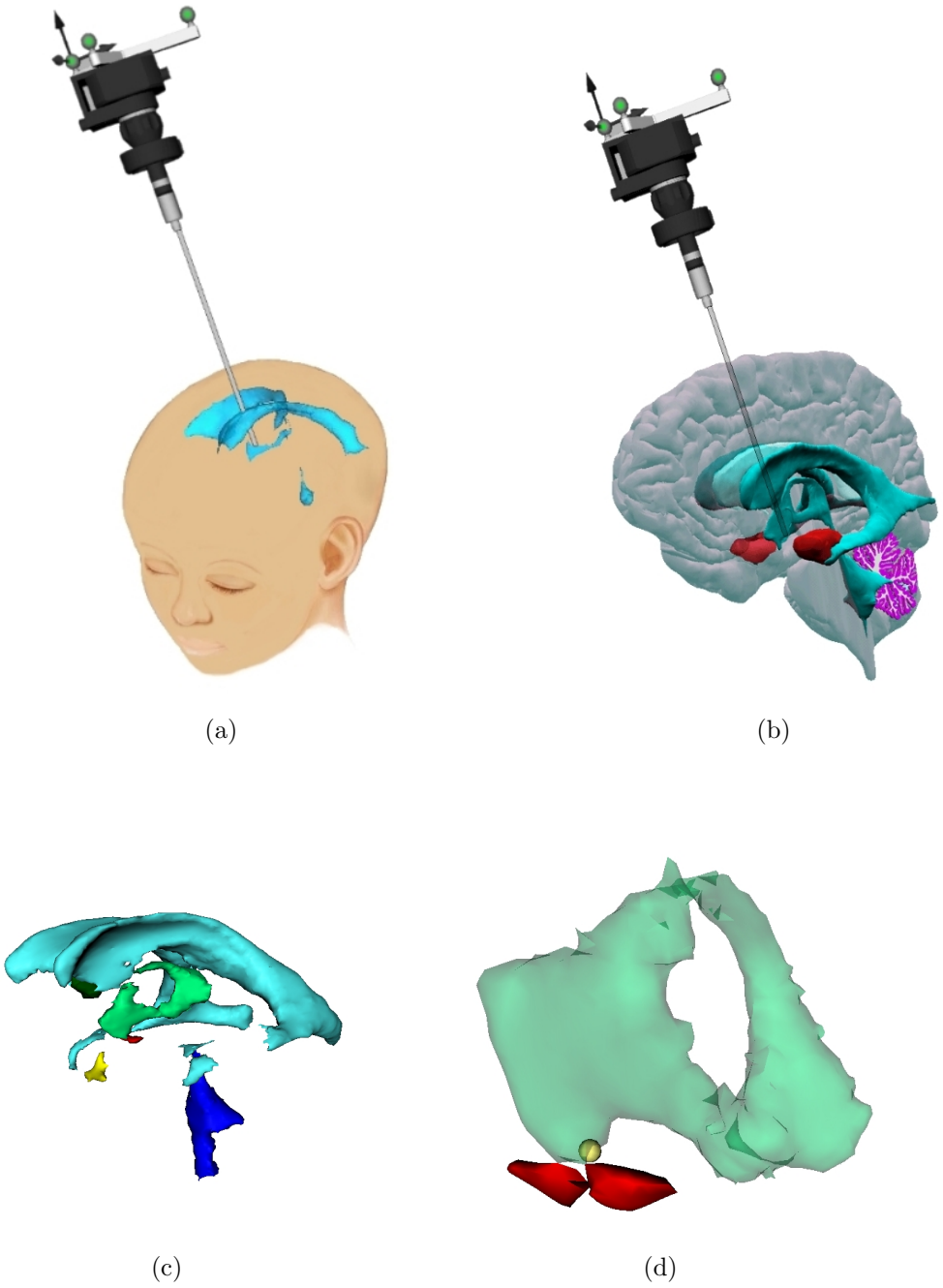
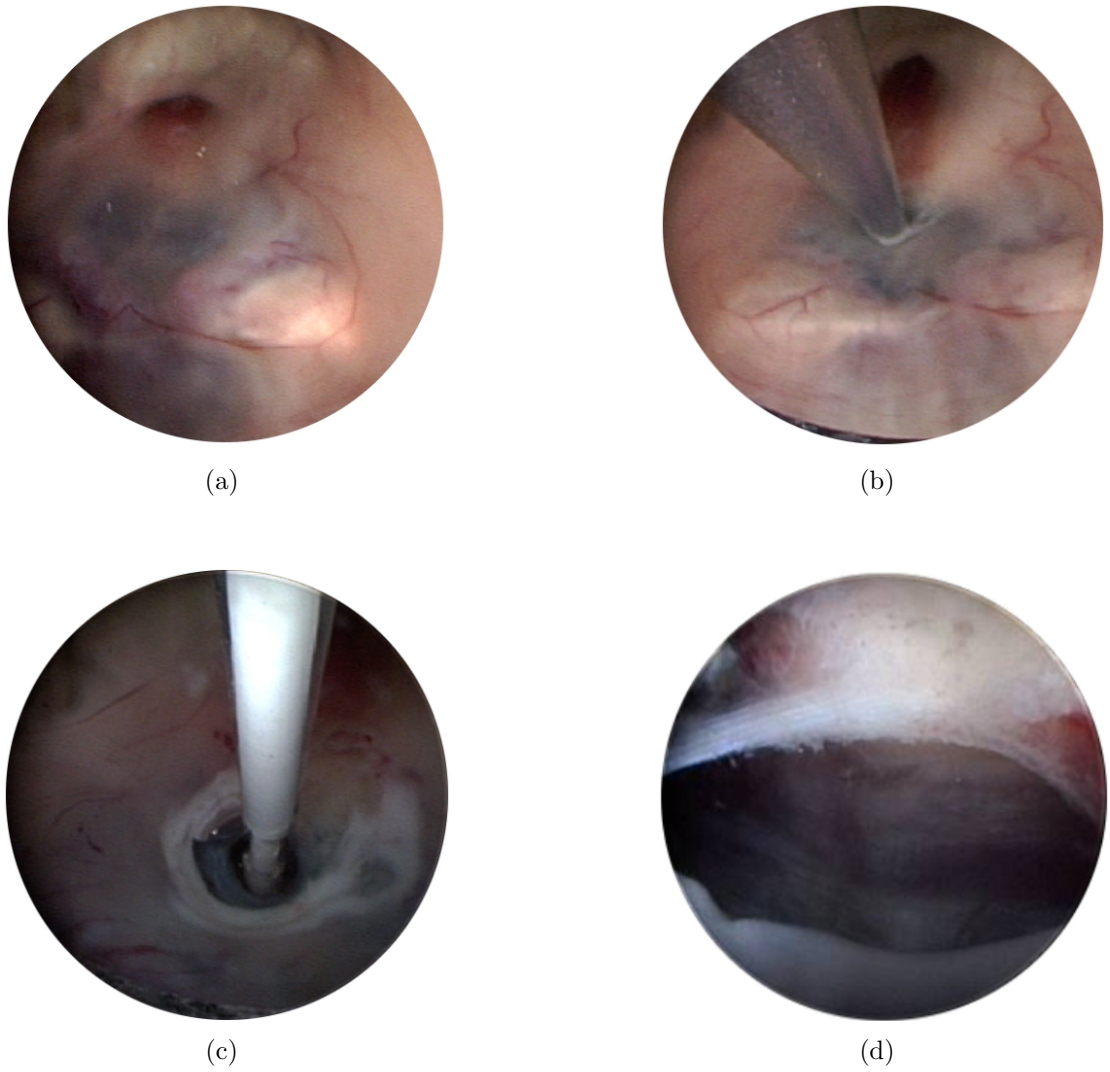


Figure 9: Endoscopic third ventriculostomy model. Patient positioning (a); endoscope insertion (b); ventricular system (c); fenestration point (d)

3.1.5 The brain shift

In neurosurgery, the brain shift effect describes an intraoperative brain deformation happening because of the change in intracranial pressure after the craniotomy,



(a) (b)

(c) (d)

Figure 10: Endoscopic third ventriculostomy views. Fenestration point (a); third ventricle floor (b); fenestration opening (c) fenestration end (d). Images courtesy of IRCCS, Pavia

elastic relaxation of soft tissues or tumour resection. As it introduces an error between the CT/MRI data space and the patient space, it's one of the most important causes affecting the overall accuracy of image-guided neurosurgical procedures. Shifts on the brain cortex in the range of of 5-20 mm have been reported by several authors [47, 48, 49] and approaches to its correction rely on intraoperative imaging techniques; since the brain shift affects also deeper structures of the brain like tumours, intraoperative MR or ultrasound imaging is another way to crrect for its deformations.

Chapter 4

State of the art

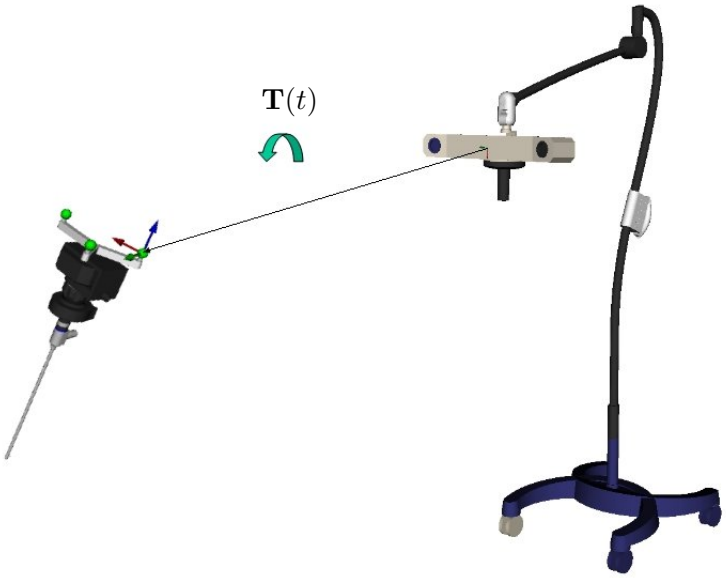
This chapter presents the state of the art in tracking systems, endoscopy, augmented reality and methods for 3D reconstruction, with a special focus on endoscopic applications. A critical review of the medical augmented reality and a short description of the most similar projects to this work is given at the end of the chapter.

4.1 Tracking systems

The process of tracing the 3D coordinates of moving objects in real-time is known as tracking. Tracking systems are devices composed by two or many stereo cameras surrounded by infrared light sources which compute the position and the orientation of special objects of known geometry called rigid bodies.

Depending on their markers, rigid bodies are distinguished between active and passive. Active markers are LEDs: they emit infrared light which is then received by the positional sensor of the tracking system. Passive markers are retroreflective spheres: they reflect back to the system the infrared light emitted from the tracking system light sources. The tracking system detects the images of each single marker from its two cameras, triangulates them and computes for each marker its 3D position in the tracker frame. By comparing the measured rigid body geometry in the tracker frame against its known rest geometry expressed in the rigid body frame, the tracking system computes the map between the two frames $\mathbf{T}(t)$, where t is the time and the transformation is a roto-translation $[\mathbf{R}, \mathbf{t}]$, with $\mathbf{R} \in SO_3, \mathbf{t} \in \mathbb{R}^3$. Each rigid body has then 6 independent degrees of freedom, 3 translational and 3 rotational. Typical tracking system frame rates range from 30 to 100 Hz.

(a) (b)



(c)

Figure 11: Passive rigid body (a); passive tracking (b); tool transformation matrix (c). Image source: Northern Digital, Inc.

4.1.1 Tracking system accuracy

Understanding how a tracking system is characterized is a task of major importance [50, 51, 52]. Optical tracking systems are characterized by moving markers throughout their measurement volume in a representative manner according to some convenient reference, whose accuracy is sufficiently better than that of the systems being characterized. For the NDI Polaris tracking system, a measuring machine moves a single marker accurately in a grid of a thousand reference positions throughout the Polaris' measurement volume. Several samples are taken at each grid point and averaged to reduce the noise. For calibration data obtained from grids of several points the spatial errors at each measured point is determined by aligning the grids and comparing the measured positions $\mathbf{m}_i$ to their corresponding reference positions $\mathbf{r}_i$ as $\epsilon_i = \mathbf{r}_i - \mathbf{m}_i$.

Single marker accuracy

For an NDI Polaris tracking system the root mean square (RMS) error over the calibration volume is 0.35 mm [53]. This average distance error is obtained by stepping a single marker throughout the calibration volume. However, tools comprised of several markers are tracked and their tracking accuracy is usually confused with single marker calibration accuracy. For every single application, many other considerations such as rigid body design, rigid body characterization, rigid body tracking algorithms, dynamic motion, the use of markers different than the ones used to characterize the system and the distance between the rigid body probes and reference tools must be taken into account. Despite these limitations, single-marker characterization results do provide a common measure for all Polaris position cameras that is independent of rigid body considerations.

Distance error

The distance error ϵ_i at each calibration grid point i is the magnitude of the underlying spatial error ϵ_i. For the NDI Polaris tracking system, errors are mostly uniform within a given xy-plane except at the upper right corners and generally increase with the distance from the camera. This type of information can be very useful for certain applications. For example, users measuring the pose of a predominately 2D object such as a plane rigid body would obtain substantially better results with the object oriented in an xy-plane than they would with the object oriented along the z-axis.

Figure 12: A rigid R. Wolf Panoview endoscope, 4mm ø, 0 °

4.2 Endoscopes

Endoscopy is a minimally invasive medical procedure used to inspect the interior surfaces of an organ by inserting a small camera into the body. To perform the operation, surgeons use endoscopes, medical devices introduced into the body to relays an image out of a confined area. An endoscope usually consists of a rigid or flexible tube, a light delivery system, a lens system transmitting the image to the viewer and additional channels to allow the entry of medical instruments. The light source is normally a separate device and the light beam is directed onto the field of view via an optical fiber system.

4.2.1 Endoscopic relay systems

From the point of view of the endoscope design, endoscopic systems are divided into three distinct subsystems: the objective, the relay and the eyepiece or video coupler [54, 55]. The relay system is often used to classify the entire design. Lens endoscopes typically have three to five relay systems; the lens diameter lies in the range from 3-7 mm. From the production and design point of view the goal is to have as few as possible elements and as few cemented components as possible. Endoscopic systems are distinguished between flexible and rigid ones: flexible relay systems include fiber-optic and electronic designs; rigid relay systems can be conventional, Hopkins rod lens, or gradient index lenses (fig. 13). Fiber optic and gradient endoscopes won't be discussed here in their details, the main focus of the work being on traditional rigid endoscopes.

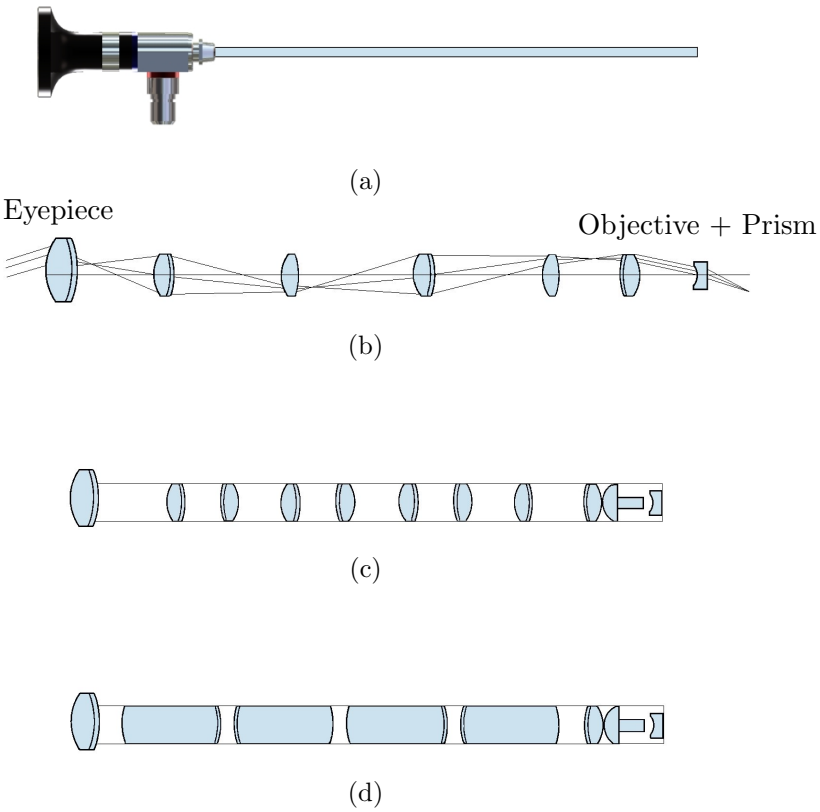


Figure 13: A rigid endoscope (a) and its optical layout (b); conventional relay system (c); Hopkins relay system (d)

Conventional rigid endoscopes

Conventional relay systems consists of a train of identical stages whose number depends on the overall system length. The traditional endoscope employs a series of achromatic doublets for the relay optics and the medium between the field lens and the relay objective lens is air.

Hopkins rigid endoscopes

Hopkins relay systems introduce on the other hand glass rods lenses, longer lenses with length to diameter ratios as high as 10. The advantage of rod lens design is the increase in the light throughput, which is proportional to the square of the refraction index of the relay rod, or, equivalently, the reduction of the ray divergence in the air gaps between the lenses, which reduces vignetting. This is deduced by analysing the first Lagrange invariant of the system, which expresses the conservation of energy

through the beam:

$$nyu = n'y'u'$$

If the index of refraction is the same, $n = n' = 1$ and the product of the object height times the object aperture equals the product of the image height times the image aperture. If instead $n' = 1.8$ in the image space with the object height unchanged, the object aperture is 80% larger and the image is more than three times brighter.

Gradient index endoscopes

Gradient index lenses have a radially-decreasing refractive index acting like a conventional converging lens. Their optical surfaces are flat compared to classical ones which simplifies the mounting of the lens by increasing the quality joint between the lens and, for example, an optical fiber.

Fiber optics endoscopes

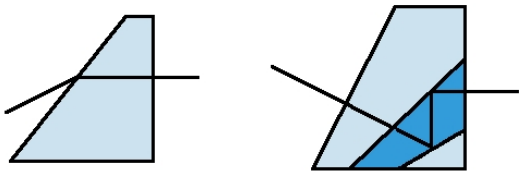
The first-order optics of a fiberoptic endoscope is straightforward: an objective lens system produces an image onto one end of a 2-dimensional array of clad optical fibers. Each fiber, with a diameter on the order of 10 micrometers with a typical bundle containing several hundred thousand fibers, receives one pixel of information about the image which is relayed to the other end of the bundle.

Endoscope prism

In all the procedures where the area of interest is tilted from the axis of the endoscope, a small prism at the tip of the endoscope is used to re-direct the field of view to the side [56]. As an advantage, rotating the endoscope allows the surgeon to easily increase its effective field of view. It may be of interest to remark that more effective prisms are reflecting ones (fig. 14). Refracting prisms are actually easier to manufacture. They are added and polished after the endoscope needle has been assembled, but are less efficient, deviate the field of view by 10^o in saline solutions and introduce lateral chromatic aberration and severe distortions. Reflecting prisms are instead more expensive, can have a cylindrical cross section, are mounted in the same inner tube as the other components and deviate the field of view by 30^o.

Sterilization

A consideration unique to medical endoscope design is sterilization. Most surgical instruments are sterilized prior to use in an autoclave, in which a combination of high temperature and pressure kills the bacteria and the spores present. Autoclaving an endoscope can be detrimental by two mechanisms: a normal endoscope



(a) (b)

Figure 14: A refraction (a) and a reflection endoscope prism (b)

because of its small size has a low thermal inertia and the inside of the scope will experience high temperatures during the sterilization. Normal lens adhesives will separate at these temperatures and even with high temperature adhesives meant to resist to the termical stress, small amounts of steam infiltrate the endoscope interior at each autoclave cycle eventually leading to irreversible fogging of the optics. To overcome the problem of a true sterilization, most hospitals allow a disinfection of their endoscopes with a solution of activated glutaraldehyde, a colorless liquid whose molecular formula is $C_5H_8O_2$ which kills the bacteria but not the spores. On defense of this practice, it must be said that the infection rates in hospitals which perform such a disinfection process are extremely low.

4.3 Augmented reality

Augmented reality (AR) is a field of computer research where 3D virtual objects are registered and blended in real time into a camera video. The information context of a real video is extended or augmented using graphics like simple annotations or 3D models. The virtual objects used to define an augmented reality application satisfy three properties: they are introduced into a real video; they are interactively blended in real time; they are registered against the real world and transform consistently to their registration. The two real and virtual components however don't share the same properties: a VR environment is one in which the participants or observers are totally immersed in a completely synthetic world which may or may not mimic the properties of a real-world environment; in contrast, a strictly real-world environment clearly is constrained by the laws of physics.

4.3.1 The reality-virtuality continuum

In 1994 Paul Milgram introduced its reality-virtuality (RV) continuum: rather than regarding the two concepts of reality and virtuality as antitheses, he proposed a classification where real and virtual world objects are presented together [57], [58].

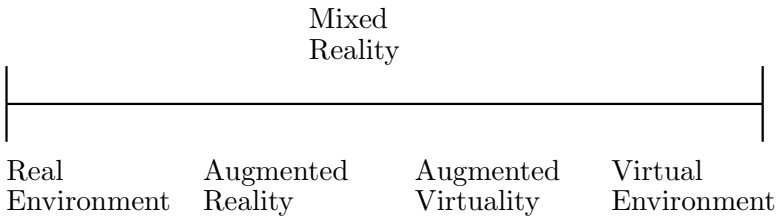


Figure 15: Milgram's reality-virtuality continuum

In the figure 15 the case at the left of the continuum defines any environment consisting solely of real objects, and includes whatever might be observed when viewing a real-world scene either directly in person, or through some kind of a window, or via a video display. The case at the right defines environments consisting solely of virtual objects, examples of which would include conventional computer graphic simulations, either monitor-based or immersive. Within this framework it is straightforward to define a generic mixed reality environment as one in which real world and virtual world objects are presented together within a single display, that is, anywhere between the extrema of the RV continuum.

The reality-virtuality continuum: critical review

The reality-virtuality continuum is used to define in a theoretical way the existence of a natural midpoint between reality and virtual reality. At the end of the 80's, the virtual reality (VR), defined by its fully immersive virtual environment which excluded any connection with the reality, started to show its limits. A need for an integration between the virtual planning information of a 3D object and its visual representation most of the time already available through video surveillance cameras, began to rise in the community of CAD designers, engineers and computer vision experts. Augmented reality realized such a need and the reality-virtuality continuum legitimated the status of AR beside VR among the new promising visualization techniques. Unfortunately, what was expressed by the continuum was in reality more a desire or a research objective of the technological community than a real achievement. The continuity expressed by the reality-virtuality diagram encompasses, for example, the immersion of a virtual cube in the real environment, that is, describes realistically its transformation properties when the camera viewpoint is moved, but begins to reach its limits when extended to include enlightments problems or when scale transformations are used to compare the information content of the surfaces of real and augmented reality objects which are not continuously updated with new information.

4.3.2 Medical augmented reality

Augmented reality (AR) generates a composite view from real images and corresponding virtual views and continues to draw the main attention of the surgical community with more and more augmented reality systems tested on patients. Advantages offered by AR are a direct and intuitive visual introduction in the intraoperative context of the preoperative CT/MRI information. Position and geometry of organs, critical structures or tumours, as well as the surgical planning are all examples of preoperative information. Active research fields in endoscopy propose to support surgeons with a broad spectrum of endoscope features: D. Dey describes in [16, 17, 18] an automated fusion of freehand endoscopic brain images to create stereoscopic panoramas; U. Bockholt uses in [11] AR to import the surgical planning during image-guided surgery; Konen introduces in [19] virtual intraoperative views to inspect otherwise inaccessible regions; Scholz shows in [20] how to build live databases of endoscopic images by saving couples of endoscope positions and views to offer stored views in case of profuse bleeding.

Medical augmented reality: critical review

Endoscopic images are augmented with virtual details even when the correspondence of preoperative models with the real scene is questioned by the elastic deformations occurring during an operation. Virtual models immersed in a real scene can indeed provide reliable information only when two obvious conditions are satisfied: their global geometrical correspondence with the real scene must remain above a certain level of accuracy and the preoperative 3D models used to augment the images must remain up-to-date for the whole life of the operation. While the first condition usually happens in a lot of augmented reality systems, the first one is in surgical practice not always true. In neurosurgery, for instance, the brain-shift effect or the change in intracranial pressure due to cerebro-spinal fluid (CSF) losses do modify the patient's state; the patient preoperative model used becomes then rapidly obsolete. In this case, a second intraoperative MRI scan performed after craniotomy is commonly accepted as a practical solution to correct the brain shift problem. More generally, any tissue removal due to standard surgical practice modifies the actual patient state and raises issues about the drop of significance of virtual models against the timeline of the operation. The aim of augmented reality systems is actually to increase the visual surgeon perception.

4.3.3 Quantitative endoscopy

3D reconstruction of observed scene using a monocular endoscope is clearly presented by C. Wengert in [59, 26, 60] in the context of markerless endoscopic registration and tracking. By tracking natural landmarks over multiple views, a 3D reconstruction of the surgical scene is obtained using photogrammetric methods.

The reconstruction is used for 3D-3D registration of the anatomy to the preoperative data and for further referencing. With the goal of performing 3D metric reconstruction of the observed scene, quantitative endoscopy could be seen as an extension to augmented reality, where intraoperative geometrical information would be first extracted, measured and then reintroduced into the observed scene. From the surgical point of view, quantitative endoscopy would become a possible bottom-up method to continuously keep models up-to-date.

4.4 Beyond the state of the art

In this section the current state of the research in augmented reality and endoscopy is presented.

4.4.1 The ETH project

The project of C. Wengert, [61], [59], developed at ETH, Zürich, proposes a fully non-invasive optical approach using a tracked monocular endoscope to reconstruct the surgical scene in 3D using photogrammetric methods. The 3D reconstruction can be used for matching the pre-operative data to the intra-operative scene. In order to cope with the near real-time requirements for referencing, a novel 3D point management method during 3D model reconstruction is used. The prototype system, with its a reconstruction accuracy of 0.1 mm and its tracking accuracy of 0.5 mm on phantom data, is one of the most accurate AR endoscopic systems.

4.4.2 The ARGUS Project

The ARGUS project [62], developed at Tübingen, uses a VectorVision image guided surgery device of the Navigation Suite BrainLAB and the public library ARTtoolkit as a basis for an AR application. Even if no endoscope is involved and the full AR capabilities are provided by the ARTtoolkit library, the project is an effort towards the integration of existing AR techniques into standard navigational IGS devices.

4.4.3 The VN project

The VN project [63], developed in Bochum, uses a R. Wolf rigid endoscope and an optical tracking system to store calibrated endoscopic images together with their current endoscope position. The system uses previously stored endoscopic images sampled during the approach inside the operative field to control the red-out phenomenon. If the real endoscopic image is lost due to bleeding, the VN system enables graphic overlay of the coagulation fiber into such images.

Chapter 5

Camera calibration

The main contribution of this chapter is a novel camera calibration model for endoscope fish-eye lenses. The work stems from a previous idea of Dr. Hoppe, who formulated a new and easy Tsai-derived camera calibration model with very intuitive parameters. The model was extensively tested on low distortion industrial lenses but was found unsatisfying on endoscopic fish-eye lenses. The main difference with Dr. Hoppe's work lies in the computation of the distortion center, which has been decoupled from the iterative solution of the algorithm and is obtained by using general properties of distortion lines. The modified model extends Hoppe's model on fish-eye lenses and points out an unknown limitation of the original one.

5.1 Camera devices

Cameras are usually described as central ray-based sensing devices [64]. A camera is said central if all its rays intersect at a single point called camera optical center. Camera calibration in 3D computer vision is a technique to extract metric information from 2D images [65]. In a calibrated camera a usually small set of parameters allows to represent the central camera projection function associating to every 2D point on the camera plane its corresponding 3D optical ray in space. A concrete representation of the perspective central projection is the pinhole model. Camera devices are approximated most of the times as perspective cameras, where a pure perspective projection function gets distorted to a certain degree by the lens distortion. Depending only on the distortion model, a camera lens can then have both a radial and tangential distortion, both of which are usually modeled as polynomials in pixel coordinates of a certain degree or as combinations of more general functions. As the focus of this work will mostly be on perspective camera models, the following definitions will be given within the contex of a perspective projection.

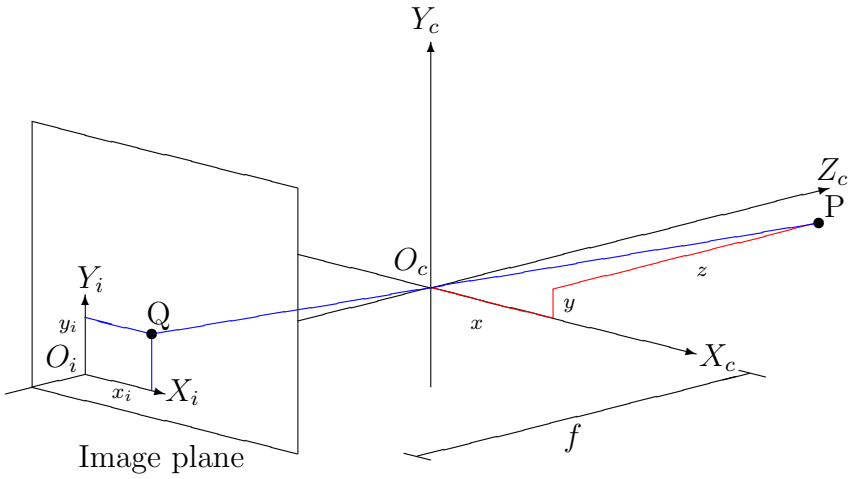


Figure 16: The object point P and its perspective projection Q in the pinhole camera model

5.1.1 Pinhole model

A pinhole camera defines a camera as an extremely small hole realizing a pure perspective projection. In this model the camera center or pinhole defines where the camera aperture is located. The image is observed on the image plane, a perpendicular plane lying at distance f from it called pinhole principal distance. The names focal distance or focal length are also widely found in literature even if at this stage no lenses and no defocus are possible. The perpendicular axis to the camera plane passing through the optical center is called the camera principal axis or principal ray and its intersection with the camera plane defines the model principal point. An observed object point P is projected by the line passing through P and O_c on the image plane at its image point Q. The line connecting P and Q is called optical ray through P (fig. 16).

The pinhole camera model realizes a perspective transformation between a 3D point and its 2D projection onto the image plane. The transformation between point and image coordinates is expressed by introducing two frames: a camera frame with its origin set at O_c and its Z_c axis perpendicular to the image plane, and an image frame, parallely translated respect to the first one along the Z_c axis at distance f. The camera plane origin O_c has coordinates $(0, 0, f)$ in image frame. If an object point P can be expressed as (x_c, y_c, z_c) in camera frame and as (x_i, y_i) in image frame, the relationship between both is derived considering similar triangles by the perspective equations:

$$\frac{x_i}{f} = \frac{x}{z} \quad , \quad \frac{y_i}{f} = \frac{y}{z} \tag{7}$$

The map from 2D to 3D described by a pinhole camera is actually a perspective

projection followed by a 180^o rotation around the Z axis: this introduces a minus sign in the previous equations; to avoid it, an implicit 180^o rotation is usually performed in the image plane in both direction, which is equivalent to define a virtual image plane at distance $+f$ on the camera Z axis. Homogeneous 3D vectors (wx, wy, w) with $w \neq 1$ are equivalent to the 2D point (x, y) by normalization of the third component and are indicated with $(\tilde{x}, \tilde{y}, \tilde{w})^T$. The perspective transformation between image points and camera vectors can be expressed then as:

$$\begin{pmatrix} \tilde{x} \\ \tilde{y} \\ \tilde{w} \end{pmatrix} = \begin{pmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} , x_i = \frac{\tilde{x}}{\tilde{w}} , \; y_i = \frac{\tilde{y}}{\tilde{w}}. \tag{8}$$

Image coordinates have been until now implicitely expressed in world units, that is, millimeters; if the horizontal and vertical pixel distances d_x, d_y are not identical, two different scale factor f_x and f_y must be used to convert them in pixel units. In addition, pixels are usually measured from an image corner; the principal point has then pixel coordinates (x_0, y_0). Using:

$$x = x_0 + \frac{x_i}{d_x} , \; y = y_0 - \frac{y_i}{d_y} , \; f_x = \frac{f}{d_x} , \; f_y = \frac{f}{d_y} \tag{9}$$

equation (8) between the image coordinates in pixels and the camera coordinates in millimeters:

$$\begin{pmatrix} \tilde{x} \\ \tilde{y} \\ \tilde{w} \end{pmatrix} = \begin{pmatrix} f_x & s & x_0 & 0 \\ 0 & f_y & y_0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} , \; x = \frac{\tilde{x}}{\tilde{w}} , \; y = \frac{\tilde{y}}{\tilde{w}}. \tag{10}$$

where a possible non-unit skew factor s between x and y image axis has been introduced.

It must be remarked that the pinhole model is a pure ideal projection: as such, it doesn't include any optical lenses to focus light and therefore any optical aberration like the blur of unfocused objects, the finite aperture effects, the diffraction or geometric distortion of the image. The only possible behaviour described by this model is the image magnification caused by pure perspective transformations.

Intrinsic parameters

Intrinsic parameters define the relationship between camera-centric coordinates and image coordinates. The pinhole model described so far uses five parameters: the camera focal lenghts f_x and f_y associated to x and y pixel-to-unit scale factors, the pixel coordinates of the principal point (x_0, y_0) and the skew factor s, which

describes instead a non-unit pixel ratio which is equivalent to say that the angle between the x and y image axis is not 90°.

Both different values for f_x and f_y and a non-unit s originate from possible inaccuracies of the camera setup during its manufacture. In the case of a non-unit s however, an additional complication must be considered: since the discrete nature of image sampling is not preserved in the signal convertion on ordinary sensors, the horizontal spacing between pixels in the sampled image cannot correspond to the spacing between cells in the image sensor. This effect can be described by saying that images are cropped in the horizontal direction because of inaccuracies in the timing during the reading process. On the other side in the vertical direction the sampling is controlled directly by the spacing of cells. This difference in the sampling process along horizontal and vertical directions does generate a possible non-unit pixel ratio and justifies the choice of two different focal lenghts for the camera.

Extrinsic parameters

Extrinsic parameters define the relationship between world-centric coordinates and camera-centric coordinates. The calibration pattern used [66] defines the world frame and the extrinsic parameters give the camera pose respect to the calibration pattern as a roto-translation. Since a rotation $\mathbf{R} \in SO_3$ has three degrees of freedom as well as a translation $\mathbf{t}$, six additional parameters are enough to describe the camera position relative to some external frame. If $\mathbf{x}_w$ are world system coordinates of a point, $\mathbf{x}_c$ its camera system coordinates and $\mathbf{C}$ the pinhole with world coordinates $(x_C, y_C, z_C)^T$:

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \mathbf{R} \begin{pmatrix} x_w - x_C \\ y_w - y_C \\ y_w - z_C \end{pmatrix} \ , \ \mathbf{C} = \begin{pmatrix} x_C \\ y_C \\ z_C \end{pmatrix} \tag{11}$$

The previous equation can be rewritten as:

$$\begin{pmatrix} \tilde{x} \\ \tilde{y} \\ \tilde{w} \end{pmatrix} = \begin{pmatrix} f_x & s & x_0 \\ 0 & f_y & y_0 \\ 0 & 0 & 1 \end{pmatrix} \mathbf{R}[\mathbf{I}|-\mathbf{C}] \begin{pmatrix} x_w \\ y_w \\ z_w \\ 1 \end{pmatrix} \tag{12}$$

where $\mathbf{I}$ is the unit 3x3 matrix and $\mathbf{P}$ is the camera projection matrix defined as:

$$\begin{pmatrix} x \\ y \\ w \end{pmatrix} = \mathbf{P} \begin{pmatrix} x_w \\ y_w \\ z_w \\ 1 \end{pmatrix} \tag{13}$$

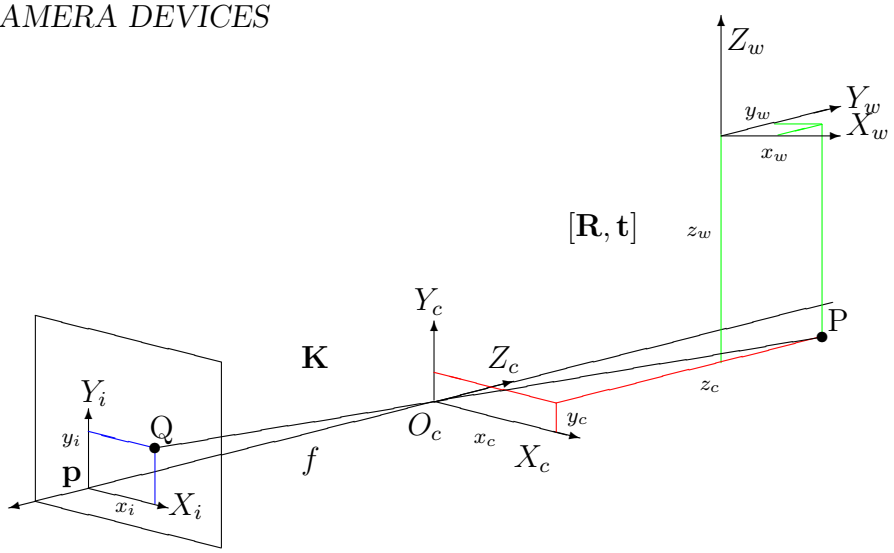


Figure 17: The image, camera and world frames with the camera-to-image matrix $\mathbf{K}$ and the world-to-camera matrix $[\mathbf{R}, \mathbf{t}]$

5.1.2 Projection matrix

In the most general case the camera projection matrix $\mathbf{P}$ is a 3x4 matrix determined up to a scale factor: 11 degress of freedoms are to be determined. Assuming no noise, since every 3D point generates 2 constraints, at least 6 points would be enough to reconstruct the whole projection matrix. In presence of noise however, $n >= 6$ correspondencies define an overdetermined linear system which can be solved with the pseudo-inverse or with the SVD method.

Once the projection matrix $\mathbf{P}$ is known, its left 3x3 submatrix can be QR-decomposed in an upper triangular matrix $\mathbf{K}$ and an orthogonal matrix $\mathbf{R}$. The calibration matrix $\mathbf{K}$ can be derived after decomposition up to a scale using:

$$\mathbf{P} = \mathbf{K}[\mathbf{R}|\mathbf{t}] \ , \ \mathbf{t} = -\mathbf{RC} \tag{14}$$

The scale factor can be fixed at the end by setting $k_{33} = 1$.

5.1.3 Optical interpretation of camera parameters

Cameras are in reality optical systems before than perspective ones. The camera principal axis is called optical axis and defined as the axis passing through the center of curvature of each mirror, lens or catadioptric surface. The camera plane is called focal plane and defined as the plane on which the camera lens is on focus. For compound optical systems like real cameras, front and back principal planes are defined to concentrate any possible refraction effects on them, letting light rays *appearing* on the back principal plane as if they had cross the front principal plane at the same distance from the optical axis. Principal planes are crucial in defining

the optical properties of any optical compound system, since it is the distance of the object and image from the front and rear principal planes that determines the magnification of the system according to a perspective law. With the identification of pinhole with optical center, optical axis with principal axis and camera plane with focal plane allowed by the theory of thin lenses, the same parameters have found different names depending on the perspective or optical context meant.

5.2 Image distortion

5.2.1 Image distortion as a special optical aberration

Pinhole cameras are ideal perspective models: they can be used as a first approximations of real cameras, but it must be taken in mind that to mimic a real camera all its optical aberrations must be taken into account. Optical aberrations are imperfections in image formation by an optical system. To the lowest degree of perturbation theory where the angle θ between a ray and the optical axis can be approximated as small:

$$\sin(\theta) = \theta - \frac{1}{3}\theta^3 + O(\theta) \tag{15}$$

they include defocus, spherical aberration, coma, astigmatism, curvature of field and image distortion. Cromatic aberration stems from the fact that different colors or wavelengths of light travels in refracting media along different paths or equivalently, that the refraction index of a material depends weakly on the considered wavelength. In general, due to aberrations, rays of light proceeding from any object point don't unite in an image point whose image is blurred; this is true for all aberrations but image distortion which preserves the one-to-one correspondence between object and image points. While the first class of aberrations convolves the image of a single imaged point with the images of neighbour points and affects the resulting image at a point level, distortion affect instead the geometry of the whole image at image level. As a consequence, distortion can be completely removed with a one-to-one transformation. The removal of image distortion, called in computer vision image warping, will be the topic of the next section.

5.2.2 Distortion models

Distortion is a form of optical aberration, a deviation from rectilinear projection in which straight lines in a scene remain straight in an image. Although distortion can be irregular or follow many patterns, the most commonly encountered distortions are approximately radially symmetric and derive from the symmetry of camera lens. Distortion happening in the tangential direction is called tangential distortion and is also well known.

Radial distortion

Radial distortion at image plane level is usually classified of two kinds, depending on its dominant component: barrel and pincushion. In the barrel distortion the image magnification decreases with the distance from the optical axis. The apparent effect is that of an image which has been mapped around a sphere. In the pincushion distortion the image magnification increases with the distance from the optical axis. The visible effect is that the lines that do not go through the centre of the image are bowed inwards, towards the centre of the image. Lens distortion can be modeled by expressing distorted radial coordinates r_d as a function of undistorted radial coordinates r_u:

$$r_d = r_u(1 + k_1 r_u^2 + k_2 r_u^4) \tag{16}$$

A mixture of both types, sometimes referred to as moustache distortion, is less common but not rare. It starts out as a barrel distortion close to the image center and gradually turns into a pincushion distortion towards the image periphery.

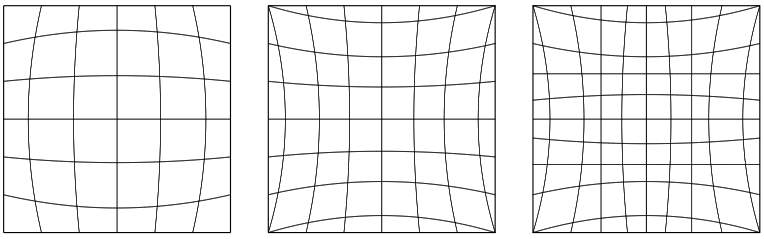


Figure 18: Lens distortion: from left, 2nd order barrel and pincushion distortions; 4th order moustache distortion.

Tangential distortion

Tangential distortion is due to decentering [67], a misalignment of the lens components relative to the optical axis in a compound lens and to a prism effect, a simple shift in image position arising when a thin prism placed in front of a lens. It is worth noticing that this distortion model was first introduced by Conrady in 1919 and carefully analysed by Brown in 1966. The model, called plumb bob model because of its usage of analytical plumb lines, extends the simple radial model by including decentering and thin prism components. The tangential distortion is usually modeled as:

$$\begin{aligned} x_d &= x_u + x_u(k_1 r_u^2 + k_2 r_u^4) + [2p_1 x_u y_u + p_2(r_u^2 + 3x_u^2)] + s_2 r_u^2 \\ y_d &= y_u + y_u(k_1 r_u^2 + k_2 r_u^4) + [2p_2 x_u y_u + p_2(r_u^2 + 3y_u^2)] + s_1 r_u^2 \end{aligned} \tag{17}$$

were $r_u^2 = x_u^2 + y_u^2$ are undistorted pixel coordinates, the first addends in the round brackets in the above relation describe the 4^{th} order radial distortion (k_1, k_2), the second addends in square brackets describe the 2^{nd} order decentering distortion (p_1, p_2) and the third addends describe the 2^{nd} order thin prism distortion (s_1, s_2). Since the decentering and the thin prism distortion can be grouped together, the tangential lens distortion is then modeled redefining the coefficients p_1, p_2 as:

$$\begin{aligned} x_d &= x_u + x_u(k_1 r_u^2 + k_2 r_u^4) + [2p_1 x_u y_u + p_2(r_u^2 + 2x_u^2)] \\ y_d &= y_u + y_u(k_1 r_u^2 + k_2 r_u^4) + [2p_2 x_u y_u + p_2(r_u^2 + 2y_u^2)] \end{aligned} \tag{18}$$

For ordinary camera lenses, the tangential distortion is usually an order of magnitude or more smaller than radial distortion. This fact explains why the order of the polynomial used for the tangential equation is lower than the oder of the one used to model radial distortion [68].

Undistortion filter implementation

In the previous section, the careful reader will have noted that the tangential distortion function (18) is not analitically invertible. This is in contrast with the radial distortion (16) where an analytical solution is found by easily solving a biquadratic equation. The tangential distortion function must be then numerically inverted when the residuals of the distorted pixels position are compared with their prediction to optimize the distortion parameters.

Algorithm 1 Undistort pixels coordinates x_{di}

for $i < n$ **do**
  $x = (x_{di} - c_x)/f_x \ , \ y = (x_{di} - c_y)/f_y$
  $x_0 = x \ , \ y_0 = y$
  for $j < 5$ **do**
    $r^2 = x^2 + y^2$
    $c_d = (1 + k_0 r^2 + k_1 r^4)$
    $\Delta_x = 2k_2 xy + k_3(r^2 + 2x^2)$
    $\Delta_y = k_2(r^2 + 2y^2) + 2k_3 xy$
    $x = (x_0 - \Delta_x)/c_d$
    $y = (y_0 - \Delta_y)/c_d$
    $x_{ui} = x \ , \ y_{ui} = y$
  end for
end for

It may be however of interest to note that, unless one doesn't want to compute the residuals of the distorted and undistorted pixel coordinates, undistorting a full image requires only the direct formula and that no explicit invertion is really needed. The direct distortion function, which gives distorted pixel coordinates (x_d, y_d) as

Algorithm 2 Table lookup for image undistortion

```
typedef struct LUTentry {
  unsigned int offset;
  unsigned char f[4];
};
LUTentry LookUptable[w*h];
LUTentry *LUT = LookUptable;
```

for $j < h - 1$ **do**
  for $i < w - 1$ **do**
    $r^2 = (x_{d_{ij}} - x_c)^2 + (y_{d_{ij}} - y_c)^2$
    $x_u = x_{d_{ij}} + (x_{d_{ij}} - x_c)\left[k_0 r^2 + k_1 r^4 + 2k_2 y_{d_{ij}} + k_3\left(r^2/x_{d_{ij}} + 2x_{d_{ij}}\right)\right]$
    $y_u = y_{d_{ij}} + (y_{d_{ij}} - x_c)\left[k_0 r^2 + k_1 r^4 + 2k_3 x_{d_{ij}} + k_2\left(r^2/y_{d_{ij}} + 2y_{d_{ij}}\right)\right]$
    if $x_u < w, y_u < h$ **then**
      `LUT->offset` $= 3(\lfloor x_u \rfloor + w\lfloor y_u \rfloor)$;
      $u = x_u - \lfloor x_u \rfloor$, $v = y_u - \lfloor y_u \rfloor$
      `LUT->f[0] = (int)(255.0*(1-u)*(1-v));`
      `LUT->f[1] = (int)(255.0*u*(1-v));`
      `LUT->f[2] = (int)(255.0*(1-u)*v);`
      `LUT->f[3] = (int)(255.0*u*v);`
    else
      `LUT->offset = 0;`
      `LUT->f[0] = 0;`
      `LUT->f[1] = LUT->f[2] = LUT->f[3] = 255.0;`
    end if
    `LUT++;`
  end for
end for

function of the ideal undistorted pixel coordinates (x_u, y_u), allows to undistort the image alone (alg. 2).

When a pixel is selected on the final buffer of the undistorted image (fig. 19), the direct formula allows to compute its correspondent distorted position. Since the selected pixel has integer coordinates but the transformed doesn't, the distorted pixel position identifies a 2x2 square of pixels defined by the integer part and the mantissa for each of its horizontal and vertical coordinates. A simple bilinear interpolation of their positions is saved into a table lookup to increase the future access speed for the corresponding distorted pixel positions.

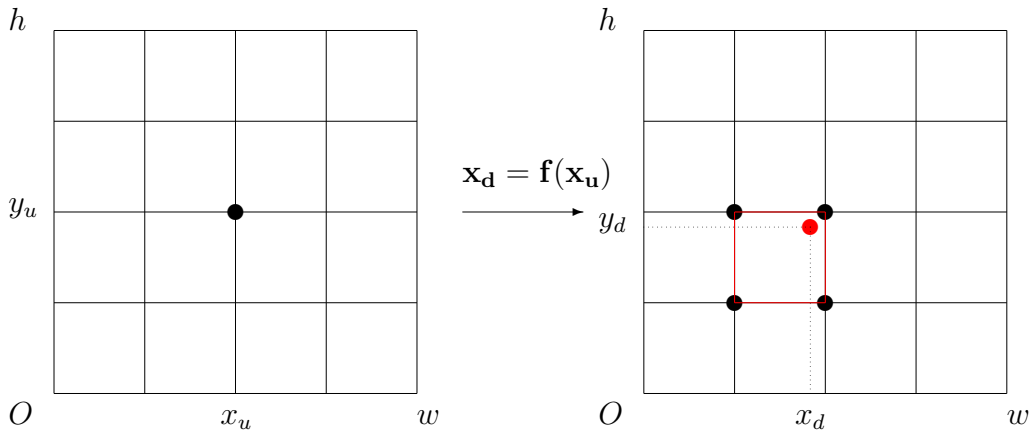


Figure 19: Undistorted pixel intensity obtained by bilinear interpolation of the nearest neighbour distorted pixels

5.3 Camera calibration models: an overview

Camera calibration models describe cameras as a small finite set of parameters [69, 70, 71, 72, 65, 73, 74, 75]. They are defined by both the set of parameters chosen to describe the camera properties and by the procedure used to compute them. Mathematically, the problem of the camera calibration reduces to the problem of finding the projection matrix $\mathbf{P}$ and its decomposition as $\mathbf{P} = \mathbf{K}[\mathbf{R}, \mathbf{t}]$. As we will see, the way a particular class of model parameters is defined does impact on their overall meaning. A primary consequence of all this is that even corresponding parameters computed using different algorithms do represent slighly different properties of the same object and can be mapped one into another only with great care.

5.3.1 DLT Method

First attempts to calibrate cameras go back to old photogrammetric techniques. The first camera calibration algorithm developed during the 70s is considered to be the direct linear transformation method (DLT) proposed by Y. I Abdel-Aziz and H. M. Karara in their early paper [76]. If the (i, j) element of the camera projection matrix $\mathbf{P}$ is p_{ij}, each correspondence between 2D pixels (x, y) and 3D world points (x_w, y_w, z_w) gives two equations:

$$x = \frac{p_{11}x_w + p_{12}y_w + p_{13}z_w + p_{14}}{p_{31}x_w + p_{32}y_w + p_{33}z_w + p_{34}}$$

$$y = \frac{p_{21}x_w + p_{22}y_w + p_{23}z_w + p_{24}}{p_{31}x_w + p_{32}y_w + p_{33}z_w + p_{34}}$$

In this method the projection between 3D points and corresponding 2D pixels is written first as similarity relation and then as a homogeneous linear equation. The previous equation can be rewritten as:

$$\begin{pmatrix} -x_w & -y_w & -z_w & -1 & 0 & 0 & 0 & 0 & xx_w & xy_w & xz_w & x \\ 0 & 0 & 0 & 0 & -x_w & -y_w & -z_w & -1 & yx_w & yy_w & yz_w & y \end{pmatrix} \begin{pmatrix} p_{11} \\ \vdots \\ p_{34} \end{pmatrix} = \mathbf{0}$$

and solved by standard techniques. The method is quite general and relies on the property that a similarity constraint can be rewritten as two correspondent constraints in the perpendicular plane.

5.3.2 Non-linear minimization

Like all linear methods, DLT has however a serious drawback: no lens distortion can be included within the model. That means, the pixel coordinates (x, y) used in equation (19) should be the ideal undistorted pixel coordinates, whose knowledge would require in turn the camera pose, which is exactely what the DLT method aims to determine. The problem is in general iteratively solved by minimizing on the image plane the total reprojection error:

$$\min_{\mathbf{P}} \sum_i \|\mathbf{x}_i - \mathbf{P}\mathbf{X}_i\|^2 \tag{19}$$

where $\mathbf{x_i}$ are 2D pixels and $\mathbf{X_i}$ are 3D points. Non-linear iterative techniques like the Levenberg-Marquardt algorithm are particularly well suited to these class of problems. Like every iterative non-linear method however, the final solution of the equation (19) depends strongly on the choice of the initial guess. The solution of the linear problem obtained by neglecting the lens distortion becomes then the first guess for all non-linear minimization algorithms used.

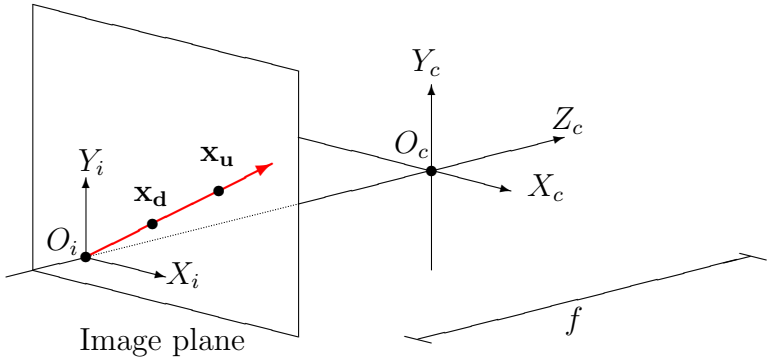


Figure 20: The radial alignment constraint

5.3.3 Tsai's method

In the mid 1980's, R. Y. Tsai introduced a new technique for the determination of the camera extrinsic parameters which proved to be highly successful and very popular. Tsai introduces the so called Radial Alignment Constraint (RAC) to model the lens distortion: this constraint depends only on the assumption that the distortion is a central and radial field and will be briefly summarized in the following (fig. 20). Assuming the principal point $O_i(x_0, y_0)$ coordinates to be known, image coodinates (x_I, y_I) can be expressed on the image plane as:

$$x'_I = x_i - x_0 \ , \ y'_I = y_i - y_0 \tag{20}$$

these coordinates must satisfy the relationship:

$$\frac{x'_I}{f} = s\frac{x_c}{z_c} \ , \ \frac{y'_I}{f} = s\frac{y_c}{z_c} \tag{21}$$

Under a pure radial distortion, the direction of the point in the image plane as measured from the principal point is independent on the unknown focal distance f and on the radial distortion function as the following equation shows:

$$\frac{x'_I}{y'_I} = s\frac{x_c}{y_c} \tag{22}$$

where $O_i(x_c, y_c, z_c)$ are pinhole coordinates. As a consequence, the vector connecting the origin in the image plane to the distorted point $\mathbf{x_d}$ is radially aligned and parallel to the vector extending from the optical axis to the undistorted object point $\mathbf{x_u}$ *for*

every radial distortion. Expanding the second term and using the components of the rotation matrix $\mathbf{R}$, the previous equation can be written as:

$$\frac{x'_I}{y'_I} = s\frac{r_{11}x_S + r_{12}y_S + r_{13}z_S + t_x}{r_{21}x_S + r_{22}y_S + r_{23}z_S + t_y} \tag{23}$$

giving a linear homogeneous equation in eight unknowns for every pixel-point correspondence.

The strenght of the RAC constraint is that it doesn't depend on any explicit form for the function used to model radial lens distortion, neither on the focal length, nor on the z coordinate of the translation vector. Its introduction allows to formulate a two stage radially-decoupled algorithm by reducing the dimensionality of the parameter space. A reduced radial pose consisting of the camera 3d orientation, its x-axis and y-axis translation can be computed at the first stage of the algorithm. The result is used as first estimate in a successive non-linear optimization where the effective focal length, the distortion coefficients and the z-axis translation are computed using homogeneous equations. The camera calibration problem with radial distortion is therefore reduced to a special DLT problem performed on a smaller subset of parameters. Like the DLT method, Tsai's algorithm requires a good initial guess of the second set of parameters; as a main improvement over DLT, the first set of parameters already includes all the radial distortion effects generated by the lens.

Tsai's method: a critical review

As a retrospective comment, it can be said that there was no reason to look for such a decoupling of the problem but to include as much as possible in the initial guess the lens distortion effects. The main motivation was therefore more an algorithmic one: the awareness of a good initial guess in the non-linear refinement of the parameters in the DLT procedure required a method to include at least part of the lens distortion effects into it. The assumption of dealing with a total radially simmetric lens distortion was going exactly in this direction.

From the physical point of view however, a better decoupling in the computation of the parameters would have rather distinguished between intrinsic and extrinsic ones instead of between radial and non radial parameters. The sensitivity of the algorithm to the parameter errors was even different among simple translation parameters.

As a brief and final remark about its method: since its first original article, the author pointed out a tiny drawback of his algorithm. Quoting R. Y. Tsai [69]: "The results of the real experiments show that when a full resolution CCD camera is calibrated with the proposed technique, it is so well equipped as to be able to make 3D measurement with one part in 4000 average accuracy. To see the consequence of having a wrong guessed image center when doing calibration, we intentionally alter

the apparent image center by ten pixels. The results of 3D measurement still is about as accurate". R. Tsai was able to reach remarkable subpixel precision on the image plane with many possible solutions spread over an area well wider than the algorithm precision. The proposed technique was in any case a major breakthrough in machine vision and influenced the field for over twenty years.

5.3.4 Heikkila's method

A refinement of two-steps Tsai's model was proposed at the beginning of the 90s by J. Heikkila and O. Silven [77]. Heikkila refines the two stage model of Tsai adding two steps more and taking into account two previously neglected effects. In the third step he compensates for a perspective distortion caused by circular features and distinguishes at image plane level between interpolated centroids centers and projected pattern circle centers (fig 21).

In the fourth step he corrects the distorted image coordinates by implicitely inverting a distortion model with radial and tangential components. The proposed iterative method gives a direct solution of the back-projection problem by numerically inverting the non-analytically invertible direct distortion function. This method for numerically inverting distortion models is implemented as described in the source code of OpenCV, the open source library for computer vision (alg. 1).

5.3.5 Zhang's method

The next major milestone in camera calibration models came at the end of the 90s. Z. Zhang [65] observed for the first time the constraints on the camera intrinsic parameters provided by observing single planes. Expressing these constraints as homographies, he was able to formulate and solve with a closed form solution a calibration problem for intrinsic parameters only. He gave therefore an answer to the long standing question concerning the observed slight differences and shifts in intrinsic parameters values when different calibration were performed from different camera poses.

Homographies

The calibration method of Z. Zhang relies upon images of the same planar calibration pattern, a chessboard whose homogeneous world coordinates will be indicated with capital letters and whose pixel coordinates with small-case letters. If the plane $Z = 0$ is used as calibration plane and $\mathbf{r_i}$ are the columns of the rotation matrix $\mathbf{R}$

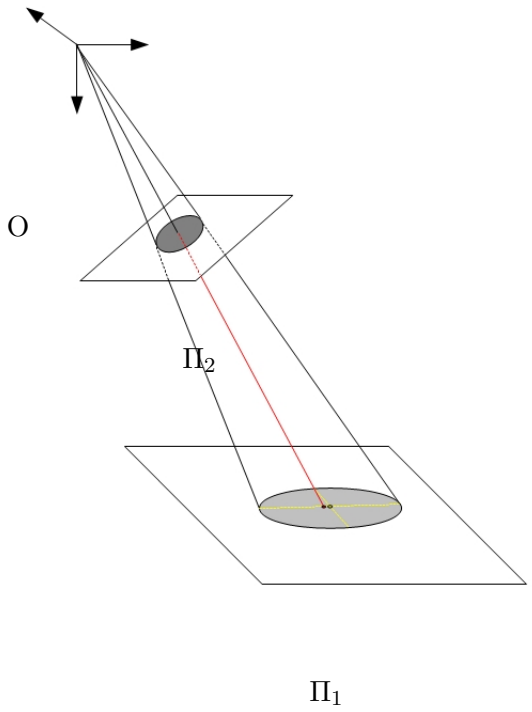


Figure 21: Perspective projection of a circle according to Heikkila: the projection of a center is not the centroid of the projected ellipse.

the pixel point correspondence is:

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \mathbf{K} \begin{bmatrix} \mathbf{r}_1 & \mathbf{r}_2 & \mathbf{r}_3 & \mathbf{t} \end{bmatrix} \begin{pmatrix} X \\ Y \\ 0 \\ 1 \end{pmatrix} = \mathbf{K} \begin{bmatrix} \mathbf{r_1} & \mathbf{r_2} & \mathbf{t} \end{bmatrix} \begin{pmatrix} X \\ Y \\ 1 \end{pmatrix}$$

The transformation between corresponding pixels over different images is expressed with the 3x3 homography matrix **H**:

$$\lambda \mathbf{K} \begin{bmatrix} \mathbf{r}_1 & \mathbf{r}_2 & \mathbf{t} \end{bmatrix} = H = \begin{pmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{pmatrix}$$

The matrix is defined up to a scale factor and has 8 degrees of freedom: 4 correspondencies are then enough to determine it with standard techniques used to solve ordinary linear systems. Since rotation vectors $\mathbf{r}_1, \mathbf{r}_2$ are othogonals, from the previous equation two constraints are obtained:

$$\mathbf{h}_1^T \ \mathbf{K}^{-T}\mathbf{K}^{-1} \ \mathbf{h}_2 = 0 \tag{24}$$

$$\mathbf{h}_1^T \ \mathbf{K}^{-T}\mathbf{K}^{-1} \ \mathbf{h}_1 = \mathbf{h}_2^T \ \mathbf{K}^{-T}\mathbf{K}^{-1} \ \mathbf{h}_2 \tag{25}$$

where the 3x3 symmetric matrix is defined as:

$$\mathbf{K}^{-T}\mathbf{K}^{-1} = \omega = \begin{pmatrix} \omega_{11} & \omega_{12} & \omega_{13} \\ \omega_{21} & \omega_{22} & \omega_{23} \\ \omega_{31} & \omega_{32} & \omega_{33} \end{pmatrix} \tag{26}$$

The symmetric matrix are written using its corresponding 6D vector ω; if the i^{th} column of $\mathbf{H}$ is $\mathbf{h}_i$, we rewrite as:

$$\omega = [\omega_{11}, \omega_{12}, \omega_{22}, \omega_{13}, \omega_{23}, \omega_{33}]$$

$$\mathbf{h}_i^T \omega \mathbf{h}_j = \mathbf{v}_{ij}^T \omega$$

with:

$$\mathbf{v}_{ij} = [h_{i1}h_{j1}, h_{i1}h_{j2} + h_{i2}h_{j1}, h_{i2}h_{j2}, h_{i3}h_{j1} + h_{i1}h_{j3}, h_{i3}h_{j2} + h_{i2}h_{j3}, h_{i3}h_{j3}]^T$$

The previous two fundamental constraints from a given homography (24) are written as:

$$\begin{pmatrix} \mathbf{v}_{12}^T \\ (\mathbf{v}_{11} - \mathbf{v}_{22})^T \end{pmatrix} \mathbf{b} = \mathbf{0} \tag{27}$$

If $n \geq 3$ of such images are observed, the final linear system:

$$\mathbf{V}\mathbf{b} = \mathbf{0} \tag{28}$$

obtained by stacking the equations can be solved up to a scale factor. The solution of (28) is the right singular vector of $\mathbf{V}$ associated with the smallest singular value. Once ω is estimated, the intrinsic parameters of the matrix $\mathbf{K}$ are easily computed:

$$\begin{aligned}\mathbf{r}_1 &= \lambda \mathbf{A}^{-1}\mathbf{h}_1 \\ \mathbf{r}_2 &= \lambda \mathbf{A}^{-1}\mathbf{h}_2 \\ \mathbf{r}_3 &= \mathbf{r}_1 \times \mathbf{r}_2 \\ \mathbf{t} &= \lambda \mathbf{A}^{-1}\mathbf{h}_3\end{aligned}$$

with $\lambda = 1/\|\mathbf{A}^{-1}\mathbf{h}_1\| = 1/\|\mathbf{A}^{-1}\mathbf{h}_2\|$. Because of noise in data, the so-computed matrix $\mathbf{R} = [\mathbf{r}_1; \mathbf{r}_2; \mathbf{r}_3]$ does not in general satisfy the properties of a rotation matrix and must therefore be projected via a SVD decomposition onto the rotation group SO_3. Extrinsic parameters are then computed by relying on the intrinsic parameter guess. A final Levenberg-Marquardt step using the functional:

$$\sum_{i=1}^{n}\sum_{j=1}^{m}\|\mathbf{m}_{ij} - \mathbf{m}(\mathbf{K}, \mathbf{R}_i, \mathbf{t}_i, \mathbf{M}_{ij})\|^2$$

where $\mathbf{m}(\mathbf{K}, \mathbf{R}_i, \mathbf{t}_i, \mathbf{M}_{ij})$ is the projection of the point M_j in the image i is required to minimize the difference between the detected and the projected points.

Maximum likelihood estimation

The Zhang's method models the lens distortion with a radial fourth order polynomial:

$$\begin{aligned}x_d &= x_u + x_u(k_1 r_u^2 + k_2 r_u^4) \\ y_d &= y_u + y_u(k_1 r_u^2 + k_2 r_u^4)\end{aligned} \tag{29}$$

The first 5 intrinsic parameters are estimated assuming weak distortion; the corresponding distortion parameters are then computed by alternating these two steps and refining iteratively the found solution until convergence. Experimentally better results are obtained by minimizing the full functional:

$$\sum_{i=1}^{n}\sum_{j=1}^{m}\|\mathbf{m}_{ij} - \mathbf{m}(\mathbf{K}, k_1, k_2, \mathbf{R}_i, \mathbf{t}_i, \mathbf{M}_{ij})\|^2$$

where $\mathbf{m}(\mathbf{K}, k_1, k_2, \mathbf{R}_i, \mathbf{t}_i, \mathbf{M}_{ij})$ is the projection of the point M_j in the image i followed by distortion according to equation (29).

Zhang's method revisited

Even from the point of view of the distortion, Z. Zhang relied in its original paper [65] on the radial distortion showing how much the work of Tsai had been influential on his own. The current and more popular implementation of the Z. Zhang algorithm is provided by the open-source library OpenCV which includes instead a distortion model with both radial and tangential components. The advantage in using this techniques are several: a decoupling of intrinsic and extrinsic parameters coming from the homography-based formulation and an improvement in the flexibility of the overall camera calibration procedure which doesn't require any 3D rigid calibration object but can be made using simple 2D high quality chessboard print-outs glued on a rigid flat surface.

5.4 Fish-eye lenses

Fish-eye lenses are designed to cover the whole hemispherical field in front of the camera. Since their angular field of view is almost 180^o and it's impossible to project an hemisphere on a finite image plane using a perspective projection only, a pure application of the pinhole model would not be justified for such lenses. According to [78], the perspective projection of a pinhole camera and fish-eye lenses can be described as:

$$
\begin{aligned}
r &= f\tan(\theta) && \text{(perspective projection)} \\
r &= 2f\tan(\frac{\theta}{2}) && \text{(stereographic projection)} \\
r &= f\theta && \text{(equidistance projection)} \\
r &= 2f\sin(\frac{\theta}{2}) && \text{(equisolid projection)} \\
r &= f\sin(\frac{\theta}{2}) && \text{(orthogonal projection)}
\end{aligned}
$$

where θ is the angle between the optical axis and the incoming ray, r is the distance between the image point and the principal point and f is the focal length (fig. 22).

5.4.1 Kannala's model

To deal with all the lenses at once, Kannala [79] extends the projection to its general form:

$$r(\theta) = k_1\theta + k_2\theta^3 + k_3\theta^5 + \dots$$

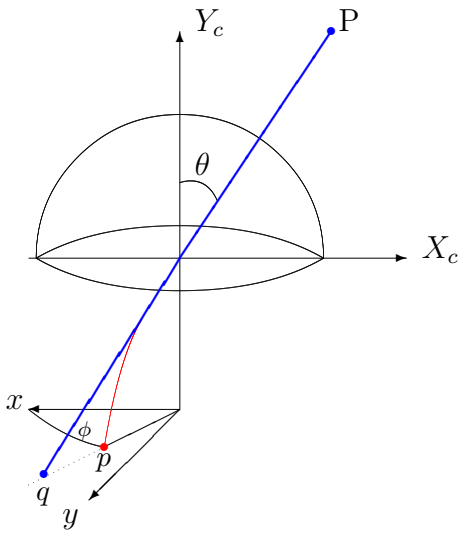


Figure 22: Fisheye camera model. The image of the point P is p whereas it would be q by a pinhole camera.

and adds the first terms of a spherical Fourier expansion to model lens radial and tangential distortion:

$$\Delta_r(\theta,\phi) = (l_1\theta + l_2\theta^3 + l_3\theta^5)(i_1\cos(\phi) + i_2\sin(\phi) + i_3\cos(2\phi) + i_4\sin(2\phi))$$
$$\Delta_t(\theta,\phi) = (m_1\theta + m_2\theta^3 + m_3\theta^5)(j_1\cos(\phi) + j_2\sin(\phi) + j_3\cos(2\phi) + j_4\sin(2\phi))$$

The model of Kannala describes fish-eye lenses as intrinsically not-perspective devices.

Kannala model reviewed

Kannala model has a tiny mathematical drawback: it approximates the fish-eye lens projection function by using a truncated Taylor expansion for $\tan(\theta)$. Because of such approximation however, the model loses the fundamental property which justified at the very beginning its extension of the projective perspective law: no hemisphere can be mapped onto a finite plane using polynomials. The model therefore implicitly justifies the usage of a pinhole model for fisheye lenses, where the tangential distortion coefficients will include any possible deviations from the pure perspective projection law. The approximation of fish-eye lenses as pinholes is obviously also valid in a suitable small neighbour of the distortion center where the pure perspective projection is almost obeyed.

Chapter 6

NEAR camera calibration model

The algorithm of Dr. Hoppe can be defined as a Tsai-derived, pure vector-based camera calibration algorithm. Its specificity relies upon a very intuitive parameters formulation. The description of the perspective camera calibration model has been for the first time presented in [80]; to let the reader get introduced with its formulation we summarize it in the following paragraphs. As is usually done in many other well established camera calibration models, Hoppe's model decouples the problem of recovering the camera pose estimation parameters, derived from a linear perspective distortion of the original plane image, from the estimation of the non-linear lens distortion parameters whose effect is immediately perceived as a bending of control straight lines of pixels [81].

The model implementation, as it has been presented in the original article, assumes the convergence of the procedure when an optically distorted solution is used as input for the perspective pose estimation problem and the optical distortions parameters are computed successively. Once the optical center position and the positions of the image plane in the world space are known, it's then possible to compare the measured distorted pixel positions against an ideal perspectively distorted pattern as observed from the computed optical center.

6.1 Model parameters

The following sections describe the model parameters and the modifications required to let the algorithm converge on endoscope fisheye lenses.

6.1.1 Perspective parameters

The projective part of Hoppe's model establishes a perspective relationship between the 2D camera plane and its immersion into the 3D space. Its only requirement is that the camera image plane matches the projection of the image sensor taken

from the optical center through a definite point. The model is thus defined by four intuitive parameters: the optical center Z and the two camera pixels vectors $\mathbf{a}$, $\mathbf{b}$ which define the 2D single pixel vectors projected onto the camera plane. The fourth model parameter is defined by the intersection of the optical axis on the camera plane and stems from that requirement that the projected camera plane should pass though the origin of the 3D world system (fig. 23). The same point on a plane can in this way be described in the two systems by its world and undistorted pixel coordinates with the following equation:

$$\mathbf{x}_u = \mathbf{a}(n_u - n_0) + \mathbf{b}(m_u - m_0) \tag{30}$$

where $\mathbf{x}_u$ is a generic world point, (n_u, m_u) are its undistorted coordinates on the image plane, $(\mathbf{a}, \mathbf{b})$ are two vectors defining a pixel and spanning the plane and (n_0, m_0) are the pixel coordinates of the world origin O.

Projecting the world point $\mathbf{x}$ onto the image plane with $\mathbf{z}$ as a pinhole, the intersection point on the image plane can be expressed in two equivalent ways:

$$\mathbf{z} + s(\mathbf{x} - \mathbf{z}) = \mathbf{a}(n_u - n_0) + \mathbf{b}(m_u - m_0) \tag{31}$$

where s is nothing but the distance between the pinhole and the projected point on the plane. With a scalar product of the previous equation by $\mathbf{b} \times (\mathbf{x} - \mathbf{z})$ and respectively by $\mathbf{a} \times (\mathbf{x} - \mathbf{z})$ one can obtain the undistorted pixel coordinates:

$$\begin{pmatrix} n_u \\ m_u \end{pmatrix} = \begin{pmatrix} n_0 \\ m_0 \end{pmatrix} + \frac{1}{(\mathbf{x} - \mathbf{z}) \cdot (\mathbf{a} \times \mathbf{b})} \begin{pmatrix} \mathbf{x} \cdot (\mathbf{z} \times \mathbf{b}) \\ \mathbf{x} \cdot (\mathbf{a} \times \mathbf{z}) \end{pmatrix} \tag{32}$$

With these two equations is easy to switch from the world coordinate to the projected camera plane. The perspective projection is computed by rewriting equation (31) in a suitable form; introducing the vectors $\mathbf{k}, \mathbf{u}, \mathbf{v}$ and the scalar factor γ defined as:

$$\begin{aligned} \gamma &= \mathbf{z} \cdot (\mathbf{a} \times \mathbf{b}) & \mathbf{c} &= (\mathbf{z} \times \mathbf{a})/\gamma \\ \mathbf{k} &= (\mathbf{a} \times \mathbf{b})/\gamma & \mathbf{d} &= (\mathbf{z} \times \mathbf{b})/\gamma \\ \mathbf{u} &= n_0\mathbf{k} + \mathbf{d} & \mathbf{v} &= m_0\mathbf{k} - \mathbf{c} \end{aligned}$$

the equation (31) can be rewritten for every pixel-point correspondence as:

$$\begin{aligned} n_u\mathbf{x} \cdot \mathbf{k} - \mathbf{x} \cdot \mathbf{u} + n_0 &= n_u \\ m_u\mathbf{x} \cdot \mathbf{k} - \mathbf{x} \cdot \mathbf{v} + m_0 &= m_u \end{aligned}$$

Given eleven unknowns is then possible to solve the system by stacking at least six pixel-point correspondences or, when more correspondences are selected or the noise

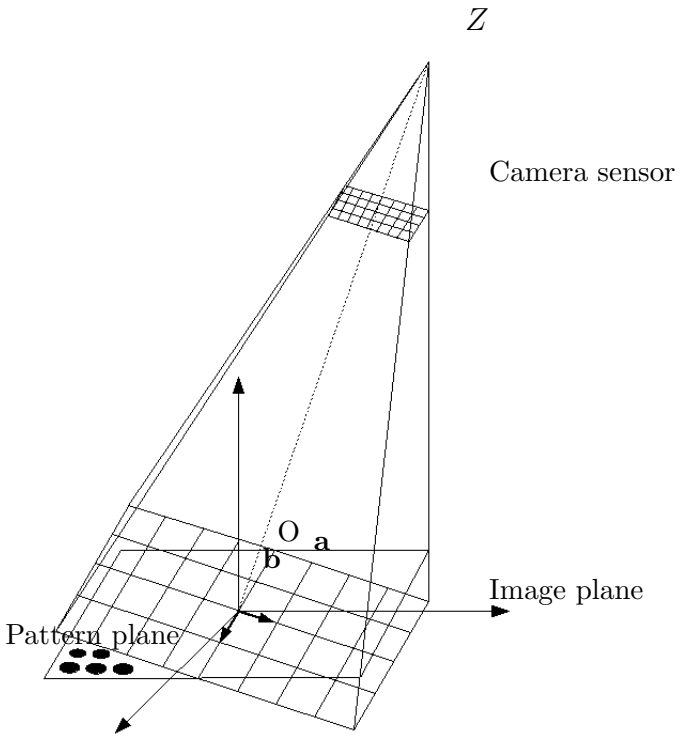


Figure 23: The perspective projection of the image sensor on the camera plane through the origin O. The pinhole Z and the projected pixel vectors **a**, **b** defining the camera plane and the pattern plane is also shown.

is considered, by the pseudo-inverse or the SVD method on the following system of equations:

$$\begin{pmatrix} n_{u_1}\mathbf{x}_1^t & -\mathbf{x}_1^t & 0 & 1 & 0 \\ m_{u_1}\mathbf{x}_1^t & 0 & -\mathbf{x}_1^t & 0 & 1 \\ n_{u_2}\mathbf{x}_2^t & -\mathbf{x}_2^t & 0 & 1 & 0 \\ m_{u_2}\mathbf{x}_2^t & 0 & -\mathbf{x}_2^t & 0 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots \end{pmatrix} \begin{pmatrix} \mathbf{k} \\ \mathbf{u} \\ \mathbf{v} \\ n_0 \\ m_0 \end{pmatrix} = \begin{pmatrix} n_{u_1} \\ m_{u_1} \\ n_{u_2} \\ m_{u_2} \\ \vdots \end{pmatrix} \tag{33}$$

The above system can be solved assuming in the first step that the distorted pixel coordinates (m_{d_i}, n_{d_i}) as measured from the image plane could approximate the ideal undistorted pixel coordinates (m_{u_i}, n_{u_i}). Solving the previous system it is possible to get the original perspective parameters back:

$$\begin{aligned} \mathbf{a} &= \gamma(\mathbf{c} \times \mathbf{k}) & \mathbf{b} &= \gamma(\mathbf{d} \times \mathbf{k}) \\ \mathbf{c} &= m_0\mathbf{k} - \mathbf{v} & \mathbf{d} &= \mathbf{u} - n_0\mathbf{k} \\ \mathbf{z} &= \gamma(\mathbf{c} \times \mathbf{d}) & \gamma &= 1/(\mathbf{k} \cdot \mathbf{c} \times \mathbf{d}) \end{aligned}$$

6.1.2 Distortion parameters

When the perspective distortion parameters are known, the undistorted pixels coordinates (n_{ui}, m_{ui}) can be computed using the equation (31). Before doing it, the pixel coordinates of the distortion center $\mathbf{f}$ must be known. These are computed by minimizing the sum of the squared surfaces built with $\mathbf{f}, \mathbf{x}_{u_i}, \mathbf{x}_{d_i}$:

$$\begin{aligned} q(n_f, m_f) &= \sum_i \frac{1}{4} \left[(\mathbf{x}_{d_i} - \mathbf{f}) \times (\mathbf{x}_{u_i} - \mathbf{f}) \right]^2 \\ &= \frac{1}{4} (\mathbf{a} \times \mathbf{b})^2 \sum_i \left[n_f \Delta m_i + m_f \Delta n_i - \Delta n m_i \right]^2 \end{aligned}$$

where $\Delta m_i = m_{d_i} - m_{u_i}$, $\Delta n_i = n_{d_i} - n_{u_i}$ and $\Delta n m_i = n_{u_i} m_{d_i} - m_{u_i} n_{d_i}$. Minimizing respect to the distortion center the coordinates (n_f, m_f), one obtains:

$$\sum_i \begin{pmatrix} \Delta m_i^2 & \Delta n_i \Delta m_i \\ \Delta n_i \Delta m_i & \Delta n_i^2 \end{pmatrix} \begin{pmatrix} n_f \\ m_f \end{pmatrix} = \sum_i \begin{pmatrix} \Delta n m_i \Delta m_i \\ \Delta n m_i \Delta n_i \end{pmatrix} \tag{34}$$

If the optical axis is othogonal to the image plane, the distortion center can be computed by perpendicularly projecting the optical center on it with the equation:

$$\begin{pmatrix} n_f \\ m_f \end{pmatrix} = \begin{pmatrix} n_0 \\ m_0 \end{pmatrix} + \frac{1}{(\mathbf{a} \times \mathbf{b})^2} \begin{pmatrix} (\mathbf{a} \times \mathbf{b})(\mathbf{z} \times \mathbf{b}) \\ (\mathbf{a} \times \mathbf{b})(\mathbf{a} \times \mathbf{z}) \end{pmatrix}$$

Hoppe's distortion model reviewed

The original model of Hoppe estimates the lens distortion parameters on the image plane by minimizing the area generated by the detected and the reprojected pixels obtained using the extimated pinhole pose. The lens parameters are then used to compute a new pose and the process is repeated again and again. This procedure has a severe drawback: when the first guess for the distortion center is too far from its true position, the risk of driving the algorithm convergence towards a spurious solution depending on the initial guess is particularly high. During the first iteration in fact, the distortion center on the image plane is computed by comparing the distorted pixel positions againts their hypothetical undistorted positions. On the first iteration however, the pose estimation is computed by completely neglecting the distortion: the distorted pixels are pretended being undistorted and the distortion center coordinates on the image plane are derived and used in successive iterations to undistort the detected pixel. The memory of the first initial guess obtained by neglecting the distortion remains in both the i^{th} pinhole position *and* in the distortion center. If the initial position of the distortion center is far from its true one, the algorithm doesn't converge.

Such effect was not perceived at all on weak distorted lens; even in Tsai's algorithm the position of the principal point, implicitely assumed to be the radial distortion center, could be shifted slighly without affecting the overall algorithm precision; R. Tsai developed its RAC constraint expressely to avoid the dependence of the algorithm convergence on a particular initial guess. The model of Dr. Hoppe converges from almost any initial guess on the low distorted lenses tested, but with high distorted lenses the problem of starting from a false distortion center shows up again.

6.1.3 New distortion parameters

It is worth to notice that the calculation of the distortion center can be performed independently of the optimization of the distortion parameters. This idea, which goes in the direction of a decoupling of the internal camera parameters, allows us to invert the order of the computations to recover Hoppe's model. Instead of first estimating an approximate pose used to guess the undistorted pixel positions, we begin by finding on the image plane a reliable distortion center. A similar strategy to compute the distortion center of a fish-eye lens out of a distorted image has been previously proposed by Asari in [82]. In the following, we are going to present it anew using some basic tools coming from homotopy theory to better highlight some of its details.

Radial distortion center

As correctly Asari points out in [82], once known the distortion center is a fixed point for a particular camera and can be used for all the images obtained for that camera. If a picture of a regular grid of control points is taken with a fish-eye lens, straight lines will be distorted to curves by its radial distortion. It is also natural for the observer to organize these curves into vertically and horizontally distorted lines. Looking at their curvature coefficients, one realizes soon that the radial distortion center must lie between couples of horizontal and vertical curves of minimum curvature and opposite sign: let's call them $\gamma_1, \gamma_2, \gamma_3, \gamma_4$. If we deform the horizontal upper curve γ_1 continuously and linearly into the horizontal lower curve γ_2 and do the same procedure with the vertical left and right distortion curves γ_3 and γ_4 we obtain two families of curves:

$$\begin{array}{ll} \gamma_h(t,x) = (1-t)\gamma_1(x) + t\gamma_2(x) & t \in [0,1] \\ \gamma_v(s,y) = (1-s)\gamma_3(y) + s\gamma_2(y) & s \in [0,1] \end{array} \tag{35}$$

In homotopy theory the interpolation above represents a continuous and linear deformation between two regular paths and defines the operation of addition between definite paths, the degree of deformation being parameterized by (t, s). As the distortion in the central region is assumed to be small, the polynomial expansion can

be limited to second degree terms: highest polynomial terms can be neglected and parabolas can be used to approximate the lines of distortion.

$$\gamma_i(x) = a_i x^2 + b_i x + c_i,\ i = 1 \dots 4 \tag{36}$$

Opposite couples of horizontal and vertical parabolas have curvature coefficients with opposite signs: there must be therefore a unique value of the parameters $(t^\star, s^\star)$ for which their curvature vanishes and each deformed curve turns into a straight line. This value defines the condition of no curvature:

$$\begin{aligned}(1 - t^\star)a_1 + t^\star a_2 = 0 \\ (1 - s^\star)a_3 + s^\star a_4 = 0\end{aligned}$$

Substituting the values of $(t^\star, s^\star)$ found in equation (35) we obtain the equations of two straight lines; their point of intersection being by construction the radial distortion center.

Radial distortion function

Once the radial distortion center has been estimated, the radial distortion function can be chosen and their coefficients can be appropriately fitted. Following the same approach shown in [80] we define the pixel coordinates of the distortion center F as:

$$\mathbf{f} = \mathbf{a}(n_f - n_0) + \mathbf{b}(m_f - m_0)$$

The radial lens distortion depends now on the distance between the optically distorted and optically undistorted pixel coordinates defined as an ideal perspective projection of a three dimensional regular grid on the image plane. We define undistorted and distorted radial vectors as usual: $r_u = \|r_u\|$, with $\mathbf{r}_u = \mathbf{x}_u - \mathbf{f}$ and $r_d = \|r_d\|$ with $\mathbf{r}_d = \mathbf{x}_d - \mathbf{f}$. The relationship between distorted and undistorted pixels can be defined choosing a proper radial distortion function. Using a Taylor expansion and differently respect to other distortion models [69], [65], all even and odd terms until fourth degree are kept.

$$\begin{aligned}\frac{r_u}{r_d} = 1 + \kappa_0 r_d + \kappa_1 r_d^2 + \kappa_2 r_d^3 + \cdots \\ \frac{r_d}{r_u} = 1 + \lambda_0 r_d + \lambda_1 r_d^2 + \lambda_2 r_d^3 + \cdots\end{aligned}$$

The relationship between distorted and undistorted coordinates becomes when expressed in world coordinates:

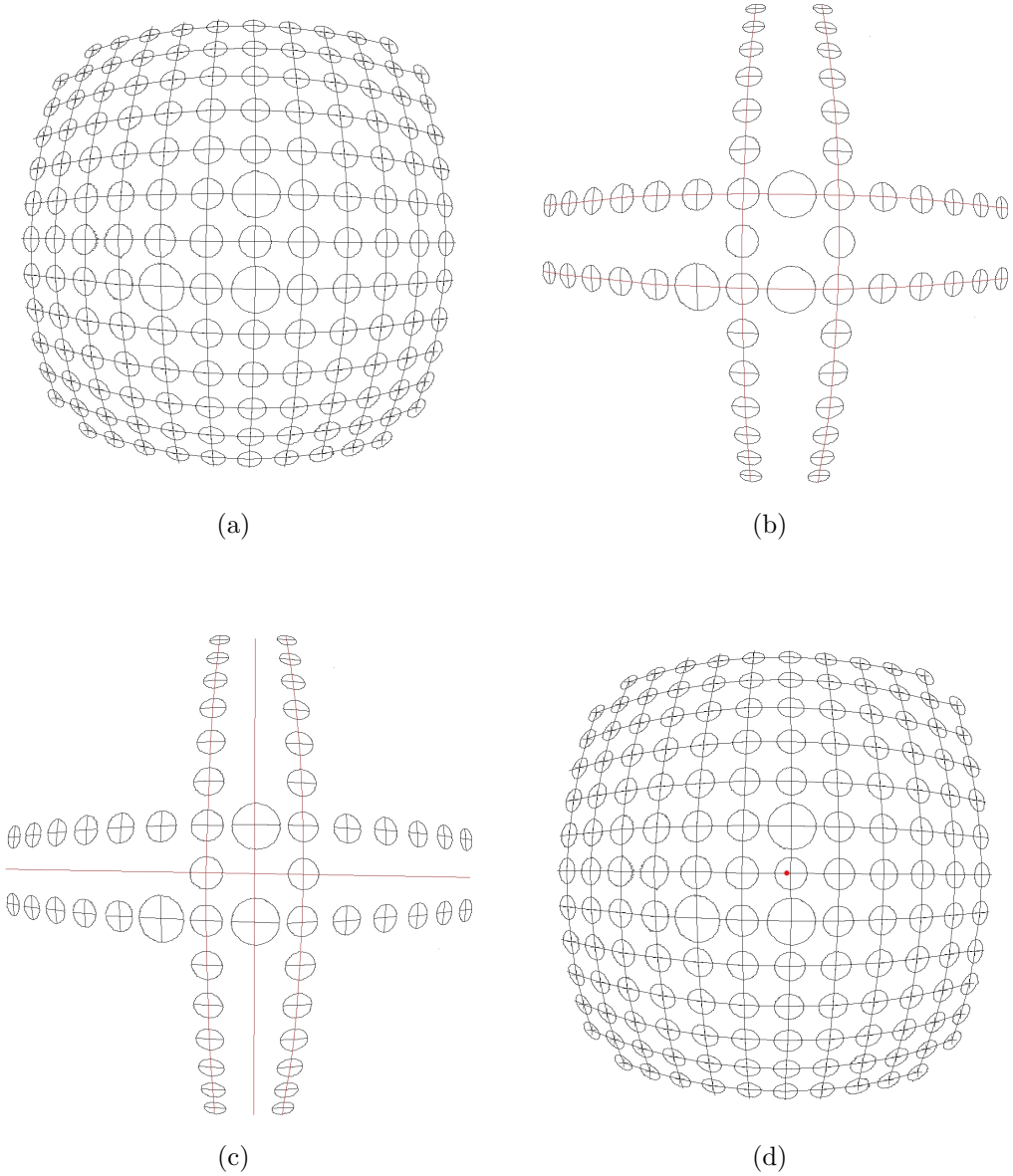


Figure 24: Detection of a control grid with a fish-eye lens: detected circles (a); horizontal and vertical minimum curvature lines (b); their deformation into straight lines (c); the distortion center (d)

$$\mathbf{x}_u = \mathbf{f} + (1 + \kappa_0 r_d + \kappa_1 r_d^2 + \kappa_2 r_d^3 + \cdots)\mathbf{r}_d$$
$$\mathbf{x}_d = \mathbf{f} + (1 + \lambda_0 r_u + \lambda_1 r_u^2 + \lambda_2 r_u^3 + \cdots)\mathbf{r}_u$$

Expressing $\mathbf{x}_d = \mathbf{a}(n_d - n_0) + \mathbf{b}(m_d - m_0)$ we derive the relationship between distorted and undistorted pixels:

$$\begin{pmatrix} n_u \\ m_u \end{pmatrix} = \begin{pmatrix} n_d \\ m_d \end{pmatrix} + (1 + \kappa_0 r_d + \kappa_1 r_d^2 + \cdots)\begin{pmatrix} n_d - n_f \\ m_d - n_f \end{pmatrix} \tag{37}$$
$$\begin{pmatrix} n_d \\ m_d \end{pmatrix} = \begin{pmatrix} n_u \\ m_u \end{pmatrix} + (1 + \lambda_0 r_d + \lambda_1 r_d^2 + \cdots)\begin{pmatrix} n_u - n_f \\ m_u - n_f \end{pmatrix} \tag{38}$$

The radial distortion parameters κ, λ can then be determined by defining the radial distorted and undistorted vectors:

$$\mathbf{r}_{u_i} = \mathbf{a}(n_{u_i} - n_{f_i}) + \mathbf{b}(m_{u_i} - m_{f_i}) \tag{39}$$
$$\mathbf{r}_{d_i} = \mathbf{a}(n_{d_i} - n_{f_i}) + \mathbf{b}(m_{d_i} - m_{f_i}) \tag{40}$$

and minimizing the following sums:

$$L_u(\kappa) = \sum_i (1 + \kappa_0 r_{d_i} + \kappa_1 r_{di}^2 + \dots)\mathbf{r}_{d_i} - \mathbf{r}_{u_i} \tag{41}$$
$$L_d(\lambda) = \sum_i (1 + \lambda_0 r_{u_i} + \lambda_1 r_{ui}^2 + \dots)\mathbf{r}_{u_i} - \mathbf{r}_{d_i} \tag{42}$$

To test the model accuracy we define:

$$D_i = \sum_i \left[\mathbf{d}_i \times ((\mathbf{z} - \mathbf{x}_i) \times \mathbf{d}_i)\right]^2 \tag{43}$$

6.1.4 Iterative convergence

After the first iteration of the procedure described before, the method is repeated until a satisfying convergence is found. The test applied for achieving a given model precision is based on minimizing its pixel residuals. In the framework of this model, the algorithm can be summarized as in the following pseudo-code.

Begin Compute the radial center (n_f, m_f) using the first image plane using method described in section 6.1.3.

Do

1. Solve the perspective over-determined system (33) by the pseudo inverse method and determine the perspective set of parameters $\mathbf{z}, \mathbf{a}, \mathbf{b}, n_0, m_0$. As first guess neglect lens distortion $(n_{u_i}, m_{u_i}) = (n_{d_i}, m_{d_i})$ and use only a central sub-region of the image. Expand it at every iteration until all image pixels have been included.

2. Obtain the undistorted pixels (n_{u_i}, m_{u_i}) with the equation (32) from a perspective projection of the regular grid of control points onto the image plane.

3. Use the pre-computed radial distortion center (n_f, m_f) to define the undistorted and distorted radial vectors with equation (39) and calculate the radial distortion coefficients (κ_i, λ_i) minimizing equation (41).

4. Obtain the undistorted (n_{u_i}, m_{u_i}) pixels with the equations (37, 38) from the distorted pixels (n_{d_i}, m_{d_i}) by removing the previously computed lens distortion.

5. Compute the direction vectors $\mathbf{d}_i = (n_{u_i} - n_0)\mathbf{a} + (m_{u_i} - m_0)\mathbf{b} - \mathbf{z}$ and the sum of the distances between the world coordinates $\mathbf{x}_i$ and the directions $\mathbf{z}_i + s\mathbf{d}_i$ by using (43).

until the convergence reduces the residuals or the difference $D_n - D_{n+1}$ is below a predefined threshold.

6.2 Experimental Results

The procedure described above has been applied for the calibration of a Sumix M72 CCD 2M pixel camera, with maximum resolution of 1600x1200, C-mount, used to acquire images from a Panoview Wolf endoscope, with 0^o optic and 6 mm of diameter. Since the goal of this paper is to describe the camera calibration model and not the pixel detection, only a brief report of the pattern identification procedure is given in the following.

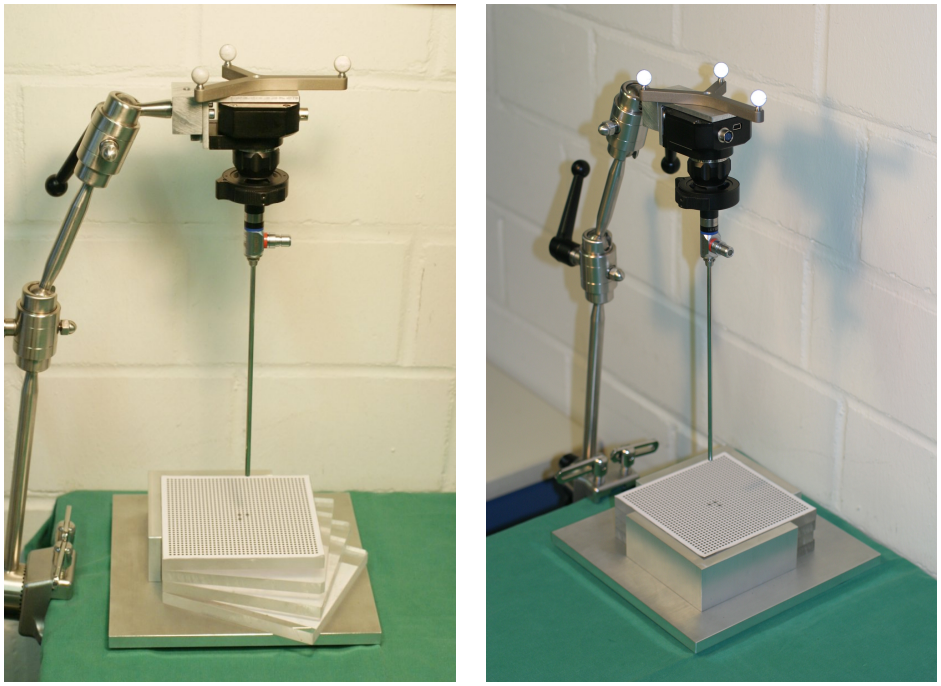
Pattern identification

A calibration pattern composed of a square grid of circles (circle diameter 1.5 mm, circle distance 3mm) was selected to acquire the distorted pixel cloud from four equidistant planes, each plane lying 10 mm from the previous one (fig. 25). The image was first denoised by appliying a gaussian filter and then binarized.

Three circles with bigger radius were chosen to identify the pattern origin and the reference system. A region-growing algorithm (see algorithm 3) was used to acquire

Algorithm 3 Region growing algorithm

```
GrowRegion(const int& X, const int& Y)
{
  int  i, x, y, lower = 0, upper = 1;
  bool  edge, hid = false;

  edgePixel->Reset(); pixArray->Reset();
  visited[Y][X] = true;
  pixArray->Add(CPixel(X, Y));

  while (upper > lower) {
    for (i=lower; i<upper; i++) {
      edge = false;

      x = pixArray->array[i].x - 1;  /* step left */
      y = pixArray->array[i].y;
      add_pixel_to_region(x, y, cutoff, aoiLeft, &hid, &edge);

      x += 2; /* step right */
      add_pixel_to_region(x, y, cutoff, aoiRight, &hid, &edge);

      x -= 1; y -= 1;   /* step above */
      add_pixel_to_region(x, y, cutoff, aoiTop, &hid, &edge);

      y += 2; /* step below */
      add_pixel_to_region(x, y, cutoff, aoiBottom, &hid, &edge);

      if (edge == true)
       edgePixel->Add(CVec2d(pixArray->array[i].x, pixArray->array[i].y));
    }
    lower = upper;
    upper = pixArray->num;

    if (hid == true || pixArray->num < minNumOfCirclePixel)
     return false;
    return true;
  }
}

void add_pixel_to_region(x, y, cutoff, aoi, int *hid, bool *edge) {
  if (x < aoi) *hid = true;
  else if (pixel[y][x] >= cutoff) *edge = true;
  else if (visited[y][x] == false) {
    pixArray->Add(CPixel(x, y));
    visited[y][x] = true;
  }
}
```



(a) (b)

Figure 25: The endoscope calibration setup with its four calibration planes each liying 10 mm apart from the next one

every circular pattern. For every circle, its middle-point was extracted with one of these two different methods: by using the center of an interpolated ellipse when the interpolation was precise enough and its radial distortion was correspondently small, or by using its center of area otherwise.

The ordering between the detected pixel cloud and the corresponding circle positions on the pattern was performed in this way: first, the three big circles were found by selecting the first three circles with the biggest pixel area. Second, the middle point of the three big circles was identified by looking at the mutual distances among their centroids, the center circle being the one with the smallest distance from the other two (fig. 27a). Third, the x and y vectors between centroids were identified and their cross product used to define a right-handed frame (fig. 27b).

Once identified the line directions from the pattern origin, the algorithm started a search along these basis directions by looking for the next closest point at distance $\mathbf{x}_n - \mathbf{x}_{n-1}$ from the current position. In case of low distortion, a linear search is enough to order all the blobs (fig. 27c). In case of high distortion, the nearest circle from the search point at a radial distance smaller than a definite cutoff was selected (fig. 27d); if no center was found within the required distance, the search along the

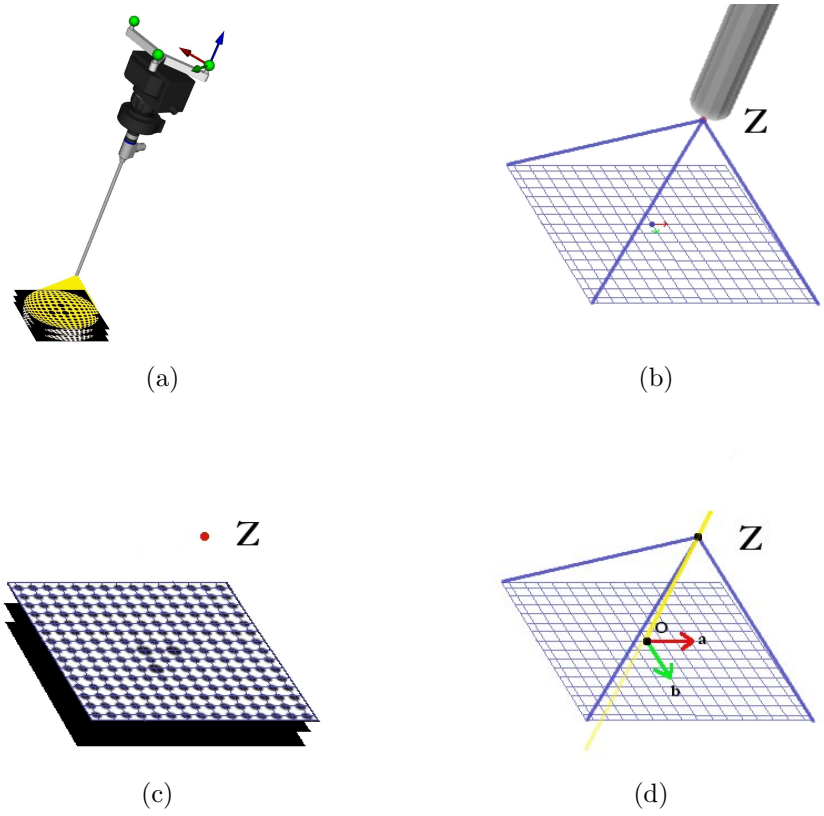


Figure 26: The pose estimation using the modified algorithm. The pose is shown as a small pyramid representing the four optical rays from the pinhole position to the four corners of the almost squared frame enclosing the circular observed endoscopic image.

chosen direction was stopped. The search direction was updated whenever a new next neighbour was found until all the points were examinated.

Comparison

The figure 28 shows the algorithm's original convergence using Dr. Hoppe estimation of the distortion center and its modified convergence obtained by using distortion lines to extrapolate the initial guess for the distortion center. The other two magnified images show details pointing out the original principal point wrong first guess obtained using equation 34 and its corresponding position obtained instead by using continuously deformed distortion lines.

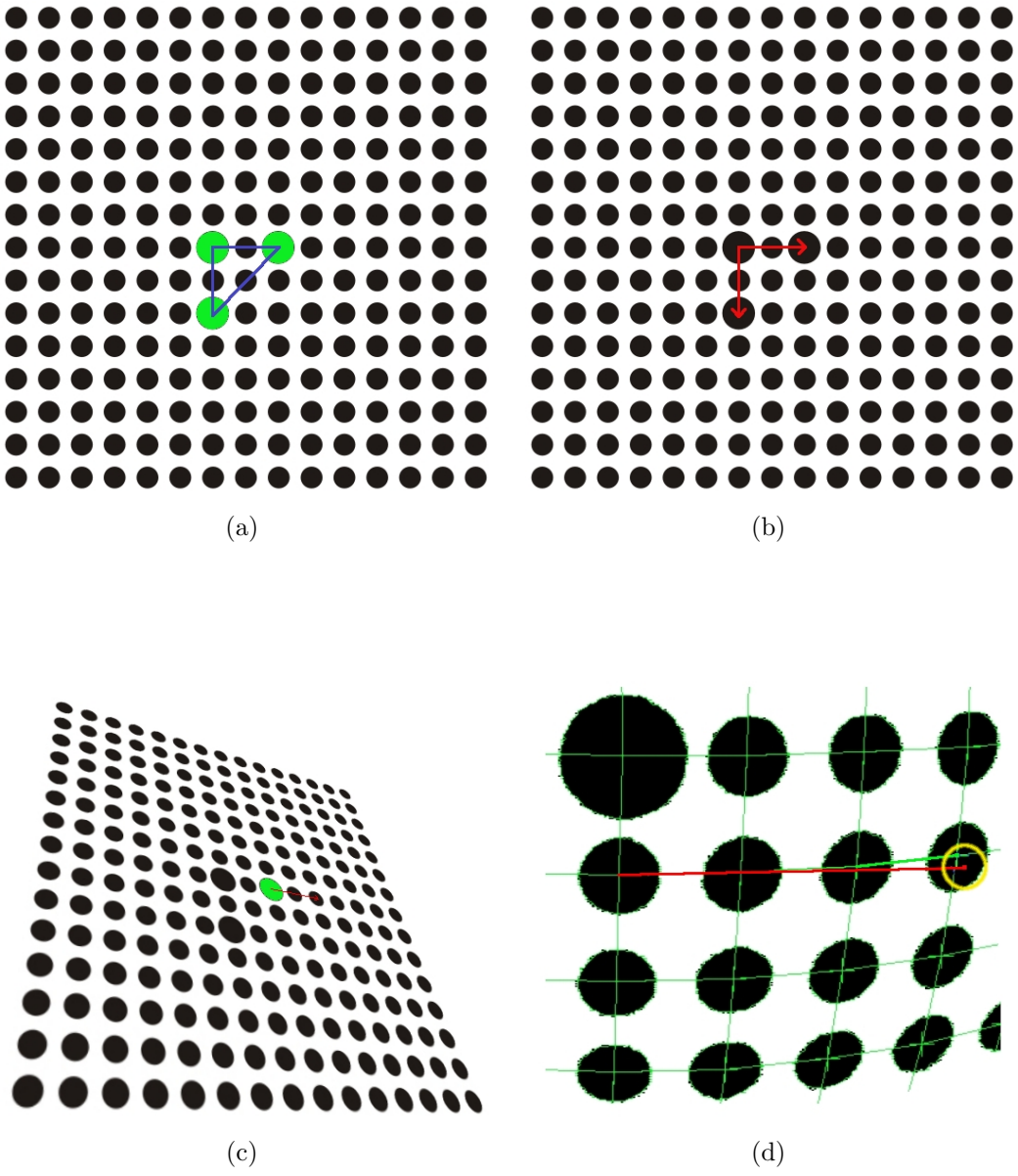
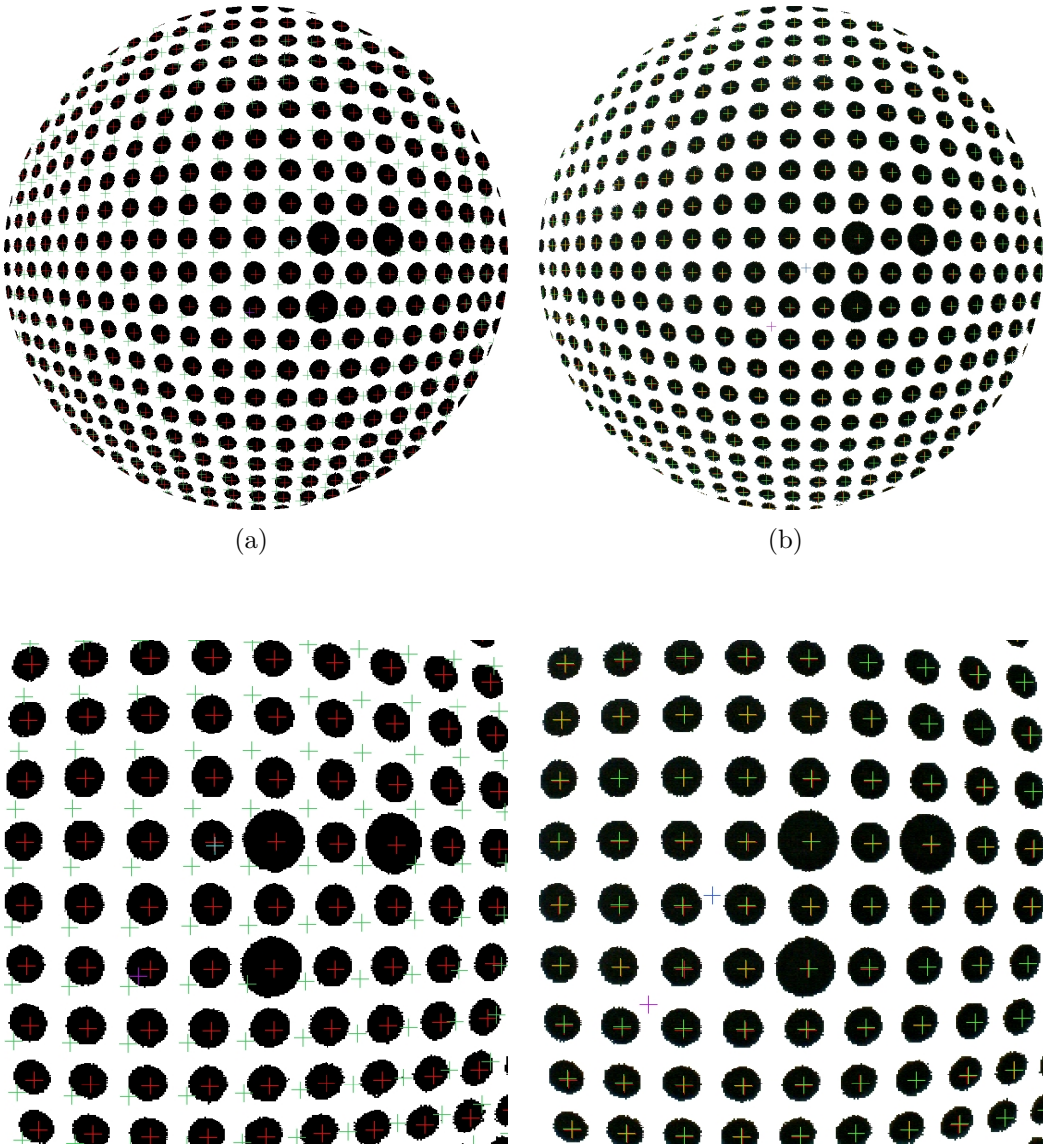


Figure 27: Image analysis: identification of the three big blobs (a); definition of a 3D frame (b); circle search in case of low distortion (c); search in case of high distortion (d)



(a) (b)

(c) (d)

Figure 28: A direct comparison of Hoppe's algorithm applied to fish-eye lenses vs. its modified version. The original, non converging algorithm (a); the modified algorithm with distortion center computed according to Asari (b)

	optical centre (mm)	principal point (pixel)
n	$\mathbf{z}$	(n_0, m_0)
1	(-0.89, 5.00, -48.92)	(567.85, 520.60)
2	(-0.78, 4.64, -41.20)	(568.12, 519.40)
3	(-0.80, 4.66, -41.14)	(568.42, 519.38)
	projected pixel n	projected pixel m
n	$\mathbf{a}$	$\mathbf{b}$
1	(6.8e-2, 1.7e-4, -9.7e-4)	(+3.9e-5, 6.8e-2, 7.1e-3)
2	(6.5e-2, 1.7e-4, -1.4e-3)	(-1.0e-4, 6.5e-2, 1.5e-3)
3	(6.5e-2, 1.7e-4, -1.4e-3)	(-9.8e-5, 6.5e-2, 1.5e-3)

Table 1: Perspective parameters convergence: optical center Z, pixel vector projections on the image plane **a**, **b** and pixel coordinates of the principal point (n_0, m_0)

	Distortion parameters (mm)	Residuals: (avg, max)
n	$\mathbf{k}_i$	
1	(8.9e-3, -1.3e-3, 6.9e-5, -8.5e-7)	(9.31, 25.80)
2	(-1.1e-2, 1.5e-3, -5.8e-5, 9.7e-7)	(0.87, 4.04)
3	(-1.2e-2, 1.6e-3, -6.0e-5, 1.0e-6)	(0.86, 4.30)

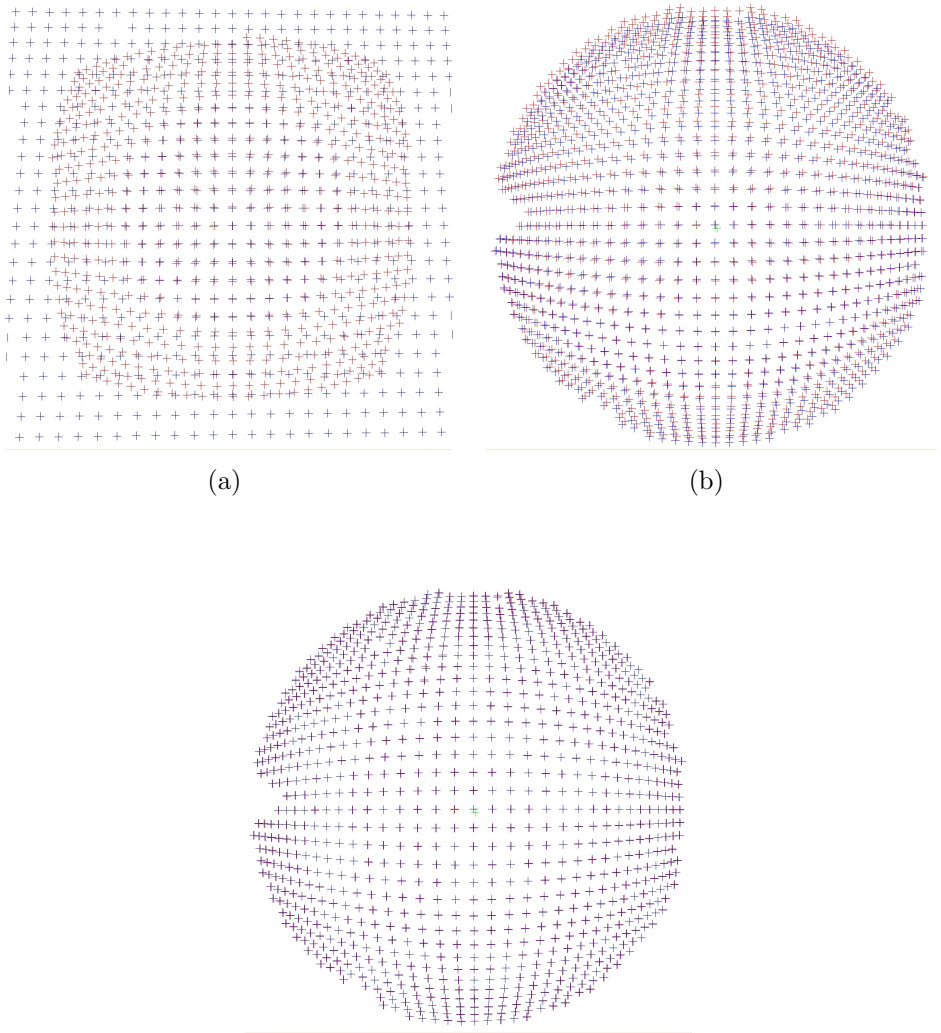
Table 2: Distortion parameters convergence: distortion coefficients $\mathbf{k}_i$

Convergence

The procedure of convergence described in section 3 was iterated to calibrate the endoscope until the residuals between measured pixels positions and corresponding computed positions are minimized. Experimentally it was found that the convergence was reached already after three steps; its results for every step are shown in fig. 29. Since the final solution achieved after the convergence was found depending on the first guess of the pinhole position and since the original algorithm [80] was found reliable on undistorted cameras, the area of interest used for estimating the first-order pinhole position was limited to the central region of the image and was let grown with every iteration step until the whole image area was covered. Table 1 shows the fast convergence achieved within this model.

6.2.1 Analysis of the results

The results of table 6.2 show some properties of the iteration method used. First of all, it can be easily noted that the first guess in the position of the optical center given by the **z** coordinates is usually greater than its final value. This is a property which depends on the way the model deals with the radial distortion during the first step of its convergence phase. Since the non-linear optical distortion of the lens is on the first step completely neglected, on high distorted lenses like fish-eye ones the number of detected circle centers coming from the exterior image



(a) (b)

(c)

Figure 29: The three steps of the algorithm convergence. Detected pixels are shown in red and reprojected pixels in blue. First algorithm iteration with distortion neglected (a); the second iteration with first distortion correction (b); the third and last iteration (c)

regions is N^2 denser than the same number coming from the inner image region and drives therefore the algorithm behaviour on the initial phase. The first guess for the perspective parameters indeed approximates a regular grid with a high distorted grid coming mostly from the external part of the image. As a result of this approximation, since the real inter-center distances are smaller because of the lens distortion but no parameters for any lens distortion are available during the first stage, the center to center distance is approximated as a pure perspective effect and results in a bigger distance as a first guess for the algorithm iteration.

The original solution proposed in [80] assumed that the distortion center had in the image plane the same position as the principal point at each iteration step or, more generally, that its position could have been computed by comparing the measured distorted pixel positions against an ideal perspectively distorted pattern as observed from a distortion-free computed optical center. This choice, performed at every iteration step on the basis of the iterative values of the calibration parameters, is not enough to let the algorithm converge towards its final solution. High distortions radial effects of fisheye lenses can however be included into the iterative and perturbative procedure if the final position of the distortion center on the image plane is computed *before* in a decoupled way by using the general properties of the control lines of the calibration pattern.

6.2.2 Conclusion

The camera calibration model developed in [80] has been tested on high distorted fish-eye lenses and improved with a decoupled guess of the distortion center (section 6.1.3). A change in the computation of the radial distortion center allows to extend a perturbative solution of the model to high distortion fisheye lenses. Satisfying results and a fast convergence have been obtained.

Chapter 7

Stereo reconstruction

This chapter describes how to perform sparse stereo reconstruction of tracked features. The first camera frame is measured respect to the calibration pattern and then navigated using an external tracking system. Corresponding object features are tracked using optical flow techniques and triangulated by crossing optical rays.

7.1 Mathematical introduction

This section introduces some mathematical tools required in this chapter: quaternions and dual quaternions, frames, feature selection and tracking methods and opical flow techniques.

7.1.1 Quaternions

Quaternions are members of a non-commutative field which extends the complex numbers. Its elements q are quadruples of $\mathbb{R}^4$ like $q = a + bi + cj + dk$ and 1 is the group identity. The product of two quaternions is determined by the products of the basis elements $(1, \imath, \jmath, \kappa)$ and the distributive law according to the basis elements' algebra:

$$\imath^2 = \jmath^2 = \kappa^2 = \imath\jmath\kappa = -1$$

Quaternions can be decomposed into scalar and vector parts $(s, \mathbf{q})$, where $s \in \mathbb{R}, \mathbf{q} \in \mathbb{R}^3$. In this case they satisfy the following operations:

$$q_1 + q_2 = (s_1 + s_2, \mathbf{q}_1 + \mathbf{q}_2)$$
$$\lambda q = (\lambda s, \lambda \mathbf{q})$$

The conjugate quaternion $\bar{q}$ is defined as $(s, -\mathbf{q})$. The multiplication between two quaternions $(s_1, \mathbf{q}_1)$, $(s_2, \mathbf{q}_2)$ is defined as:

$$q_1 q_2 = (s_1 s_2 - \mathbf{q}_1 \mathbf{q}_1, s_1 \mathbf{q}_1 + s_2 \mathbf{q}_2 + \mathbf{q}_1 \times \mathbf{q}_2)$$

Quaternions and rotations

There exists a map between the rotation $\mathbf{R}_\theta$ of an angle θ around the axis $\mathbf{n}$ and the quaternion $q = (\cos(\theta/2), \sin(\theta/2)\mathbf{n})$ which allows to represent rotations as quaternions. A point $\mathbf{x} \in \mathbb{R}^3$ represented as a pure vector quaternion $(0, \mathbf{x})$ transforms under such a rotation as:

$$q' = q(0, \mathbf{x})\bar{q}$$

7.1.2 Dual numbers

Let $a, b \in \mathbb{R}$. A dual number $\check{z}$ is a couple defined as:

$$\check{z} = a + \epsilon b, \quad \text{with } \epsilon^2 = 0$$

Using the same rules of addition and scalar multiplication, dual vectors $\check{v} = \mathbf{v}_1 + \epsilon \mathbf{v}_2$ can be easily defined. Dual vectors with orthogonal real and dual parts are representation of lines in $\mathbb{R}^3$ known as Plücker lines.

7.1.3 Dual Quaternions

Dual quaternions are defined as couples of dual numbers and dual vectors $(\check{s}, \check{v})$ which satisfy the quaternions operations:

$$\begin{aligned} \check{q}_1 + \check{q}_2 &= (\check{s}_1 + \check{s}_2, \check{\mathbf{q}}_1 + \check{\mathbf{q}}_2) \\ \lambda \check{q} &= (\lambda \check{s}, \lambda \check{\mathbf{q}}) \\ \check{q}_1 \check{q}_2 &= (\check{s}_1 \check{s}_2 - \check{\mathbf{q}}_1 \check{\mathbf{q}}_1, \check{s}_1 \check{\mathbf{q}}_1 + \check{s}_2 \check{\mathbf{q}}_2 + \check{\mathbf{q}}_1 \times \check{\mathbf{q}}_2) \end{aligned}$$

The dual quaternion $\bar{\check{q}}$ is the conjugate dual quaternion $\check{q}$. Dual vectors $\check{\mathbf{q}}$ can be written as dual quaternions $(0, \check{\mathbf{q}})$ and the multiplication of two dual vectors satisfies:

$$(0, \check{\mathbf{q}}_1)(0, \check{\mathbf{q}}_2) = (-\check{\mathbf{q}}_1^T \check{\mathbf{q}}_2, \check{\mathbf{q}}_1 \times \check{\mathbf{q}}_2)$$

Dual quaternions and rotations

There exists a map between the rototranslation $[\mathbf{R}_\theta, \mathbf{t}]$, where $\mathbf{R}_\theta$ is the rotation of an angle θ around the axis $\mathbf{n}$ and $\mathbf{t}$ is the translation vector and the dual quaternion $\check{q} = q + \epsilon q'$, where q is the rotation quaternion and $q' = \frac{1}{2}(0, \mathbf{t})q$. A line $\mathbf{l}_a$ in the space $\mathbb{R}^3$ passing through a point $\mathbf{p}$ with direction vector $\mathbf{l}$, represented as a dual vector quaternion $\check{l}_a = (0, \mathbf{l}) + \epsilon(0, \mathbf{p} \times \mathbf{l})$, transforms under a rototranslation $[\mathbf{R}_\theta, \mathbf{t}]$ as:

$$\check{l}'_b = \check{q}\check{l}_a\bar{\check{q}}$$

The transformation of a line under a rototranslation mimics the transformation of a point under simple rotation.

7.2 Pose tracking: problem statement

Augmented reality relies upon camera pose tracking which allows camera poses to be tracked in 3D space. The transformation from the tracking system to the camera frame can be decomposed in two steps: a transformation from the tracking system to the endoscope $\mathbf{T}_e^t(t)$ which depends on the endoscope position at time t and a constant transformation from the endoscope system to the camera $\mathbf{T}_c^e$. However, even if the first transformation is known, the second has to be regarded as an unknown, since the camera frame can be only expressed respect to an external calibration pattern as $\mathbf{T}_c^w$. The problem of the camera pose tracking becomes therefore the problem of finding a way of measuring the unknown $\mathbf{T}_c^e$ given $\mathbf{T}_e^t$ and $\mathbf{T}_c^w$ (fig. 30).

7.2.1 Endoscope tracking

Navigation relies upon tracking systems which allow objects to be tracked in 3D space. Tracking systems use special rigid bodies composed of three or more standard retroreflective spheres and known geometry which are fixed onto an object to allow its tracking. Their position and orientation can be easily computed using two or multiple cameras in a fixed position. In case of the endoscope, the transformation from the tracking system frame to the endoscope frame is $\mathbf{T}_e^t$ (fig. 31).

7.2.2 Camera pose estimation

The initial camera frame is measured respect to the world frame and defined by a known camera calibration pattern. The camera pose is then obtained at the end of the camera calibration procedure from the camera extrinsic parameters. In case

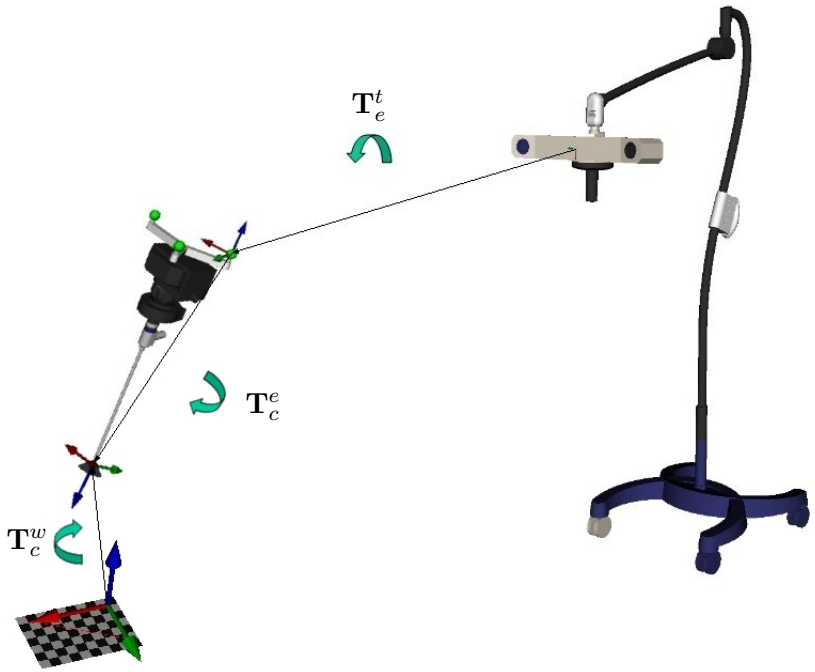


Figure 30: Camera tracking $\mathbf{T}_c^t = \mathbf{T}_e^t \mathbf{T}_c^e$ transforming tracking system coordinates $\mathbf{x}_t$ into camera coordinates $\mathbf{x}_c$

of the camera pose, the transformation from the calibration pattern frame to the camera frame is noted as $\mathbf{T}_c^w$ (fig. 32).

Camera optical center

If external parameters are represented as $[\mathbf{R}, \mathbf{t}]$, the origin of the camera optical center in world coordinates is given by:

$$\mathbf{x}_w = -\mathbf{R}^{-1}\mathbf{t} \tag{44}$$

In a real system, there exist two axes of symmetry, one optical and one mechanical. The camera optical center describes the ideal camera pinhole position as an optical and not a mechanical property. As a result of a slighly misalignment of parts in fact, its position cannot exactly align with the mechanical symmetry axis. The camera optical center however must transform rigidly with the camera frame as the camera is moved throughout the space.

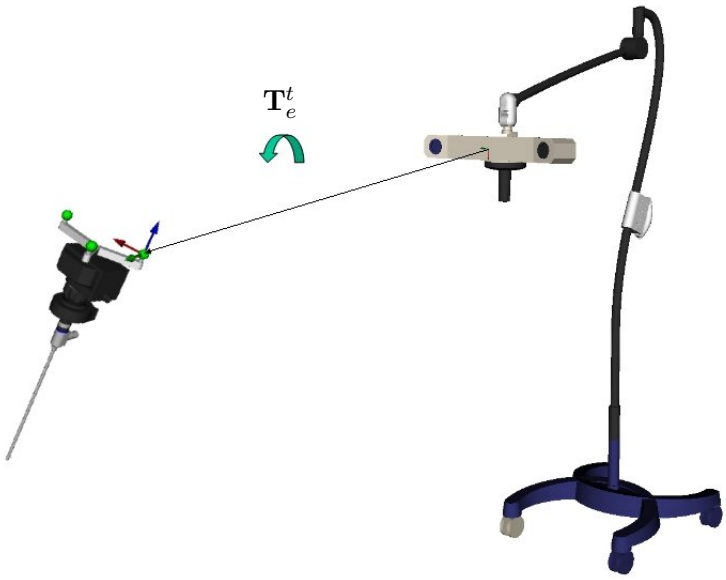


Figure 31: Endoscope tracking $\mathbf{T}_e^t$ transforming tracking system point coordinates $\mathbf{x}_t$ into endoscope coordinates $\mathbf{x}_e$

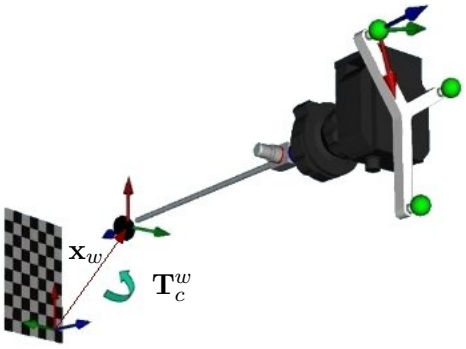


Figure 32: Camera pose estimation $\mathbf{T}_c^w$ transforming world point coordinates $\mathbf{x}_w$ into camera coordinates $\mathbf{x}_c$

7.3 Pose tracking: endoscope to camera

Using the tracking system to navigate the camera frame requires to measure the endoscope to camera transformation.

7.3.1 Direct measure

The direct computation of the relative position and orientation between a tracked endoscope and a camera mounted rigidly on it can be performed by introducing a third frame to switch from endoscope to camera frames. The fact that the common frame must be tracked respect to the tracking system and to the camera suggest the optically tracked calibration frame as the suitable one.

Let be $\mathbf{X}_e^c$ the unknown camera to endoscope transformation, $\mathbf{T}_c^w$ the transformation matrix from the camera to the world coordinate system, $\mathbf{T}_e^t$ the transformation form the tracking system to the endoscope and $\mathbf{T}_w^t$ the transformation form the tracking system to the calibration pattern (fig. 33) . The obvious relation between the transformation matrices can be used to compute the unknown $\mathbf{X}$ transformation:

$$\mathbf{X} = \mathbf{T}_c^w \mathbf{T}_w^t (\mathbf{T}_e^t)^{-1} = \mathbf{T}_c^w \mathbf{T}_w^t \mathbf{T}_t^e$$

The direct computation requires however the introduction of the calibration pattern navigated frame as a possible additional source of error.

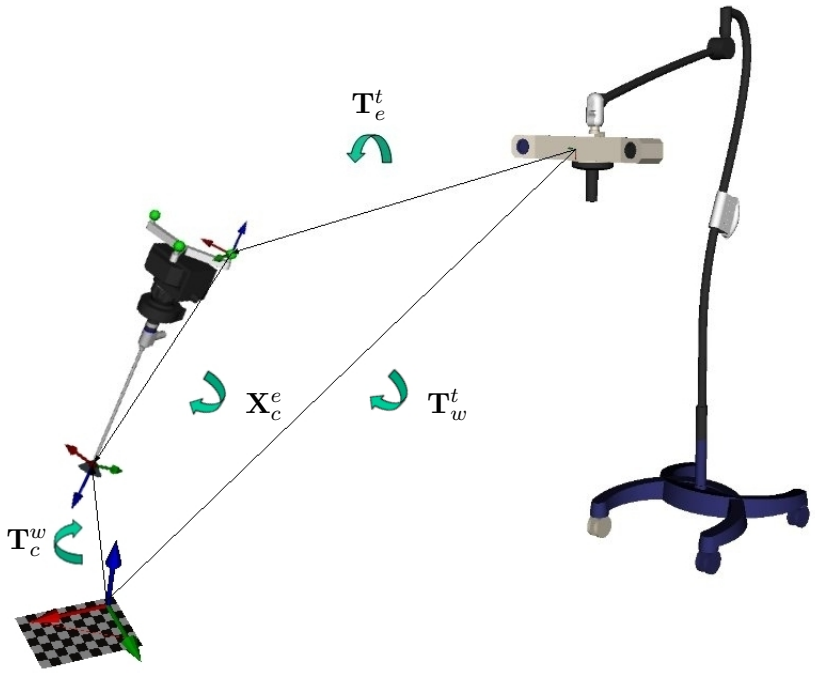


Figure 33: Direct measure of the endoscope to camera transformation matrix $\mathbf{X}$

7.3.2 Hand-eye calibration

Hand-eye calibration is the computation of the relative position and orientation between a tracked endoscope and a camera mounted rigidly on it [83] without introducing any common third frame. Let be $\mathbf{X}$ the unknown camera to endoscope transformation, $\mathbf{A}_i$ the transformation matrix from the camera to the world coordinate system and $\mathbf{B}_i$ the transformation form the tracking system to the endoscope at the i^{th} pose.

The camera to world transformation is obtained by measuring the extrinsic camera parameters using a calibration pattern as a reference for each pose; the tracking system to endoscope transformation is obtained directly from the tracking system as a result of tracking the endoscope position. For each pose there are two unknown transformations: tracking system to world system and camera to endoscope. Considering relative movements the hand-eye equation can be written:

$$\mathbf{AX} = \mathbf{XB} \tag{45}$$

where $\mathbf{A} = \mathbf{A}_2\mathbf{A}_2^{-1}$ and $\mathbf{B} = \mathbf{B}_2\mathbf{B}_2^{-1}$. The problem statement of recovering the unknown transformation $\mathbf{X}$ depends therefore on variations of transformations $\mathbf{A}, \mathbf{B}$ only. In homogeneous matrix representation, where $\mathbf{X} = (\mathbf{R}_X \ \mathbf{t}_X)$, the hand-eye equation is:

$$\mathbf{R}_A\mathbf{R}_X = \mathbf{R}_X\mathbf{R}_B \tag{46}$$

$$(\mathbf{R}_A - \mathbf{I})\mathbf{t}_X = \mathbf{R}_X\mathbf{t}_B - \mathbf{t}_A \tag{47}$$

This problem statement avoids any explicit introduction of a third switch frame and therefore any possible additional independent source of error.

Two motions containing non parallel rotation axis are required at least to solve the problem. Most approaches decompose the matrix $\mathbf{X}$ into its rotational and translational parts and optimize first the rotation and then the translation: [84] is the classic paper on the topic. The same approach is kept in [85], where quaternions are used to represent the rotation; in [86] the euclidean group is used.

Dual quaternions approach

A different approach is used [83] where, using the representation of dual quaternions and the screw theory, the rotation and translation of $\mathbf{X}$ are determined simultaneously.

7.4 Tracking Features

Features are interesting parts of an image used as a starting point for many computer vision algorithms. Their exact definition depends on the kind of problem wanted to be solved and there is therefore no universal definition for a feature: corners, edges, blobs, ridges are all possible points or regions of interest in an image. Features are then selected based on some measure of texturedness or cornerness, such as a high standard deviation in the intensity profile or the presence of zeros of a suitable operator applied to the image like, for instance, the Laplacian. Feature detection is the act of identifying the features in an image: it's a low-level image processing operation performed on every pixel to determine if a selected feature is present. Features are however instrinsically 2D objects and even a region rich in texture can produce poor features: a reflection on a glossy surface would certainly be a good feature respect to its intensity properties, but since it wouldn't be attached to any fixed point in the world, would be a useless or even harmful choice for tracking purposes. Good features can also become occluded and the quality of a feature should be tested continuously during its tracking.

7.4.1 Shi-Tomasi feature definition

In [87] J. Shy and C. Tomasi define the dissimilarity to quantify the change of appeareance of a feature between the first and the current frame. Dissimilarity is defined as the feature's rms residue between the first and the current frame. Given two images I and J, if a point $\mathbf{x}$ in first image moves to the point $\mathbf{x} + \delta$ in the second image; it can be said that:

$$J(\mathbf{Ax} + \mathbf{d}) = I(\mathbf{x})$$

where $\mathbf{A} = \mathbf{D} + \mathbf{1}$, $\mathbf{D}$ is the deformation matrix, $\mathbf{1}$ is the 2×2 unit matrix and image position $\mathbf{x}$ is measured respect to window's center. The displacement at point $\mathbf{x}$ is $\delta = \mathbf{Dx} + \mathbf{d}$, where $\mathbf{d}$ is the translation of the window's center.

The quality of a feature respect to the first frame is measured with its dissimilarity respect to the initial selected feature:

$$\epsilon = \int_W \left[J(\mathbf{Ax} + \mathbf{d}) - I(\mathbf{x})\right]^2 w(\mathbf{x}) d\mathbf{x}$$

where W is the given feature window, $w(\mathbf{x})$ is a weighting function and the full affine displament A is used. The equation is linearized with a truncated Taylor expansion:

$$J(\mathbf{Ax} + \mathbf{d}) = J(\mathbf{x}) + \mathbf{g}^T(\mathbf{u})$$

and the previous equation is restated as a 6×6 linear system:

$$T\mathbf{z} = \mathbf{a} \tag{48}$$

where $\mathbf{z}^T = [d_{xx}\ d_{yx}\ d_{xy}\ d_{yy}\ d_x\ d_y]$ are entries of the deformation D and displacement $\mathbf{d}$, the vector:

$$\mathbf{a} = \int_W [I(\mathbf{x}) - J(\mathbf{x})] \begin{pmatrix} xg_x \\ xg_y \\ yg_x \\ yg_y \\ g_x \\ g_y \end{pmatrix} w(\mathbf{x}) d\mathbf{x}$$

depends on the difference between the two images and the matrix $\mathbf{T}$ and can be computed from one image as:

$$\mathbf{T} = \int_W \begin{pmatrix} \mathbf{U} & \mathbf{V} \\ \mathbf{V}^T & \mathbf{Z} \end{pmatrix} w(\mathbf{x}) d\mathbf{x}$$

where:

$$\mathbf{U} = \begin{pmatrix} x^2\mathbf{Z} & xy\mathbf{Z} \\ yx\mathbf{Z} & y^2\mathbf{Z} \end{pmatrix}\ ,\ \mathbf{V} = \begin{pmatrix} x\mathbf{Z} & y\mathbf{Z} \end{pmatrix}\ ,\ \mathbf{Z} = \begin{pmatrix} g_x^2 & g_x g_y \\ g_x g_y & g_y^2 \end{pmatrix}$$

A Newton-Raphson minimization criterium is used on equation (48) to overcome the linear approximation. The same equation (48) is then used at three different moments: at first, on the selection of a particular feature in a window W depending on the eigenvalues λ_1, λ_2 of $\mathbf{Z}$. Since low values identify uniform regions and high values corner-like regions, according to the Shi-Tomasi algorithm a robust feature must satisfy the condition:

$$\min(\lambda_1, \lambda_2) > \lambda \tag{49}$$

with λ is a predefinite threshold.

Tracking means, for a given window W, determining the six parameters of $\mathbf{A}$ and $\mathbf{d}$: since the variation between similar frames is small, a pure translation model with $\delta = \mathbf{d}$ and small windows will be preferable. Equation (48) holds therefore a second time during the tracking between a frame and the successive, with $\mathbf{D} = \mathbf{0}$ to guarantee pure translation.
Monitoring for dissimilarities between the current and the initial frame still uses the equation (48) with the full affine deformation $\mathbf{D}$. The monitoring criterium (49) selects when the dissimilarity has grown too much and a feature has to be discarded.

7.5 Optical flow

Optical flow is the pattern of apparent motion of objects in a visual scene caused by the relative motion between an observer and the scene [88]. The majority of optical flow methods are differential: they use partial derivatives to compute the motion between two image frames. Common to all optical flow methods are the brightness constancy assumption, which leads to the image constraint equation:

$$I(y, y, t) = I(x + \delta x, y + \delta y, t + \delta t)$$

Assuming a small motion and defining the velocity field or optical flow of $I(x, y, t)$ as $\mathbf{v}$ and indicating partial derivatives of intensity as $(\partial_x I, \partial_y I, \partial_t I)$, allows to recast the previous equation as:

$$\nabla I^T \cdot \mathbf{v} = -\partial_t I \tag{50}$$

The fact that equation has two unknowns and cannot be solved without further assumptions is known as the aperture problem in computer vision and the introduction of additional contraints leads to various optical flow methods. Phase correlation methods derive the motion between two images by computing the inverse of the normalized cross power spectrum; block based methods minimize their sum of absolute differences; the Horn-Schunk method optimizes a functional based on the residuals from the brightness constancy constraint and a particular regularization term expressing the expected smoothness of the flow field.

7.5.1 Lucas Tomasi Kanade

Lucas-Tomasi-Kanade (LTK) optical flow method [89, 90], which is the one we are intersted in this work, assumes the temporal persistence and the spatial coherence of image motion. The first condition requests the image motion of a surface patch to change slowly in time; the second condition assumes that neighbouring points in a scene belong to the same surface and have then a similar motion. The constraint (50) can be assumed to hold on all pixels of a $n \times n$ small window:

$$\begin{pmatrix} I_x(p_1) & I_y(p_1) \\ I_x(p_2) & I_y(p_2) \\ \vdots & \vdots \\ I_x(p_n) & I_y(p_n) \end{pmatrix} \begin{pmatrix} u \\ v \end{pmatrix} = - \begin{pmatrix} I_t(p_1) \\ I_t(p_2) \\ \vdots \\ I_t(p_n) \end{pmatrix} \tag{51}$$

Pyramidal Implementation of LTK

The weaker of Lucas-Tomasi-Kanade's hypothesis is the request of the spatial coherence: large image motion is commonly observed on images. To detect it, a large

window of interest should be used; however, on large windows the assumption of the spatial coherence for a uniform optical flow breaks down. A beatiful solution which unifies the detection of small and big image motions is offered by working at diffent image scales: at lower resolutions big image changes become small ones and large pixel jumps can be described in term of few coarse-grained pixel jumps. Small scale pixel motion is therefore detected at lower scale and then iteratively refined on higher resolution: previous results are used as first guesses for the next iteration step and the procedure is repeated until a certain degree of accuracy is obtained. The classical implementation of this multiscale procedure, which makes use of smaller and low resolution replicas of the original image, is called pyramidal implementation of the LTK optical flow.

7.6 Triangulation

In the reconstruction problem the aim is to compute the 3D coordinates of a point given two or more of its 2D views. The usual method to do it is using a triangulation of the detected optical rays drawn through the pixels. Since detected pixels contain noise, two optical rays won't cross at a single point exactely. Various triangulation methods which differ in the way they can be generalized and in their transformation properties have been developed [91, 92].

Middle point

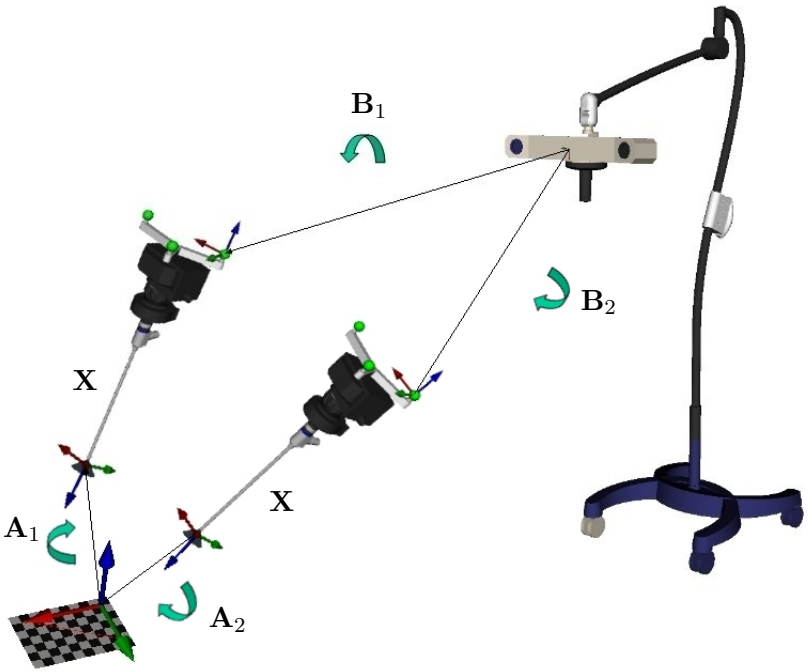
The middle point method assumes that the intersection between two skew lines lies on the middle point over the nearest distance between the two projected lines. The method is euclidean invariant but neither affine nor projective invariant.

The linear least square method

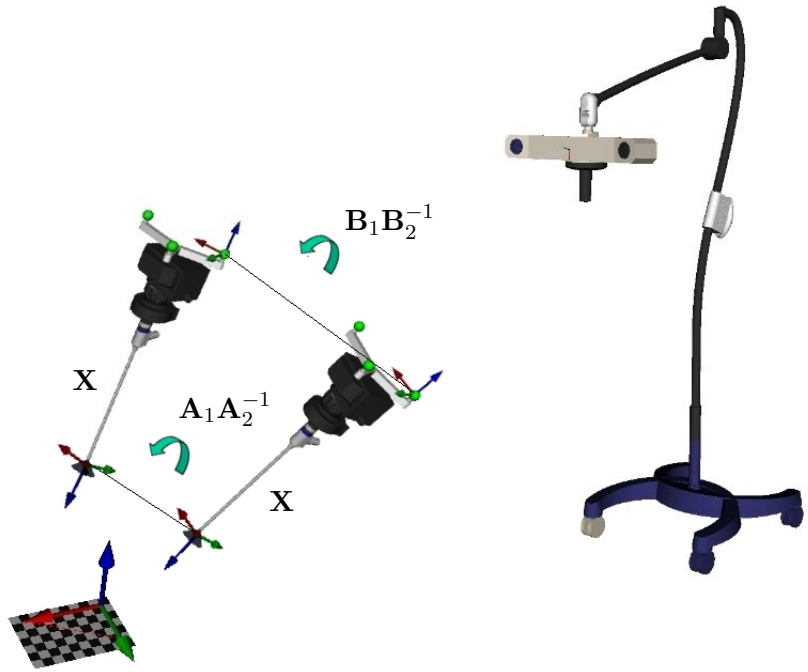
The linear least squared method solves the homogeneous linear equations obtained by stacking many pixel point projection equations. The method is euclidean and affine and euclidean invariant, but not projective.

The epipolar distance method

The epipolar distance method attemps to minimize the sum of the distances between the corresponding 2D points and their corresponding epipolar lines, computed by using the fundamental matrix. The method is eucledean, affine and projective invariant.



(a)



(b)

Figure 34: Camera pose estimation $\mathbf{T}_c^w$ transforming world point coordinates $\mathbf{x}_w$ into camera coordinates $\mathbf{x}_c$

Chapter 8

The NEAR project

In the following section the NEAR Project is presented as an exemplifying implementation of an active endoscope. The experimental setup and the methods used to build an active endoscope to perform 3D reconstruction are discussed with emphasis concerning standard and available OR devices.

8.1 Hardware

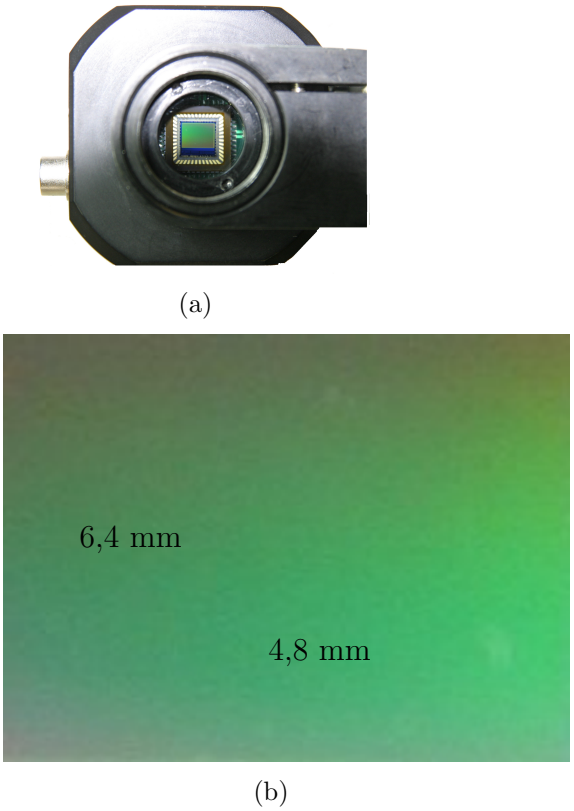
The setup of the NEAR system is composed by standard endoscopes, cameras and optical tracking systems. The devices selected for this implementation are:

- 1 Wolf Panoview Endoscope, 0°, 4 mm
- 1 Wolf Panoview Endoscope, distortion-free, 0°, 10 mm
- 1 Wolf C-mount adaptor
- 1 NDI Polaris Tracking System
- 1 Sumix M-72 USB color camera, 1600x1200 Mp, 48 fps

The standard setup makes use of the 4 mm 0 ° Panoview R. Wolf endoscope, while the 10.0 mm 0° distortion-free R. Wolf endoscope was mainly used to check the effects of a fisheye lens distortion on the overall system precision.

8.1.1 Camera

Cameras are classified depending on their image sensor. Cameras with CCD sensors have a better S/N ratio but are quite expensive; CMOS sensors are cheaper but they are less sensitive than CCD. All the other things being equal, larger sensors capture images with less noise and greater dynamic range than smaller sensors;



(a)

(b)

Figure 35: Sumix M72 CMOS camera (a); 1/2" image sensor format (b)

their signal-to-noise ratio (S/N) and sensor unity gain (SUG) both scale with the square root of sensor area.

The Sumix M72 USB color camera, with a maximum resolution of 1600x1200 at 48 fps, was used with a square active window of 1200x1200 pixels at 24 fps to fit the 1/2" image sensor (fig. 8.1.1). The camera and the endoscope were coupled through a R. Wolf, C-mount standard adaptor.

Bayer filter

A Bayer filter mosaic is a color filter array used to arrange RGB color filters on a square grid of photosensors. The M72 Sumix Bayer filter uses a permutation of the standard 2x2 RGGB unit structure, where twice as many green elements as red or blue are used to mimic the human eye's greater resolving power with green light. Alternatives to the Bayer filter are the CYGM filter (cyan, yellow, green, magenta), the RGBE filter (red, green, blue, emerald), the Foveon X3 sensor, which layers red, green and blue sensors vertically rather than using a mosaic or uses three separate sensors, one for each color. A Bayer raw image is restored using a usually in-camera demosaicing algorithm. The simplest demosaicing algorithm assigns to each interpolated output pixel the value of the nearest pixel in the raw input image

(algorithm 4).

Algorithm 4 Simple Bayer algorithm

```
typedef struct _rbgframe {
    unsigned char b,g,r;
} rgbframe;

void
bayer_filter(unsigned char *in, int w, int h, rgbframe *out)
{
    register int i, j;
    unsigned char *in1 = in;
    unsigned char *in2 = in + w;
    rgbFrame *out1 = out;
    rgbFrame *out2 = out + w;

    for (j = 0; j < h; j += 2, in1 += w, in2 += w, out1 += w, out2 += w)
    {
        for (i = 0; i < w; i += 2, out1 += 2, out2 += 2 )
        {
            unsigned char g00,b10,r01,g11;

            g00 = *in1++; r01 = *in1++;
            b10 = *in2++; g11 = *in2++;

            out1->g = g00; out1->r = r01; out1->b = b10;
            out1[1] = out1[0];
            out2->g = g11; out2->r = r01; out2->b = b10;
            out2[1] = out2[0];
        }
    }
}
```

C-Mount

Endoscopes are coupled to cameras using a C-mount adapter. The cinema C-mount standard defines the optical source side consisting of a tube concentric with the optical axis, ending in a 1”-32 male thread and the optical receiver consisting mechanically of a 1”-32 female thread with a detector and/or further optics at the image location. The light rays of the optical source form an image plane 0.69 inch or 17.526 mm away from this flange (fig. 37).

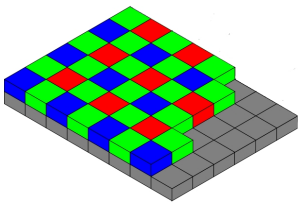
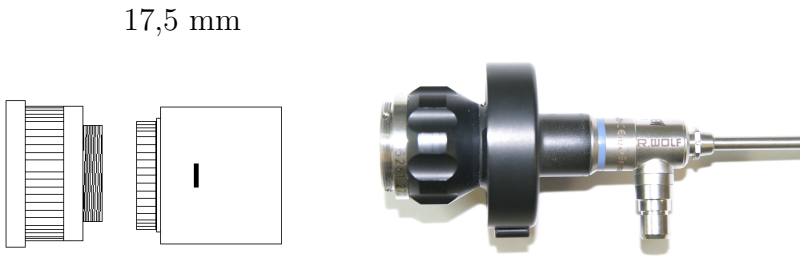


Figure 36: A drawing of the Sumix GRGB Bayer sensor



(a) (b)

Figure 37: Camera and C-Mount coupling (a); Endoscope and C-Mount coupling (b)

Image Resolution

Since the Sumix camera has square pixels $4,2\ \mu m$ wide, from the smaller linear dimension of the 1/2" image sensor its easy to compute the corresponding image resolution:

$$\text{Image resolution} = \frac{4.8\ mm}{4.2\ \mu m} = 1142 \text{ pixels}$$

In the application a 1120x1120 image resolution has been used as the bright spot of a white image was larger than the image sensor. The focal lenght was set at its calibration distance 10 cm by adjusting the focal of the C-mount.

Light Box

The light source is a R. Wolf halogen lamp with two light bulbs of 250 W each. To avoid any electrical shock of the workstation when turning on and off the light, two separate electrical line have been used.

8.2 Software

The NEAR application is a standalone application written in C++ which handles a camera and a tracking system and provides navigation, registration, augmented

(a) (b)

Figure 38: R. Wolf endoscope light box (a); its light fiber guide coupled to the endoscope (b)

reality and triangulation modules in a virtual environment. In the following section, the software architecture is described and discussed.

8.2.1 Libraries

The NEAR application is built on a number of open source libraries which provide the main functionalities (fig. 40):

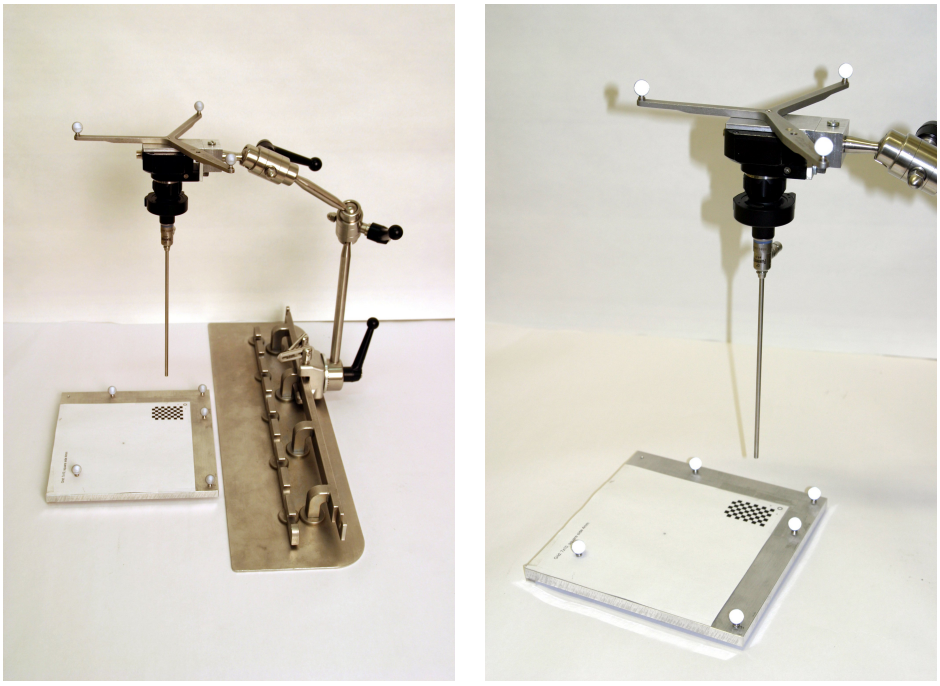
Qt Open source library used for the graphic user interface, Qt is a cross-platform application and UI framework released under commercial, GNU LGPL and GPL licence. It has a powerful mechanism for inter-object communication called signals and slots. It is used mainly to implement the graphic user interface.

VTK Open source library used for the virtual environment, VTK [93] is an open-source freely available visualization toolkit for 3D computer graphics, image processing and visualization. VTK consists of a C++ class library and several interpreted interface layers including Tcl/Tk, Java, and Python, released under BSD license. It is used mainly to implement the virtual environment.

OpenCV Open source library used for performing camera calibration and image processing, OpenCV [94] is the Intel's Open Computer Vision library; it offers real time computer vision algorithms and methods and is released under BSD license [94]. It is mainly used for image processing.

CLAPACK Open source library, freely-available, copyrighted but not trademarked software, the CLAPACK library is a C version of the Linear Algebra Package LAPACK, converted from Fortran to C with *f2c* and then modified to improve its readability [95]. It is used mainly used to solve linear systems.

Numerical Recipes Mathematical licenced library which implements most of the routines from its correspondent book. It should be replaced with the open



(a) (b)

Figure 39: NEAR system setup (a); passive markers used to navigate the endoscope and the calibration pattern (b)

source GSL, the GNU Scientific Library. It is mainly used to compute eigenvalues and eigenvectors and for singular value decomposition (SVD).

MathBase Local library providing a C++ interface for vectors, matrix, quaternions and their algebra including the principal convertions between various rotation representations. The class in particular introduces the C++ classes `CVecd`, `CMatd` to represent vectors and matrices.

MathToolBox Local library providing a wrapper between the MathBase C++ classes and the powerful C based CLAPACK and Numerical Recipes algorithms.

AVL tiny local library, which defines the balanced trees and is used by the MathBase library only. It could be in future removed from the set of the libraries used.

TrackerBase Local library providing IO control for NDI optical tracking systems. It provides an own implementation of the communication protocol between an NDI, Polaris and standard RS232 serial port.

SMXM72 Local library providing IO low-level control for the sumix M7X family of cameras. It replaces the Microsoft DirectShow OpenCV framework, not properly supported by the original sumix driver.

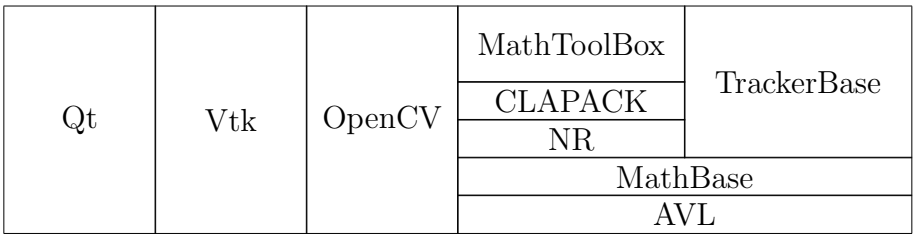


Figure 40: NEAR library interdependence: Qt, Vtk, OpenCV and CLAPACK are the fundamental libraries used in the NEAR application

8.3 The software architecture

The following section describes the software architecture of the NEAR application. To explain the choices taken in designing the application in the clearest way , it will surely help the reader to summarize in the following paragraphs the key-points of the assumed scenario. In its very basic form providing only navigation, the software application must control two harwdware devices: an optical tracking system and a camera. The tracking system computes the position and the orientation of the navigated surgical tools and patient returning them as homogeneous matrices or frames. The endoscope camera simply reads the endoscope video to the surgeon. This skeleton shape for the application together with the choice of the Qt and VTK libraries, already constraints very much the software architecture. Since Qt owns the main process thread to monitor and control the GUI, the easiest and better documented way for doing some other kind of processing while monitoring the user interface in a non-blocking way is using the Qt thread support provided by the `QThread` class. Another implementation might use the `QTimer` class; the disadvantage of this choice would be a freezed user interface during the image and frame reading and processing. Moreover, two `QWidget`s must provide the visualization of the endoscopic video and virtual scene.

A `QWidget` embedding VTK: the `QVTKWidget`

Another feature offered by the Qt library is the `QVTKWidget`, which embeds in a single widget a complete VTK virtual scene. Since both Qt and VTK are event-driven environment which try to get full control of the `main()` thread, QThreads must be used instead of system threads or VTK threads.

8.3.1 I/O and data flow: images and frames

The actual implementation of the NEAR application uses therefore two independent `Qthread`s for I/O: a first thread is used to read the positions of the various

surgical instruments from the tracking system; a second thread is used instead to read the video image from the endoscope camera. Generalizing, for each external device in the Qt application, there is a dedicated `Qthread` responsible for it. Now let's describe how the information read (image or frames) reaches the corresponding `QWidget` where it's processed. In this case inspired by the Qt Mandelbrot example, the data communication between C++ classes chosen explicitly exploits a defining feature of the Qt library: the signal-slot mechanism. When a new image or matrix is read, each thread emits a Qt signal which is connected via the Qt signal-slot mechanism to the corresponding camera or virtual scene `QWidget`, where the rendering happens. The internal event queue of Qt is used to update respectively the 2D pixmap in the camera widget or the 3D virtual scene which for the moment contains the actor positions only. The user interface reflects at the presentation level the same choice: two top-level widgets are used to paint respectively the original 2D endoscopic video image and the virtual environment allowing surgical navigation (fig. 42).

Augmented reality

The augmented reality (AR) is obviously, in this work, the crucial part of the application. Since AR can be obtained by overlaying a real image and a corresponding virtual view, the problem splits in two parts: how to compute and track a virtual view which corresponds to the endoscope camera view and how to implement it. To provide to the reader a comprehensive understanding of the software implementation, the remaining part of this paragraph focuses on this last part, while the next paragraph will deal with the issue of computing a proper VTK virtual camera from the parameters of the camera calibration models used in this work.

The overlaying of two images can be obviously done in two different ways. At the image level overlaying the real image onto the virtual view (or, equivalently, by overlaying the virtual view onto the real image) using the transparency properties of 2D images with transparency coefficients computed from the z-buffer of the virtual scene. A second possibility if to work at virtual scene level, introducing in the virtual scene a plane actor used as a plain screen. Of course, its position and orientation will be such that, when observed from the virtual camera, its embedded video will look like the 2D camera video.

In the actual NEAR implementation the second solution presented has been preferred to the first one. AR is obtained therefore by rendering a video onto a semi-transparent `vtkPlaneActor` class as a texture, by setting a `vtkCamera` at a position computed first in the camera frame and by navigating it using the endoscope positional information from the tracking system. Once the endoscopic camera is calibrated, its vector parameters like the optical centre, the view-up vector and the focal points can be easily expressed into the moving endoscope frame and transformed consistently during its navigation. The endoscope VTK actor and its camera vector parameters are updated every time a new matrix describing a new endoscope

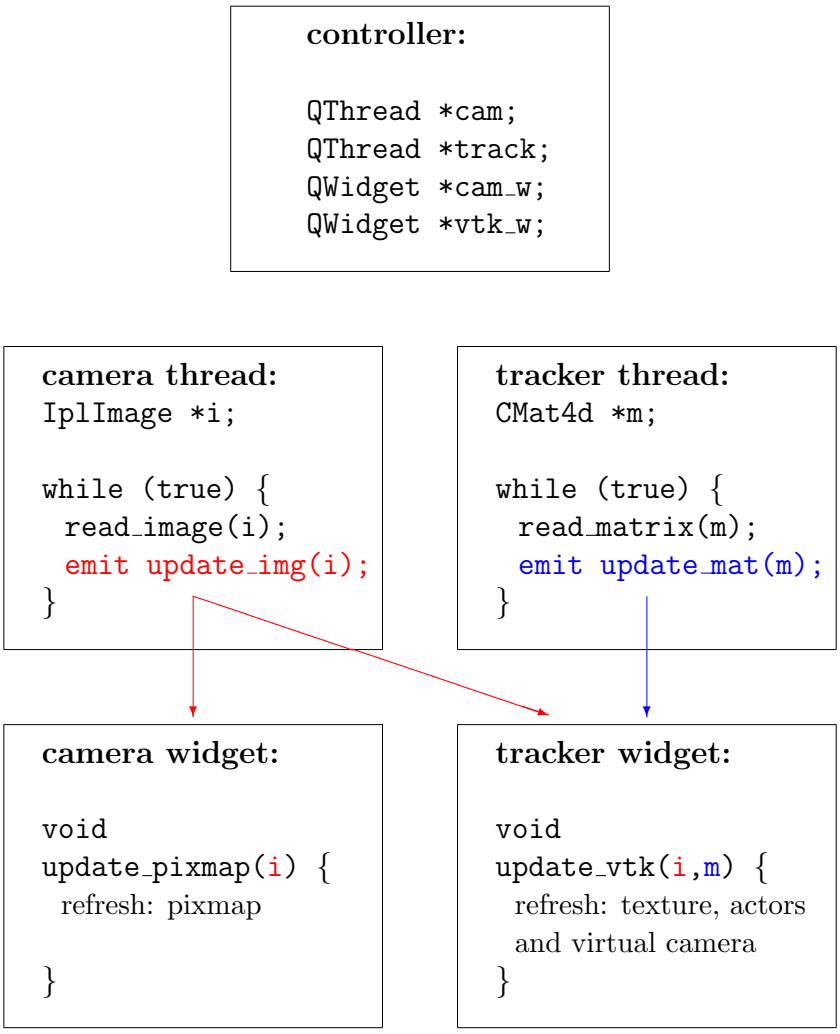


Figure 41: NEAR architecture: two Qt threads read respectively the image and the positions of the tools. A Qt signal is sent to refresh the image and the virtual scene

position is available. Accordingly, every time a new image is read from the camera, the texture on the semitransparent plane must be updated as well. (fig. 41).

8.4 The virtual camera from the calibration matrix

In computer vision, a virtual camera is a nothing but a view point on a virtual scene: typically the user interacts with it using a mouse or a keyboard. A virtual camera in VTK, for example, is completely defined by its position, view-up vector, focal point and view angle. To represent a real camera with a virtual view though, one must first solve the problem of computing the virtual camera parameters from the camera

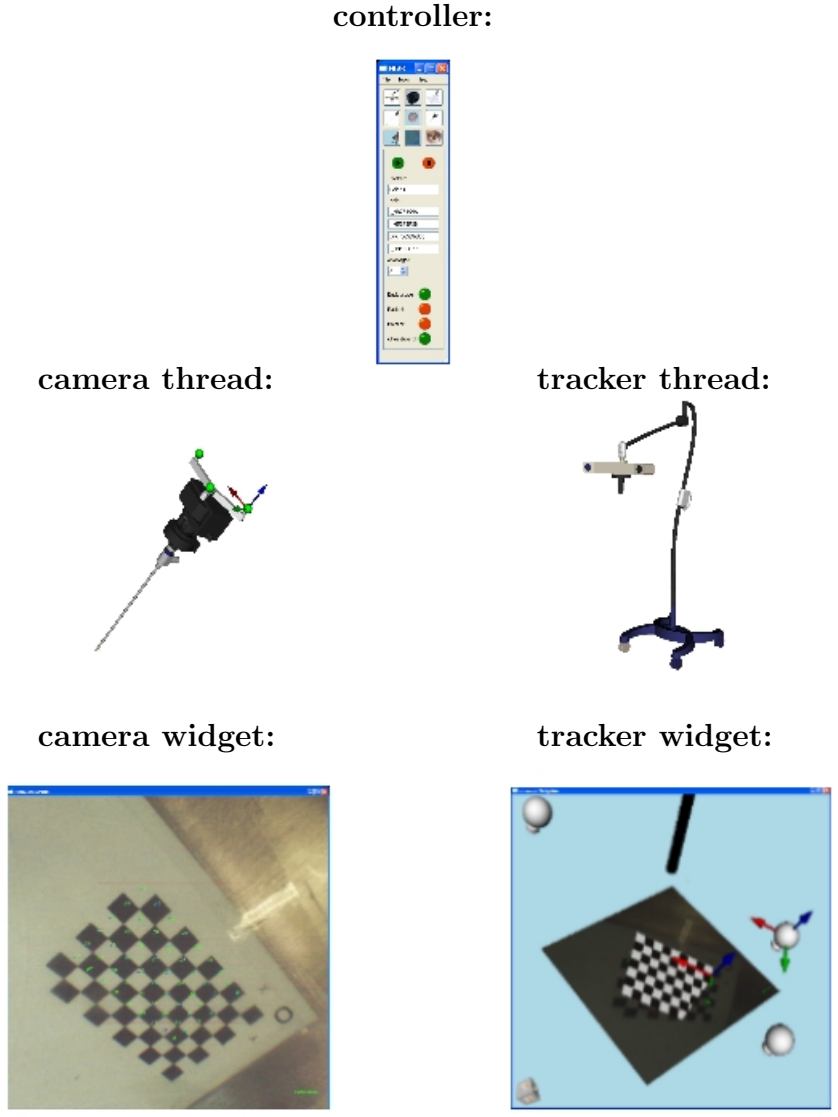


Figure 42: NEAR GUI architecture: a controller uses two Qt threads to handle the camera and tracking system devices. Their information is rendered into two QWidgets

calibration parameters. These parameters can be then consistently tranformed in the moving frame of the camera actor, or, in the case of this work, in the frame of the endoscope actor and used to keep the correspondence between the real and the virtual view consistent during the navigation. The following paragraph describes how to compute the virtual camera parameters from the real camera calibration parameters for each of the two models used in this work. In the open source library VTK, this problem reduces to the one of computing the virtual camera position, focal point and view up vectors together with its view angles. In the following, we remind that the class `CVec3d` defines euclidean vectors and that the class `CMat4d` defines 4x4 matrices.

8.4.1 Virtual camera in the modified Hoppe's model

In the model of Dr. Hoppe, the class `CPinhole` contains the camera calibration parameters: the camera optical centre $\mathbf{z}$, the pixel projection vectors on the image plane $\mathbf{a}, \mathbf{b}$, the principal point pixel coordinates (n_0, m_0), the distortion center pixel coordinates (n_f, m_f) and the distortion coefficients $\lambda_{i=1\ldots4}$. A virtual camera is in this case defined aligning the image window to the camera optical axis by centering the image midpoint $(w/2, h/2)$ with the line connecting the camera position and the focal point, defining the view up vector and setting the distance between the camera position and the camera plane correspondingly (algorithm 5). Since cameras usually have zooms, a view angle along the up direction must be defined by requiring that the angle between the camera position and the upper edge of the camera plane equals the angle subtended by the camera.

The details of the distortion model don't impact upon the definition of the virtual camera; the definition of the actual pixel-to-point map contains all the non-linear function. It must be at this point kept in mind that the correspondence between the pixels and the optical rays is well defined once a projection plane is chosen and that a natural plane during the camera calibration procedure do exists and is the calibration pattern plane. Therefore the `pixel2point()` function can be decoupled from the definition of the virtual camera and defined as a class independent on the particular distortion model (algorithm 6).

8.4.2 Virtual camera in OpenCV model

In the camera calibration model of Zhang [65], a calibrated camera is completely described by its camera calibration matrix $\mathbf{K}$, containing the f_x and f_y pixel-to-unit scale factors, the skew factor s, the coordinates of the principal point in pixels u_0, v_0, together with the camera extrinsic parameters giving the camera position and orientation respect to an external frame defined by the calibration pattern. In this case, since the camera will be transformed accordingly to the endoscope frame, the virtual camera parameters are defined in the camera frame, where its position is the origin and its view up vector lies on the y-axis. The focal point is then computed as

Algorithm 5 Virtual Camera: model of Hoppe

```
void virtual_camera(CPinhole *pinhole)
{
        double factor = 180.0 / 3.1415926535;

        /* camera focal point */
        CVec3d top = pinhole->pixel2point(pinhole->width/2.0, 0));
        CVec3d bottom = pinhole->pixel2point(pinhole->width/2.0, pinhole->height));
        CVec3d f = 0.5 * (top + bottom);

        /* camera position and its projection of along the plane */
        CVec3d z = pinhole->center, normal = ndir ^ mdir;
        CVec3d zp = z - (z * normal) * normal;

        /* principal point, view up and position - view plane distance */
        CVec3d d = (z * normal) * normal;
        CVec3d up = 0.5 * (top - bottom);
        CVec3d p  = f + d;

        /* camera view angle */
        double view_angle = factor * 2.0 * atan(up.Length() / d.Length());

        /* projection of the view point - focus vector on image plane */
        CVec3d r = zp - f;

        /* oblique angles */
        CVec3d ndir = pinhole->ndir(), mdir = pinhole->mdir();
        double beta  = factor * atan(d.Length() / r.Length());
        double alpha = factor * acos(r.normalize() * ndir.normalize());

        if (- mdir * r < 0.0) alpha = 360.0 - alpha;

        /* pixel aspect */
        double aspect = ndir.Length() / mdir.Length();
}
```

Algorithm 6 Virtual Camera: the distortion part

```
CVec3d pixel2point(u, v)
{
        double d_n = u - nf, d_m = v - mf;

        /* radial distance from the distortion center */
        double r = (ndir * d_n + mdir * d_m).Length();
        double s = r * (d[0] + r * (d[1] + r * (d[2] + r * d[3])));

        /* point on the calibration pattern plane */
        return ndir * (u + s * d_n - n0) + d_m * (v + s * d_m - m0);
}
```

the projection of the principal point on the calibration pattern plane and the view angles dependent accordingly on the camera intrinsic horizontal and vertical scale factors f_x, f_y (see algorithm 7).

8.4.3 Virtual camera: implementation in VTK

The following paragraph describes how to implement an augmented reality camera in VTK once its corresponding virtual parameters are known. A projection plane `vtkPlaneActor` is used as a semitransparent screen to project the camera video upon (see algorithm 8). Once the projection plane position and orientation have been determined in the camera frame, they must be transformed in the endoscope initial frame to allow the virtual camera to move together with the endoscope.

8.5 Single parts accuracy

In this section, the accuracy of every single component is measured and discussed.

8.5.1 Endoscope Camera Calibration

Endoscope camera calibration has been performed for both endoscopes with Zhang's camera calibration algorithm [65] using the computer vision open source library OpenCV. The model decouples intrinsic from extrinsic camera parameters and describes the radial lens distortion as a fourth degree polynomial with the distortion center set at the principal image point. The tangential distortion, usually at least of one order of magnitude smaller than the radial distortion, is instead limited to a second degree polynomial. To perform the camera calibration and obtain the camera intrinsic parameters, a chessboard pattern of 7x10 squares with sides of 3,4,5,6

Algorithm 7 Virtual Camera: OpenCV

```
void virtual_camera(CWindow w, CMat4d K)
{
        CVec3d pixeldir, focus, plane_point;

        /* set camera frame */
        CVec3d camera_position = CVec3d(0,0,0);
        CVec3d view_up = CVec3d(0,-1,0);

        /* set Zhang's camera variable names */
        double t = K.GetTranslation();
        double u0 = K.GetRotation()[0,2];
        double v0 = K.GetRotation()[1,2];
        double f_x = K.GetRotation()[0,0];
        double f_y = K.GetRotation()[1,1];

        /* focal point: project camera principal point onto camera plane */
        focus = fromPixel2OpticalRay(u0, v0);

        /* get the pattern plane point by its z-distance from the center */
        double scale = K.GetTranslation().Z())/focus.Z();
        focus = scale * focus;

        /* get pixel direction for point in the middle of the image */
        pixeldir = fromPixel2OpticalRay(w->width/2.0, w->height/2.0);

        /* get the pattern plane point by its z-distance from the center */
        scale = d/pixeldir.Z();
        plane_point =  scale * pixeldir;

        /* shift origin to focal point */
        camera_position += (plane_point-focus);
        focus += (plane_point-focus);

        /* camera view-angles */
        view_angle_v = 2.0 * factor * atan(0.5 * w->height/f_y);
        view_angle_h = 2.0 * factor * atan(0.5 * w->width/f_x);
}
```

Algorithm 8 Virtual Camera: implementation

```
int renderVideoImageOnPlane(vtk_camera_params p)
{

        double                  w = 0, h = 0, focal_length_cam = 0.0,
                                factor = 180.0 / 3.1415926535;
        CVec3d                  focus_cam;
        vtkMatrix4x4            *matrix = vtkMatrix4x4::New();
        vtkPolyDataMapper       *mapper = vtkPolyDataMapper::New();
        vtkLODActor             *actor = vtkLODActor::New();
        vtkPlaneSource          *plane = vtkPlaneSource::New();

        /* get the focal length */
        CMat4d  invm = p.m.GetInversed();
        focus_cam = invm.GetRotation() * p.focus + invm.GetTranslation();
        focal_length_cam = focus_cam.Z();

        /* get the plane actor dimensions in mm */
        w = 2.0 * focal_length_cam * tan(0.5*p.view_angle_h/factor);
        h = 2.0 * focal_length_cam * tan(0.5*p.view_angle/factor);

        /* define the plane object in the calibration-pattern frame */
        plane->SetOrigin(0,0,0);
        plane->SetPoint1(w,0,0);
        plane->SetPoint2(0,h,0);

        /* vtk */
        mapper->SetInput(plane->GetOutput());
        actor->SetMapper(mapper);
        actor->SetTexture(image_texture);

        /* center principal point in camera frame */
        actor->SetRotation(IDENTITY);
        actor->SetTranslation(
                CVec3d(-(p.principal_vector.X())*focal_length_cam,
                       -(p.principal_vector.Y())*focal_length_cam,
                       focal_length_cam)
        );
        return 1;
}
```

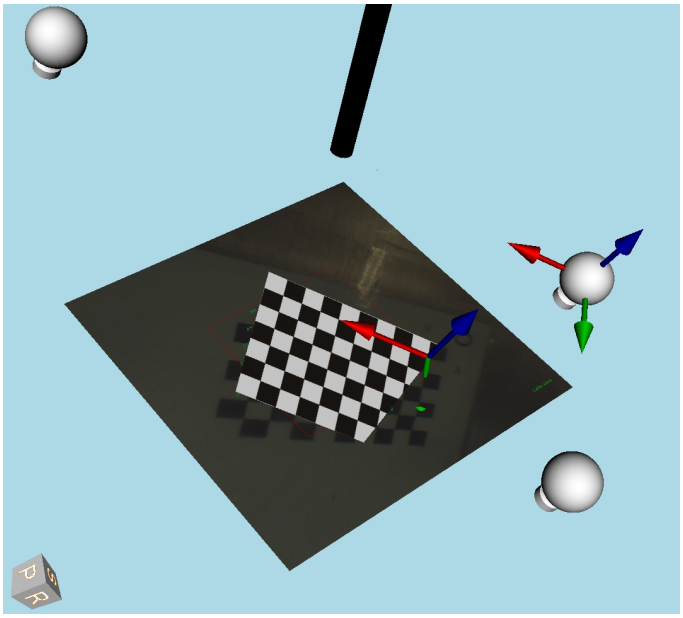
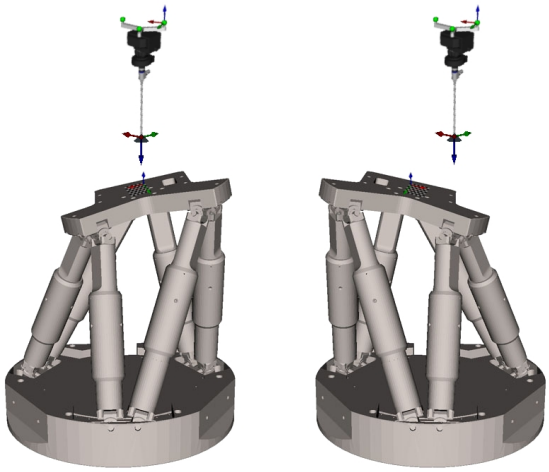


Figure 43: AR implementation as `vtkPlaneActor` texture

and mm has been printed out with a high precision printer and 40 images of the chessboard have been observed at different angles of the camera. According to the technical report [65], particular care has been taken in including many wide-angle screenshots. The camera has been calibrated by keeping the focal length constant at 70-80 mm during the whole calibration procedure to avoid successive recalibration. The accuracy of the overall obtained calibration had an average reprojection error of 0.68 pixels. To discuss this result with the actual camera calibration data, it must be remarked that, given an endoscope effective focal lenghts of 700 pixels/mm and a focal plane at a distance of 70 mm, a reprojection error of 1 pixel is equivalent to 0.05 mm only. A standard camera reprojection error is then well below the significant surgical accuracy of 1.0 mm: an endoscopic camera, as a measuring device, is therefore much more precise than a standard optical tracking system whose single marker positional accuracy is 0.35 mm.

8.5.2 Extrinsic Parameters Accuracy

The camera pose of the endoscope can be measured optically by computing its extrinsic parameters, which express the camera pinhole position from an external reference system set onto the calibration pattern. To test the accuracy of the endoscope pose estimation obtained through the calibration pattern, the camera chessboard has been fixed on a PI hexapode 850 P50 (10 μm of repeatability) and moved all along its working volume of 40x40x20 mm. The goal was to check how the translations measured with the chessboard pattern would have disagreed with the translations measured with the high precision hexapode. During the test, the



(a) (b)

Figure 44: Stability test of the camera extrinsic parameters using an hexapode: the camera is fixed, the exapode is moved

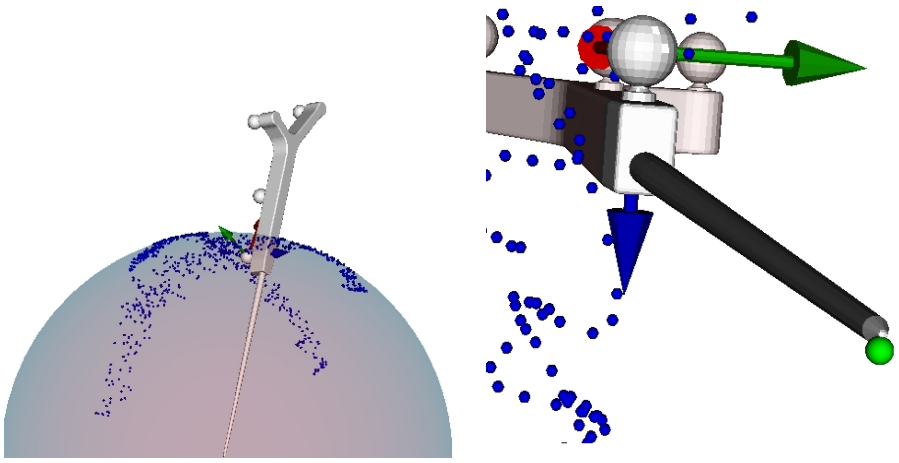
endoscope was kept perpendicular and at 50 mm from the chessboard pattern and the exapode moved along translations. The maximum error between the two camera positions was found at extremal hexapode configurations and amounted to 0.3 mm, with a rms error of 0.1 mm (fig. 44).

8.5.3 Calibration Pattern Registration

The chessboard has been tracked in the tracking system frame by adding 5 passive markers to the calibration chessboard. The transformation between the chessboard rigid body in tracking system frame and and the ideal chessboard frame with origin at the inner chessboard corners has been defined so that it would have been as similar as possible to a simple translation. The four inner corners have been selected and their coordinates picked up in chessboard frame. The accuracy of the obtained registration at each chessboard corner was 0.19 mm with 0.29 mm as maximum error all along the chessboard.

8.5.4 Point Picking

The registration of an object into the virtual scene is usually done using a navigated pointer to measure in the tracking system frame the coordinates of some markers. The coordinates of the pointer's tip are measured in the pointer's frame and then used to pick up the coordinates of any world point. Since the passive spheres of a pointer are usually far from the tip, the same problem concerning the target registration error at the pointer tip applies here.



(a) (b)

Figure 45: Sphere interpolation of pivotized points (a); in green, a magnified representation of the pointer tip pivotized position (b)

Pointer pivotization

The pivotization procedure consists in pivoting a tracked pointer around a fixed point, usually its tip, to measure its coordinates. During the pivotization, any fixed point in pointer's frame spans a sphere (fig. 45). The spanned point cloud isn't an ideal sphere because of the noise in the tracking system and a best sphere in the least squares sense must be chosen to minimize the residuals of the distances between its spherical surface and each measured point. A Gauss-Newton initial guess is first used to minimize the sum of the squared distances instead of the plain distances: this simplified functional has the same absolute minimum as the original one but weights smaller and bigger than units errors differently from the original functional. If the fixed point is therefore the same, the convergence procedure may go through a different path, especially in the last iterations where the distances cross the unit threshold and the problem of the false minima and of the small-scale noise begins to play a role; as an advantage, this functional leads to equations which can be easily linearized and solved. After being measured, the origin's coodinates of the best sphere generated by the pointer are expressed in either the tracking system's or the pointer's frame and the pointer is said pivotized. The accuracy of such procedure is provided by the rms error of the residuals of the distances over the point cloud. As such, it has obvioulsy a direct impact on the registration of any object introduced into the virtual scene.

The pivotization has been repeated 10 times using 500 points. The rms of the residuals was 0.20 mm and was used to quantify the precision of the result. Since the residual has the same order of magnitude of the tracking system's accuracy, the result was judged acceptable. No significant improvements of the accuracy in measuring the pointer tip's coordinates have been observed by increasing the number of points over 500. Since a gaussian error statistics would give a pivotization

error which would scales with the number of collected points as $1/\sqrt{N}$, the existence of this upper limit in computing the tip position acts as a hallmark of either a possible systematic error in the pointer assumed geometry as given in its ROM file, or of a non-gaussian error distribution showing up along the pivotization trajectories which are usually taken in a plane along a gravity plumb line and not along optical lines in the z-direction of the tracking system. In any case, since the pointer tip position accuracy was found below the tracking system accuracy, the result was considered acceptable.

Pointer tip pivotization accuracy

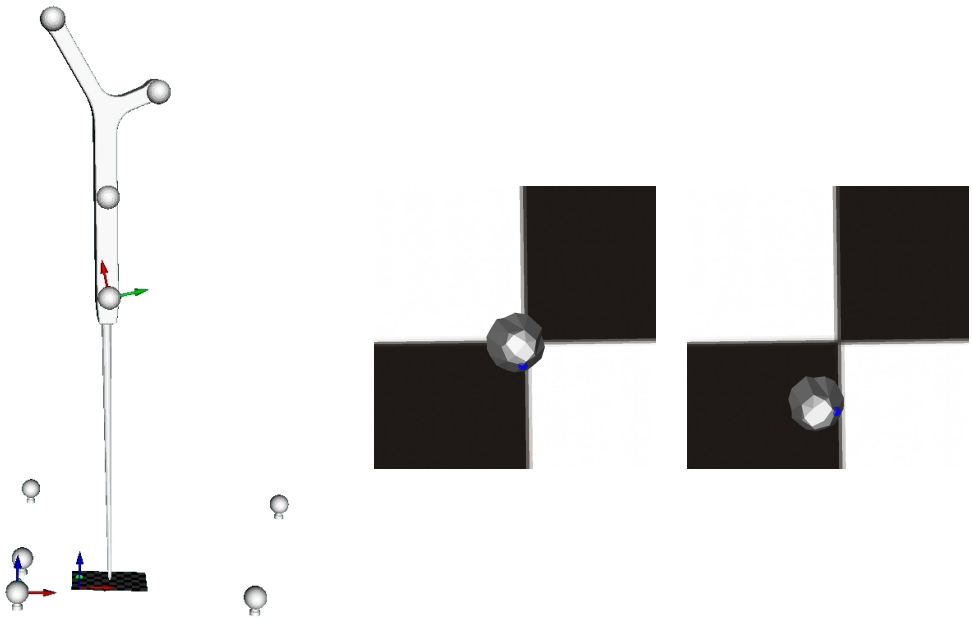
The pointer is navigated using a passive rigid body composed by four retro-reflective spheres which are bound to remain far from the pointer tip. A test of the pointer's tip accuracy shows that the single marker tracking system error of 0.35 mm degrades at the pointer tip until a maximum error of 1.0 mm. To reduce the error in point picking and increase the safety and the reliability of the measurements, a pivotization procedure must be used for each point acquisition. The following test shows how an initially good pointer position at the tip degrades while the pointer is rotated 90° along its z-axis. The tip position is aligned against the calibration pointer and the pointer is rotated along its z-axis. The small blue spot at the edge of the pointer's tip drifts significantly away respect to its initial position (fig. 46).

8.5.5 Endoscope navigation

The endoscope is navigated by attaching a passive rigid body with three retro-reflective spheres on the top of the camera (fig. 39). Retroreflective spheres are detected by the tracking system which locates the sphere with subpixel precision by triangulation, derives the tool geometry in the tracking system's frame and compares it against its ideal geometry in rigid body's frame. The process of deriving the best rigid body transformation between an observed rigid body and its ideal geometry is very similar to the patient registration problem and shares with it the absolute orientation problem of recovering the best-rigid body transformation aligning two different point clouds. In particular, the similarity between TRE and the pointer's or endoscope's tip navigation rests on the common fact that a rigid body is usually attached at the back of an endoscope while the tip coordinates are usually wanted.

Endoscope tip navigation's accuracy

The problem of determining the accuracy at a particular point of a tracked object equipped with a rigid body is very similar to the problem of determining the $\langle \mathrm{TRE}(\mathbf{r}) \rangle$ at point $\mathbf{r}$ when the $\langle \mathrm{FRE} \rangle$ is known. The equivalence stems from the fact that the $\langle \mathrm{FRE} \rangle$ for patient registration equals the single marker's claimed tracking



(a) (b) (c)

Figure 46: Pivotization accuracy test (a); initial pivotized tip position (b); 90° pivotized tip position (c)

system accuracy of 0.35mm. This fact is true in particular for surgical tools having their navigated tips usually several centimeters far from their markers. In the technical report of the NDI company [53], the error in a measurement at a tool tip is related to the orientation uncertainty of a rigid body as:

$$\epsilon \propto d \cdot \tan(\Delta\theta) \tag{52}$$

where d is the distance from the rigid body centroid to the tool tip and $\Delta\theta$ is the rigid body orientation uncertainty. As a reminder for the reader, tracking system's accuracy is defined as a single marker accuracy; the orientation uncertainty is instead defined respect to a flat rigid body only whose markers lie on a plane. Because of the similarity between the patient registration problem and the navigation problem, a better estimate could be obtained by using Fitzpatrick's formula to predict the tracking system error at the tool tip as a very special target registration error. Using the above mentioned similarity, the error ϵ at a point $\mathbf{r}$ would depend on the distance of the point $\mathbf{r}$ from the rigid body principal axis and on the number of spherical markers N as in:

$$\epsilon(\mathbf{r}) \propto \sqrt{\left(1 - \frac{2}{N}\right)\left(1 + \frac{1}{3}\sum_{k=1}^{3}\frac{d_k^2}{f_k^2}\right)} \tag{53}$$

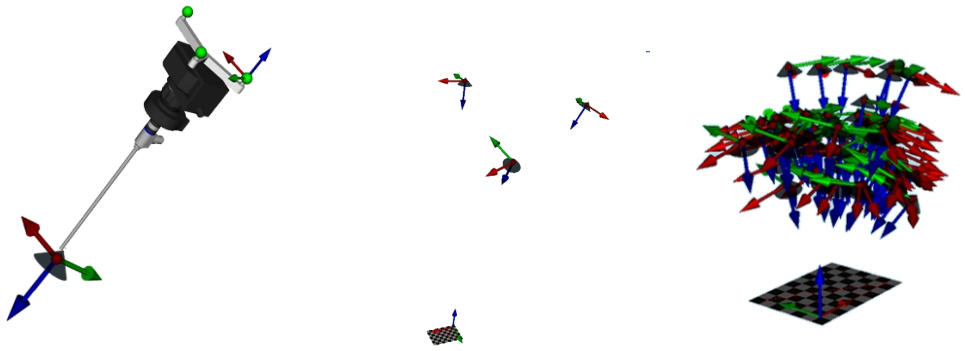
Particular care must be used in applying this formula for rigid body accuracy estimation. Fitzpatrick's formula assumes markers'positions as random variables subject to gaussian noise; as a consequence of this, $\langle \text{FRE} \rangle$ is independent of the rigid body geometry. Contrary to this assumption, the tracking system noise is strongly z-dependent because of its perspective behaviour. This difference may provide a geometry-dependent $\langle \text{FRE} \rangle$ questioning the application of a plain Fitzpatrick's formula to this case. Unless the z-dependence of the positional error is found almost constant over an average pivotization trajectory volume, another version of a Fitzpatrick like formula, with a Maxwell-like distribution and a more pronounced z-tail should be used to tackle this problem.

8.6 Calibrated camera tracking

The derive the camera pose in the moving endoscope frame, the chessboard was registered into the tracking system using well known retro-reflective marker balls. With this setup, we have been able to connect the tracking frame to the camera frame using a reliable and reproducible procedure. The camera pose has been determined in the moving endoscope frame with two different methods: the first measure has been done by difference inverting the endoscope and the chessboard pattern frames and the second one using a hand-eye calibration method. In the first measurement the camera pose is transformed from the camera frame to the chessboard frame, then to the tracking system frame and finally to the endoscope frame. In the second measurements the camera was moved all along the planned volume of interest and a C implementation of the dual quaternion hand-eye calibration method [83] was used to compute the endoscope to camera frame (fig. 47). Both these procedures do compute the same endoscope to camera transformation matrix, the only difference being that hand-eye procedures derive the best global endoscope to camera transformation matrix all over the working volume while the direct measure of the same transformation obtained by difference depends on the actual endoscope and camera (and tracking system) spatial configuration and relies on a differential procedure. For a completely different approach, see [96].

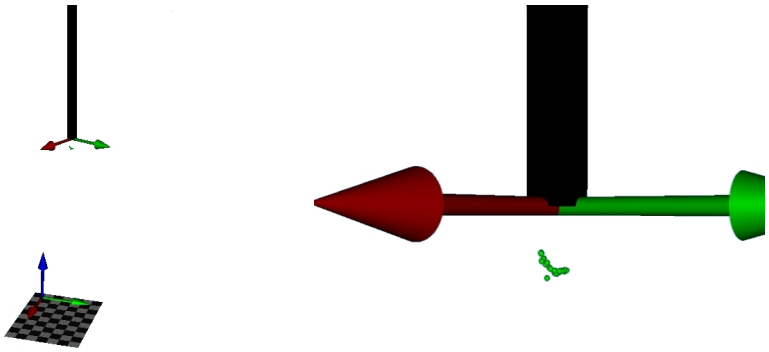
Camera pose static accuracy

The camera pose accuracy during the tracking is the main issue of the application and has been tested by moving the tracked calibration pattern in various positions while keeping the endoscope fixed. With this setup the estimated endoscope pose should have in principle remained constant all along the test volume for any chessboard position. In practice, because of the noise of the tracking system, the camera pose spreads over a sphere of rms radius of 1.0 mm (fig. 48).



(a) (b) (c)

Figure 47: Hand-eye method to measure the endoscope to camera frame; camera pose (a); a few endoscope poses (b); dense endoscope pose cloud (c)



(a) (b)

Figure 48: Static pose accuracy test (a); a magnified picture showing the endoscope camera pose spatial spread (b)

8.7 Augmented reality in the NEAR project

The measurement of the position of the camera pose in the endoscope frame allows to endow the NEAR project of augmented reality capabilities. For its realization, the endoscope camera pose was first measured in the initial endoscope frame and stored. Its position was then successively estimated by multiplying the initial camera pose frame times the difference between the initial endoscope frame and the moving endoscope frame. This expresses the obvious fact that the camera pose moves rigidly with the endoscope, a statement whose experimental truth cannot be taken for granted and depends highly on the degree of accuracy with which a researcher is able to measure its initial position in the endoscope frame. To test the augmented reality accuracy, both the navigated chessboard pattern and the phantom Lucy were used. In the first test the chessboard pattern was first detected and a corresponding virtual chessboard was overlaid onto it while the pattern and the endoscope were moved freely throughout an ETV-like working volume (fig. 49(a)-49(b)). In the second test the phantom Lucy was registered in the virtual scene using eight fiducial titanium screws segmented in a MRI phantom scan (fig. 49(c)-49(d)). Its brain ventricles position were highlighted during the navigation phase in the augmented reality view modality. The ventricle walls position were used to show the available working volume during an ETV test performed throughout one of the Lucy phantom channels (fig. 49(e)-49(f)).

A representation of the brain ventricular structure is shown on a test navigation done on the Lucy phantom (fig. 50). The brain ventricles and a tumour are highlighted against the live video background, allowing a fine positioning of the endoscope entrance angle on the patient's simulated anatomy. The red and green spheres shown respectively the planned and the measured point clouds used during the test. The differences in their alignment make the FRE visible for each fiducial position. The mismatch shown at the fiducial on the very top of the image is due to the fact that during the test the video wasn't undistorted.

8.8 Camera pose stability

The stability of the camera pose during the navigation remains a major issue for any augmented reality or 3D reconstruction application. The camera frame is in fact measured first in a single endoscope setup; its actual position is then obtained by transforming it into the time-dependent endoscope frame. Since any error collected in measuring the initial endoscope position will impact onto the actual navigated endoscope position, we found that the best method for estimating the camera pose during navigation was starting with a direct measurement of the endoscope to camera transformation. Measuring the endoscope to camera transformation with the hand-eye method gives in fact the best but already optimized transformation all over the working volume, bringing an equal amount of error on every estimated

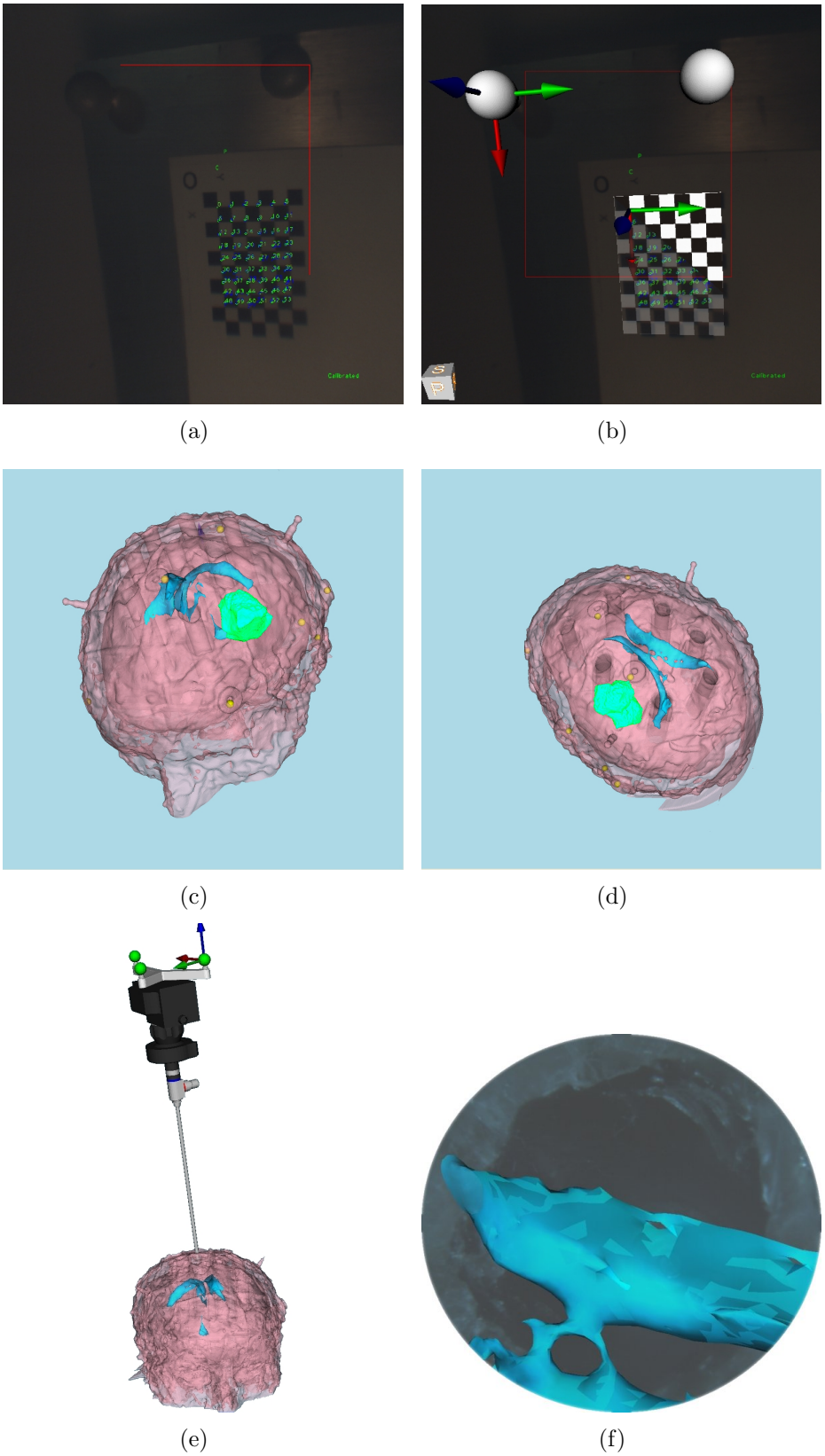


Figure 49: Detected pattern (a); AR on image plane (b); phantom registration (c); navigation through phantom channels (d); endoscope introduction (e); AR view of brain ventricle (f)

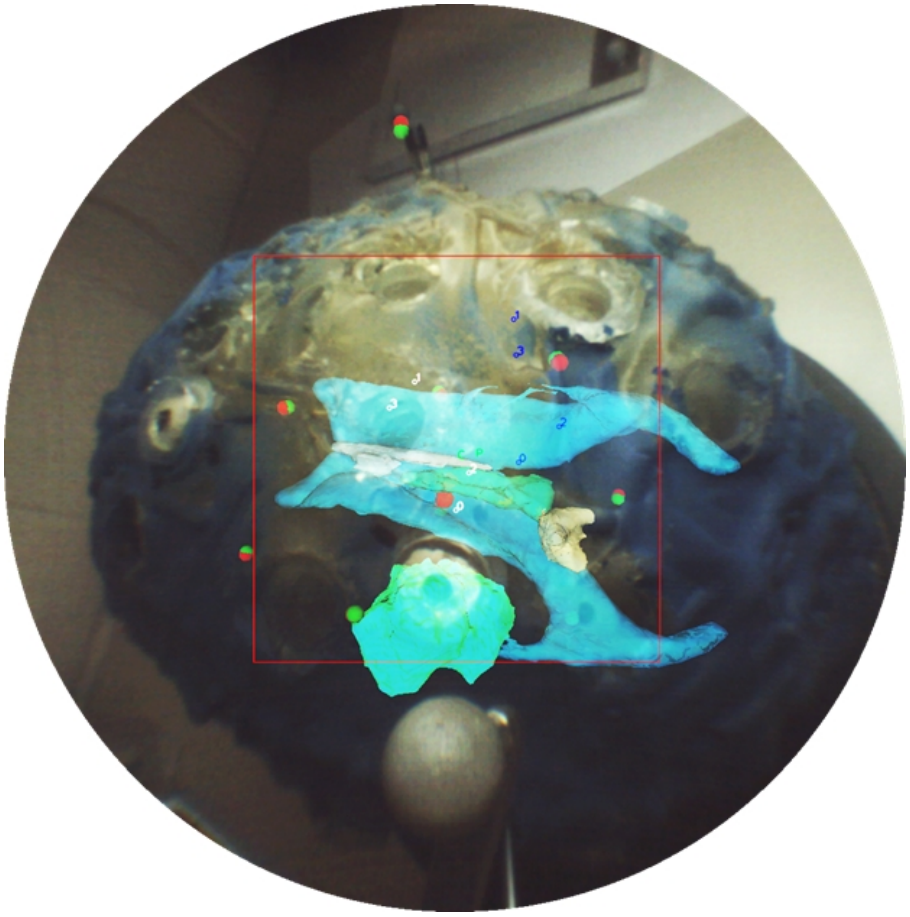
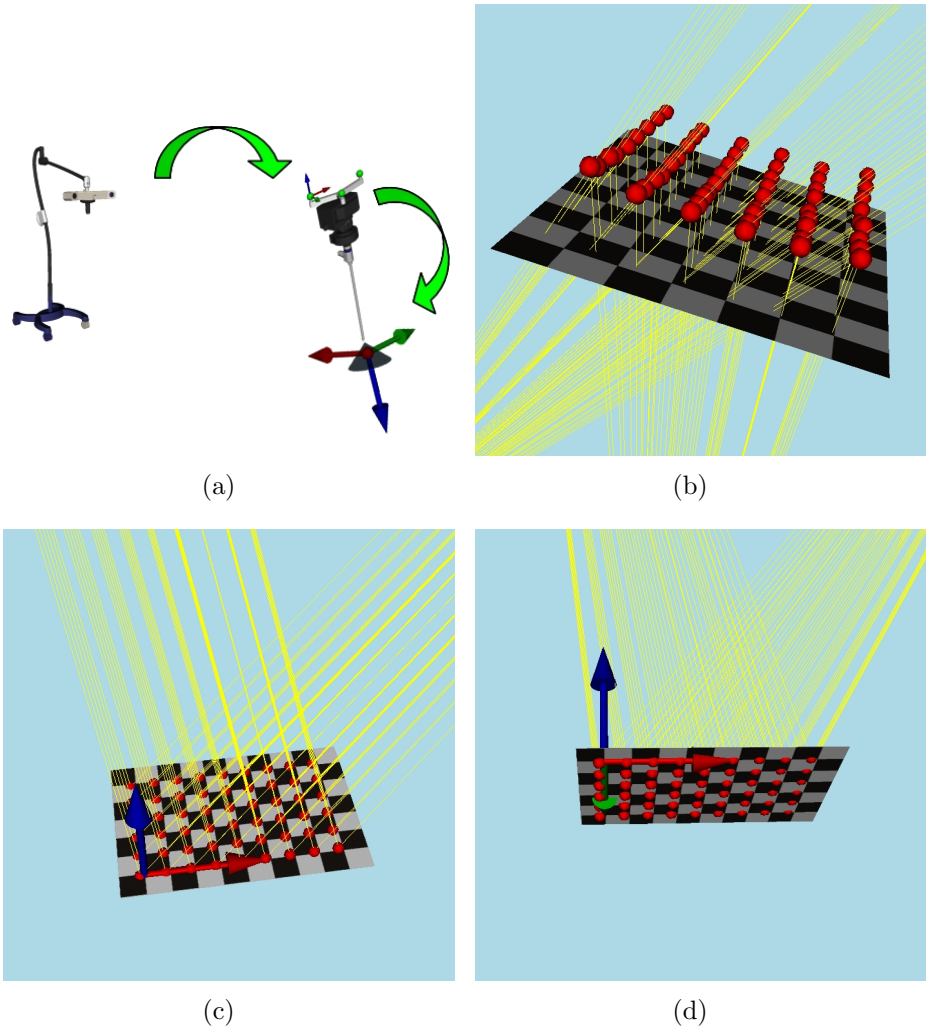


Figure 50: An augmented reality screenshot of the Lucy phantom, showing the ventricular brain structure and a tumour. The planned and the measured registration point cloud are also shown



(a) (b)

(c) (d)

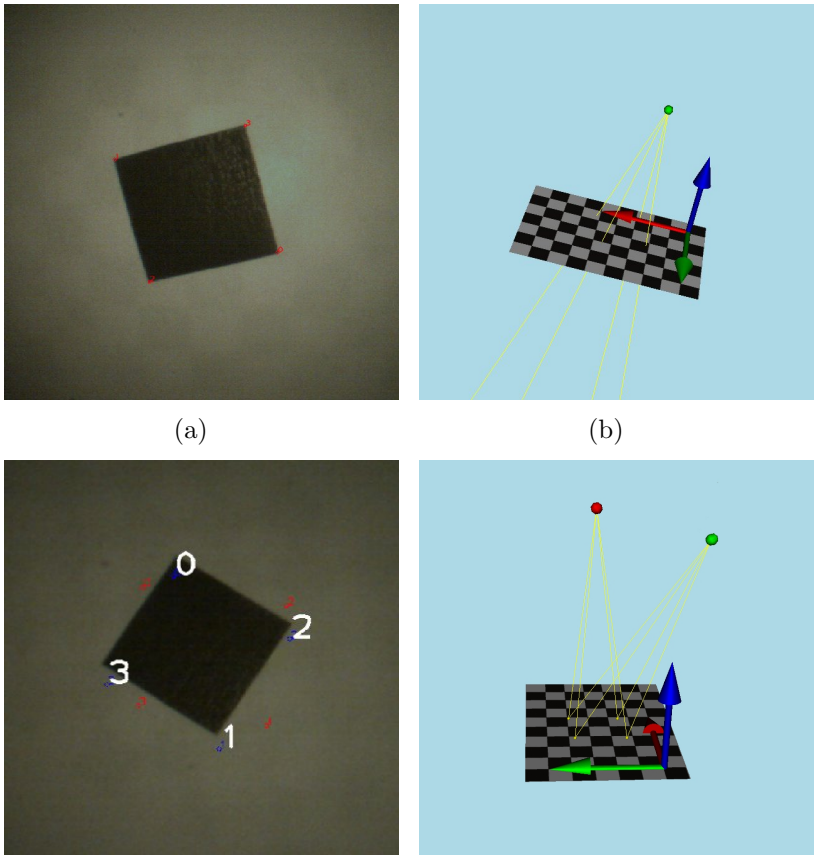
Figure 51: Hand-eye endoscope pose estimation (a); triangulation test with hand-eye global pose estimation (b); triangulation test with local differential pose estimation (c); triangulation test with local differential pose estimation (d)

pose of the navigated camera. This produces a global triangulation error equally affecting all the endoscope estimated poses, result which can be checked by performing a triangulation test using the registered calibration chessboard. Since in this particular test there's almost no error in the chessboard pixel detection step, the main error source comes from the endoscope camera pose estimation only. A triangulated position of its edge 3D positions usually gives an average coarse-grained depth triangulation resolution (fig. 51(a)-51(b)). On the other hand, a triangulation performed using a first locally measured endoscope to camera matrix and a second differential estimate of the first endoscope camera pose, even if not able to give the same degree of precision all over the working volume, can offer a substantial increase of the depth triangulation precision (fig. 51(c)-51(d)).

8.9 Triangulation

To perform the triangulation with an endoscope, corresponding features must be identified on successive video images. Corresponding pixels were selected using the OpenCV implementation of the Shy-Tomasi feature tracking algorithm [87]. The algorithm decomposes the image intensities into their eigenvalue and eigenvector components and detects among all the image features the most robust, where as usual in computer vision robust means robust to be tracked. This choice of the Shy-Tomasi feature detection algorithm has two major drawbacks: first, the features are selected by the algorithm and not by the surgeon: the features are indeed chosen on computer vision grounds, not on surgical grounds. It's possible to overcome this restriction by letting the surgeon identifying a region of surgical interest on a patient and bounding afterwards to this area the region of interest of the algorithm. This way of proceeding restores the surgeon's freedom of choosing almost any area of interest, but has also the drawback of requiring a manual intervention from the surgeon. A practical removal of this constraint is based on the obvious fact that since usually surgeons look at the center of the image when performing an endoscopy, the suitable area of interest can be set as a rectangle or a square located in the middle of the camera image. A more interesting possibility already offered by the OpenCV library would be setting the region of interest as the output of a suitably trained Viola-Jones [97] classifier. After training a database of common and relevant patient's anatomical landmarks, this procedure would allow to automatically track pixels by restricting them to surgically significant areas and identifying them as soon as they will be visible to the surgeon. Even if interesting as a research line, this work assumes that the surgical region of interest is defined as a square set at the center of the endoscopic image. The second drawback in using the Shy-Tomasi feature detection algorithm comes from the fact that the selected pixels do include specularities. Specularities are points of specular reflection whose positions depend both on the observed object and on the light geometry, properties and position. In computer vision they have always represented problems and opportunities: they don't describe directly pure object properties as object features do. Since we assume that the external object geometry doesn't change while the endoscope is navigated, or equivalently, since any elastic deformation occurring happens on a so slow time scale that it can be neglected, specularities must be filtered out from the set of good features to be tracked. Their geometrical information content depends on the endoscope's light source position which changes during the operation as the endscope moves.

OpenCV offers also a free implementation of the Lucas-Tomasi-Kanade (LTK) optical flow feature tracker to match correspondent pixels on different endoscopic images. In the OpenCV implementation good features are located by examining the minimum eigenvalue of each 2 by 2 gradient matrix, and features are tracked using the Newton-Raphson method of minimizing the difference between the two windows. To test the feature tracking module, four corners of a dark square have been selected as easy features to be tracked with the LTK algorithm. Using two



(a) (b)

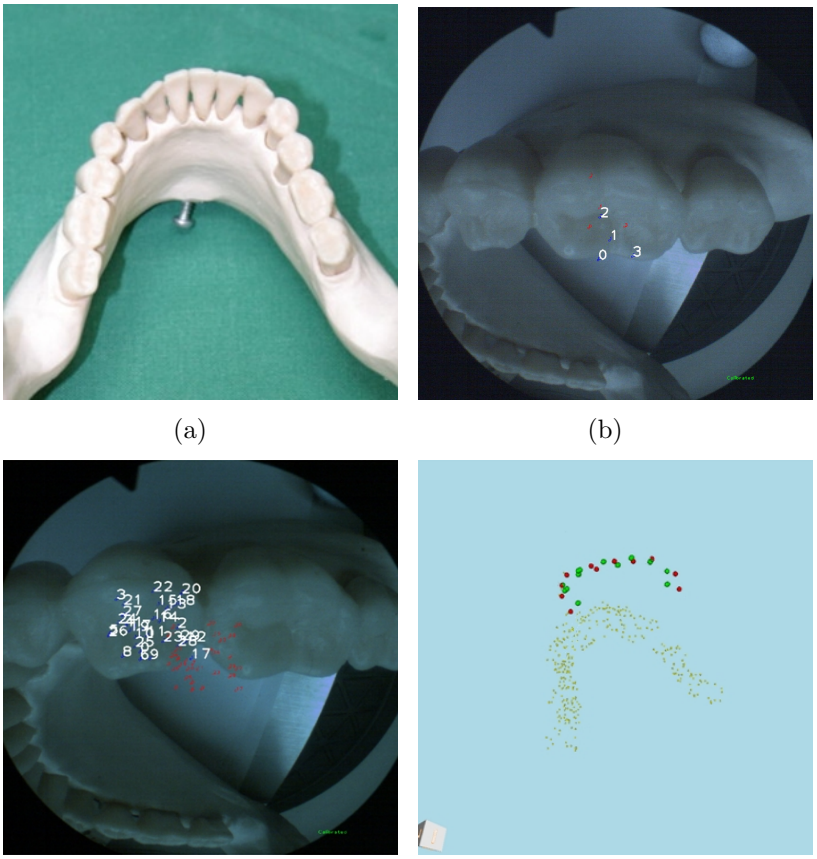
(c) (d)

Figure 52: Tomasi-Shy detection (a); edge ray reprojection (b); edge LTK optical flow (c); edge triangulation test (d)

different endoscope poses and small movements between them, the four corner positions were detected from two near endoscope poses and then triangulated (fig. 52(a)-52(d)). The following results show three different and independent remarks: the first one is that a robust feature tracking can be used to follow artificial features when endoscope movements compatible with the surgical ones are performed; the second one shows that small surgical movements do not prevent necessarily precise triangulation; the third one shows that starting with an initial measurement of the endoscope to camera transformation and estimating the second endoscope pose differentially using only standard optical tracking systems, is possible to obtain over a reduced small zone reliable triangulation results.

8.9.1 Triangulation tests

The following section describes the tests done to caracterize the NEAR triangulation module. The tests show how the triangulation is reliable only on tight working volumes. Endoscopic third ventriculostomy, with its insertion of the endoscope



(a) (b)

(c) (d)

Figure 53: Original test body (a); spare feature selction on test body (b); dense pixel clouds tracking (c); triangulated shape (d)

along a straight line belongs to the class of reliable working volumes.

Extended object triangulation

As first example of triangulation using an endoscope, a plastic jaw was selected and observed from twenty couples of views from a distance of 10 cm approximately all along its length. Pixel clouds of 30 features have been selected for every tooth and tracked with the Lukas-Tomasi-Kanade algorithm. The first image on the left shows the initial jaw model used. In the middle image an endoscopic view shows in light red numbers the initial feature positions and in white their final positions. The amount of relative movement of the features describes also the correspondent endoscope shift in space. The triangulated point clouds obtained showed a coarse resolution on the jaw where single details of teeth are not able to be resolved (fig. 53(a)-53(d)).

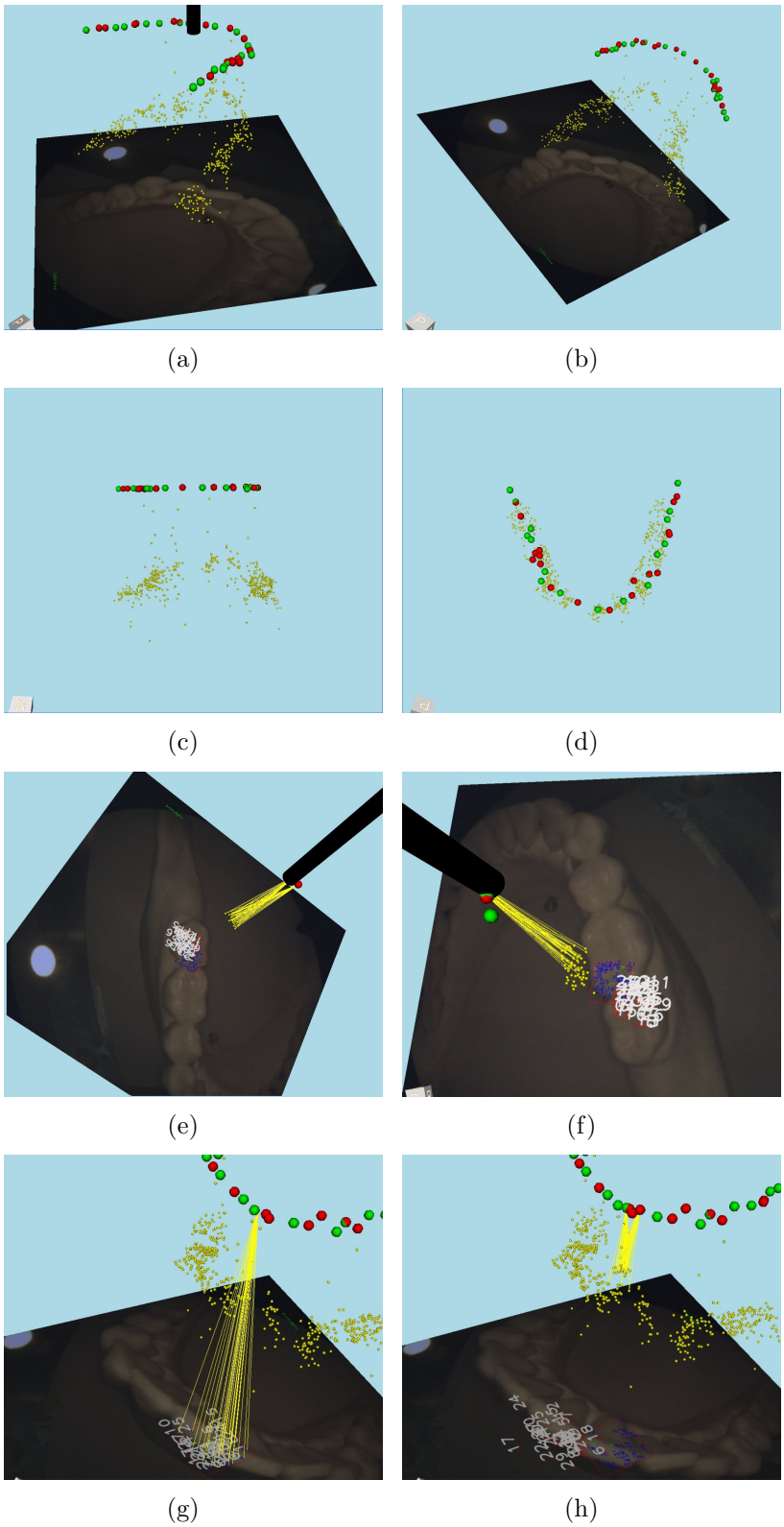
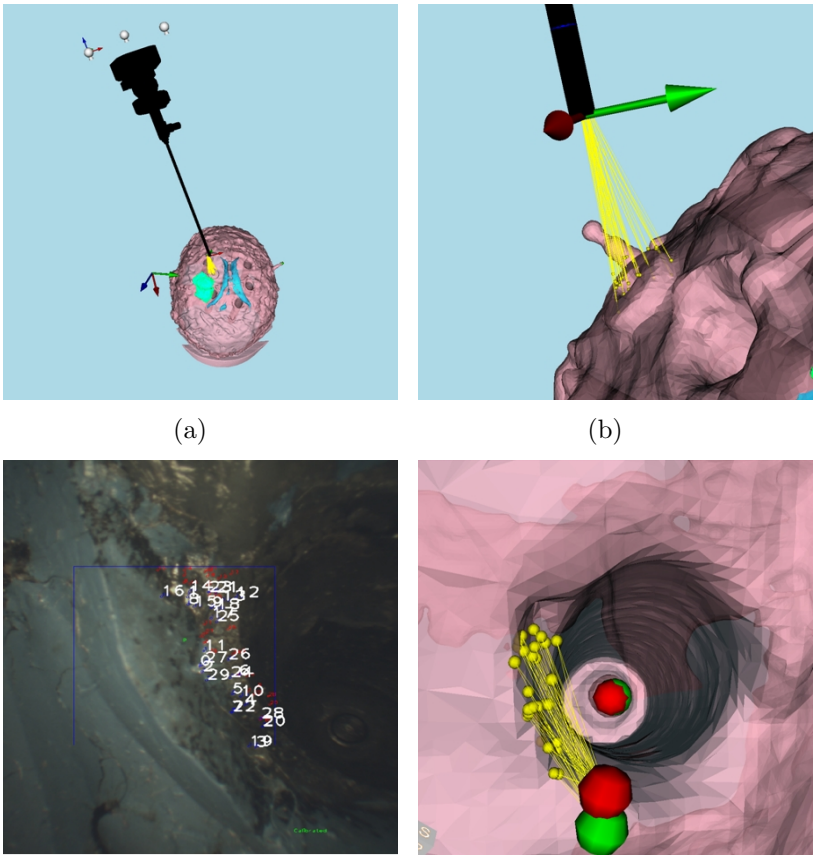


Figure 54: Triangulation phases of a plastic jaw: poses and point clouds (a-d) and the corresponding optical rays (d-h)



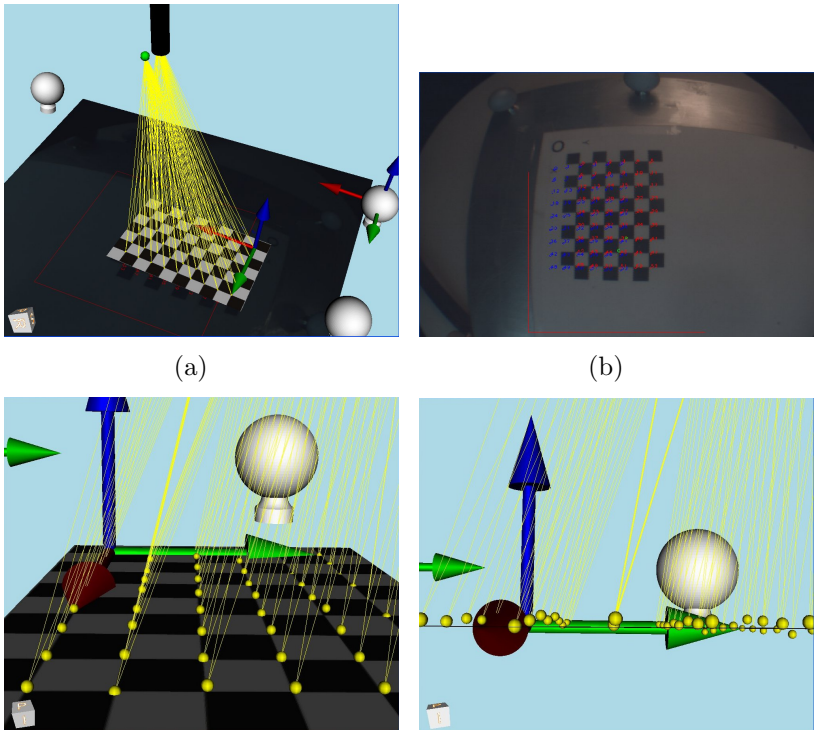
(a) (b)

(c) (d)

Figure 55: Triangulation setup (a); optical rays for features (b); detected features (c); triangulated features (d)

Local object triangulation

As a second example we tried to triangulate some detected features out of two views on a particular artificial soft tissue surface on the registered phantom Lucy. In this second case the triangulated point cloud comes to the desired area with greater accuracy than in the first triangulation example (fig. 55(a)-55(d)). If the previous test showed that global triangulation on a large working volume is not possible because of the drift error in the camera pose tracking, the second showed that a reliable triangulation over a small working volume is still allowed. In the test, some freely detected features have been selected at the edge of one of the phantom navigation channels and triangulated. The results show that the point cloud is located onto the registered phantom model with an average error of 1.0 mm.



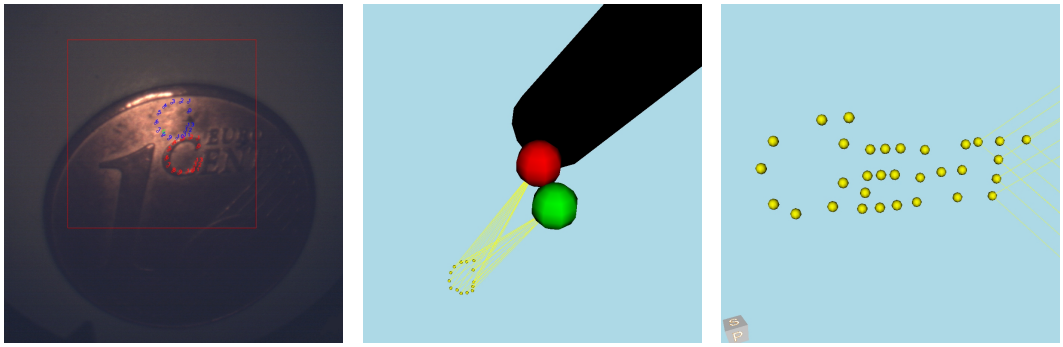
(a) (b)

(c) (d)

Figure 56: Triangulation stability against small angle endoscope poses

Triangulation stability against small angles

In this test, the stability of the triangulation module has been tested against small angles in the endoscope pose. Small angles are angles subtended by tiny movements of the endoscope, of the order of a few millimiters. Since small angles affect the measured depht via the perspective law, a possible outcome of the experiment would have been that the triangulation could have not been stable when endoscope poses submitting small angles were involved. To test the endoscope pose stability only and to avoid any error in solving the correspondence problem at the image level, the calibration chessboard has been used as a test object. The endoscope has been moved in such a way that the angle between two of its poses was about 3°. The chessboard edges have been detected [98] as usual with subpixel accuracy. As a perhaps counter-intuitive result, a triangulation under small angle poses was found reliable (fig. 56). This is possible because *small* angles produce on the image plane *big* visual displacements which are used in turn to compute a finite difference in the endoscopic pose frame transformation. This test shows that small angles don't prevent by themselves a reliable triangulation and that the negative result of the first triangulation test is entirely due to the growth of the inaccuracies in its camera pose estimation while the endoscope is freely moved over a large working volume.



(a) (b) (c)

Figure 57: Triangulation of the inscription on a 1-cent coin; the test shows how fine scale details can be resolved using the visual information of the endoscope

Triangulation of a cent coin

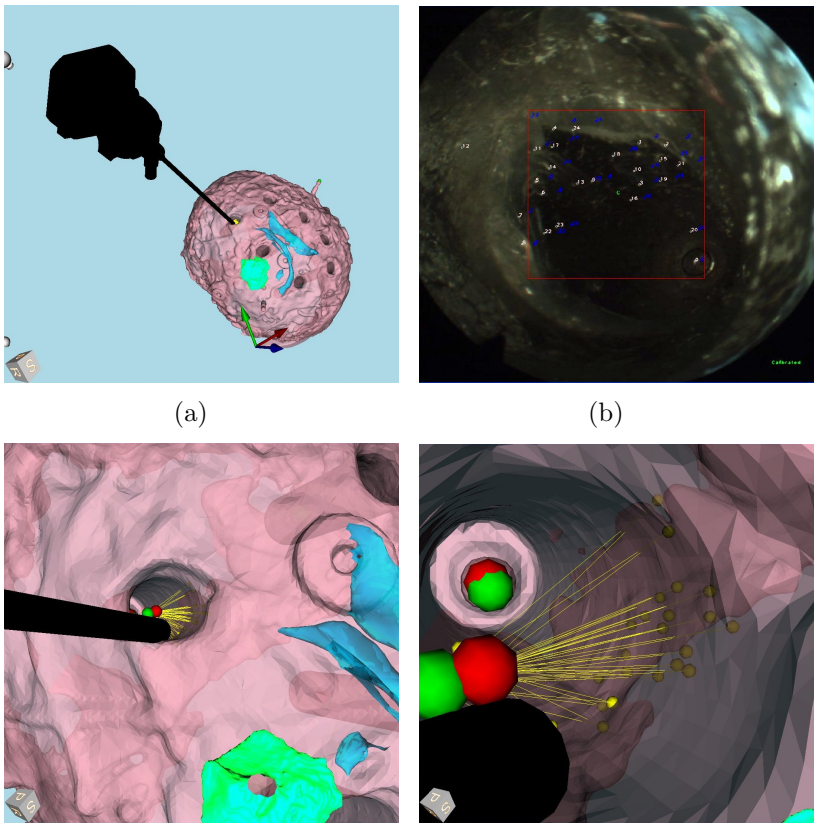
In this test, the selected pixels on the image have been picked up manually to avoid specularities problems in their detection. The poses of the endoscope have been estimated using the navigation system tranformation matrices. Small angles displacements, found successful in preveious tests, have been used between couples of endoscope's poses to extract the position of the triangulated point cloud. The test shows that also a small-scale structure can be recovered when the error in estimating the endoscope pose remains small (fig. 57).

Triangulation of inner phantom navigation channels

With possible applications to intraoperative registration, in this test features detected onto the surfaces of the inner navigation channels of the phantom have been triangulated. The endoscope have been introduced in one of the phantom navigation channels and some free features have been selected on its walls. By selecting two small angle poses, the surface pixel clouds tracked on its interior walls have been turned into 3D point clouds with an average error for the point cloud of 1.2 mm. In case of the endoscopic third ventriculostomy, the triangulated features would mimic the corresponding inner features on the third ventricles walls. The actual position of the ventricle walls could be then successively used for an intraoperative registration or for further referencing.

8.10 Clinical evaluation

The NEAR system has been shown to two different groups of 5 and 8 surgeons and assistants respectively of the Neurosurgical Department of the University Clinic of



(a) (b)

(c) (d)

Figure 58: Triangulation of inner channel features

Ulm, under the supervision of Dr. Med. Marc-Eric Halatsch and of the Neurosurgical Department of the Neurosurgical Clinic of Günzburg, under the supervision of Prof. Dr. Christian Rainer Wirtz. The interest shown in the NEAR application splits in two classes, depending on the level of experience of the surgeon. If not otherwise stated, the endoscopic third ventriculostomy was used as a surgical scenario during the group interview. To better understand the outcome of this evaluation, it must be kept in mind that at the time of this writing, endoscopy is performed in the daily surgical practice as a passive technique only, with no navigation or augmented reality support at all.

8.10.1 Assistants Evaluation

Assistants are students of medicine during their first years of specialization. None of the assistants in the test group had, at the time of the interview, had any experience in endoscopy. The following features were judged as promising:

Planning The support of a precise planning during the operation has been judged very useful by young surgeons. The plan should include but not be limited to, the endoscope insertion point, the insertion angle respect to the patient position, and the correspondent anatomical landmarks used to reference internally.

Virtual Training Assistants or young surgeons have to learn how and where to position the endoscope. They usually practice on phantoms and cadavers before performing real surgery. The possibility of a virtual simulation of any endoscopic procedure would, in their opinion, greatly help any young medical doctor.

Image Training Assistants and young surgeons learn the human anatomy from a completely new perspective when they introduce an endoscope into a patient brain for the first time. Anatomical landmarks used as a reference are not recognized as such in the first endoscopic operations and acquiring this kind of visual experience requires time. Any support in the identification and recognition of landmark 2D images done with AR support or with plain computer vision techniques would surely help the training of any young surgeon.

Navigation The navigation of the endoscope was judged as a very interesting modality to locate the endoscope tip respect to the patient model. The availability of many visualization modes, like the external and the subjective virtual modes was judged as not crucial but as a valid support to the young surgeon.

AR Augmented Reality was judged as a very interesting modality for extending the information content of the video observed. There was no general consensus in the group about how to render the virtual details: an immediately

detectable rendering obtained by using virtual colors was compared with a method obtained by texturing the model with images of the tissues observed and therefore faithful to the observed scene, but no real preference was expressed by the group. Also a non-realistic rendering of the scene plays a role by helping the surgeon to distinguish what is simulated from what is real.

Point Cloud The availability of intraoperative 3D point cloud extraction was only judged as an interesting future modality. The group of assistants found no particular application as the proper target for it.

8.10.2 Experienced Surgeons Evaluation

Experience surgeons perform open surgery routinely every day. They are used to operate with or without the support of surgical navigation. Endoscopists, in particular, have a sound experience in performing minimally invasive operations. They are trained in recognizing the human anatomy on endoscopic images.

Planning, Training Both planning and training were judged unnecessary for the experienced neurosurgeon and endoscopist. Experienced surgeons know very well where to position the endoscope respect to the patient's anatomy and which is the optimal insertion angle. The fact that the tested system could provide or at least assure a standardization of the surgical procedure was found by the experienced surgeons only as a tiny advantage.

Navigation Navigation is judged only a mere support to the experienced surgeon which usually looks directly at the plain images. The problem of a reliable and faithful patient registration, of the brain shift and of the necessary update of preoperative virtual models, reduces the acceptance of navigation support. Surgical navigation is found a helpful complementary but not a necessary tool.

AR Augmented Reality is judged by expert surgeons necessary only in those cases where a structure is not clearly visible. This is the case, in endoscopic third ventriculostomy, of the basilar artery which can be effectively located only by mistake producing a bleeding, or of the mammillary bodies which are hidden when the floor of the third ventricle is opaque.

Point Cloud The availability of 3D point cloud extraction was judged as an potential key feature in solving the problem of the brain-shift or of the model update. The possibility of performing a quantitative measurement of the observed tissues was judged helpful but not necessary. Experienced surgeons and endoscopists are trained to estimated the size of observed features from endoscopic images. The possibility of a surface scene reconstruction, possibly combined with a model update, was found highly interesting from all the experienced surgeons in the whole context of surgery.

Pressure monitoring In addition to the present features, the possibility to monitor the intracranial pressure of CSF at the endoscopic tip was suggested as a possible development of the already existing endoscopic technology.

8.11 Analysis of the medical feedback

From the previous section, it can be clearly observed that the comments about the system split in two clear classes:

- Young surgeons and assistants evaluate the role of virtual simulation and AR in endoscopy for training purposes. A good rendering environment is perceived as an advantage even if its practical implications are not clear.
- Experienced surgeons and endoscopists evaluate navigation as an unnecessary but useful support and AR as a necessary support as long as the added information can be trusted. In doubt, they prefer to turn to their experience. A good rendering environment is perceived as an advantage only when it adds reliable information.

8.12 Summary

The work presented shows the NEAR project, a novel implementation of an endoscope navigation system equipped with augmented reality. It highlights quantitative endoscopy versus augmented reality endoscopy as a future bottom-up approach in trying to collect up-to-date intraoperative models. It shows on a practical ETV phantom that differential techniques of pose estimation obtained using optical tracking systems are preferable to hand-eye calibration techniques when it's possible to constrain the operation volume to restricted movements from an initially good determined configuration.

Chapter 9

Conclusions

This chapter summarizes the results obtained in the NEAR project with particular focus to the contributions given to the fields of camera calibration algorithms and medical system development.

9.1 Overview

Endoscopes have always been used as passive inspection systems. They are extensively used in minimally invasive surgery (MIS) to inspect, operate and remove tumours and considered as a MIS viewing modality paradigm. It's not difficult to forecast that the next generation of endoscopes will allow via augmented reality (AR) techniques to overlay preoperative 3D models of tumours or critical structures on its video as a basic feature.

AR is an already mature technology not yet fully exploited in the current surgical practice. In surgical microscopes, for instance, only a 2D contour of the preoperative tumour position is rendered on its image and endoscopes with AR support are still not used in current endoscopic procedures. Neurosurgeons prefer to rely on their visual experience at the point that endoscope navigation is still in current endoscopic practice only considered as a pure support.

Though AR fits the actual surgical needs very well, it is a pure passive technique: preoperative data is overlaid onto the intraoperative images to highligh regions otherwise difficult to be distinguished from healthy tissue. This is the case, for example, of low-grade gliomas in neurosurgery. Unfortunately, no quantitative measurement of the intraoperative scene is taken during the operation. The added information is then merely preoperative.

As a matter of fact, the geometry of the inspected area changes while the operation is performed. This happens as a result of many factors: a change of intracranial pressure, elastic deformations and relaxations of soft tissues after craniotomy in the direction of the gravity, and, of course, tissue removal. As a consequence of

this design, tumours and critical structures shift from their preoperativly planned position to new intraoperative ones, with a possible shift of several centimeters. Preoperative models used to augment current surgical images with AR becomes rapidly obsolete.

The need for active andoscopes, able to combine 3D optical flow and reconstruction with AR techniques to scan the inspected surface and update the AR patient model, is then made clear. Their possible applications include model patch, intraoperative registration, further referencing until model and tumour shift correction.

9.2 Results

This work has shown the need for active endoscopes, devices able to augment the observed images with both preoperative and intraoperative information derived from the inspected scene. The NEAR project has provided an implementation of an active endoscope with the following results:

- A new camera calibration model has been discussed and developed for endoscope fisheye lenses. The previously developed model [80] has been modified in the computation of the distortion center and adapted to high distortion lenses with subpixel precision.

- The NEAR system, a multi-window based architecture with both AR and triangulation modules has been developed anew. Augmented reality has been built from scratch using VTK basic capabilities. The system accuracy has been tested on a laboratory phantoms.

- The work introduces the concept of active endoscopes, able to perform AR and triangulation on the observed scene. It suggests this features as a possible bottom-up solution to the brain-shift class of problems via a model update approach.

- In the framework of endoscopic third ventriculostomy (ETV) where the working space is very tight and the trajectories are bounded to be almost straight, it has been shown that the navigated endoscopic camera pose can be used to perform realiable triangulation. Small angles between endoscope poses don't necessary prevent the triangulation to produce useful results. The initial camera pose tracking with its drift during the endoscope navigation done with the tracking system remains the main critical issue affecting the overall system accuracy.

9.3 A retrospective look

The combination of computer vision and surgery offers huge possibilities of technological improvements to the surgical field. Today augmented reality (AR), combining the real and the corresponding virtual views of the patient, is perceived as the next crucial technological advance which will modify the way surgeons will look at their intraoperative images in the operating theatre. AR solves many practical surgical problems: it allows to discern healthy from malignant tissue, to visualize the diagnostic information onto intraoperative images, to make visible the position of critical organs, veins or vessels when they are hidden or not directly visible. When computer vision allowed the camera pose to be measured and tracked, the idea of fusing the diagnostic with the intraoperative information was welcomed as a technological advance by the surgical community.

However computer vision offers much more than the simple AR. It's true that AR enables the surgeon to get a sort of X-ray view over the patient, but it must be kept in mind that the augmentation of the real view is today still obtained by mixing the preoperative data taken *before* the operation with the real data taken *during* the operation. Medical researchers and AR engineers struggle together to avoid any visual mismatch between these two sources of information. The most promising techniques in this direction are intraoperative imaging, which updates the data by performing a new patient scan after the craniotomy and the elastic models, which try to mimic the elastic deformations happening after the craniotomy. Both solutions don't solve the problem when the brain's planned anatomy is altered during the surgical removal of brain tissue. This situation requires the continuous scan of the observed area and the use of special iron-free surgical tools and is the subject of the most advanced open MRI research technique.

Another possibility is offered by using the current image views together with scene reconstruction techniques to correct preoperative into intraoperatively up-to-date patient models. From the theoretical point of view, this extends the concept of AR which aimed in its early birth at introducing pure virtual objects into real environments respecting their perspective transformation properties. In the classic example of the AR, a virtual non existing cube is rendered into a real scene and transforms, when the camera is moved, as if it were perfectly immersed in the real environment.

The medical field pushes AR beyond its initial limits. It proposes situations where the virtual objects introduced into the real scene are models of real anatomical parts which were simply not visible at the time of their acquisition. The issue of their alignment when both models become visible, with the complication that the real parts could have suffered of elastic or plastic deformations, opens to the medical engineers new big challenges.

Scene reconstruction like structure-from-motion techniques are still not widely

used in standard medical imaging. They allow to reconstruct the surface of an object from at least two of its views obtained after a displacement of the camera viewpoint. The surface of the object is observed and then scanned, or measured. Since two views allow a minimal surface reconstruction and the process of acquisition and reconstrucion is automatic and fast, this procedure would allow a continuous acquisition and update of the surface of the observed models. The information extracted is of pure geometrical kind. Its usefulness must be compared with the one of diagnostic information. Nevertheless, up-to-date geometrical information allows medical engineers to offer to the surgeon devices with new capabilities: AR update, model patch, scene realignment or intraoperative registration. The inclusion of AR together with scene reconstruction modules does therefore define new endoscopes endowed with a higher level class of delivered services. Active AR, able to update itself from a set of observed views, is one of such new services. It could allow, for example, a bottom-up brain shift correction, the top-down approach still remaining the open MRI imaging.

Moreover, computer vision techniques are not limited to scene reconstruction techniques: active endoscopes could offer even more services. Classifiers are also other very interesting options to automatically detect and classify artifacts or tumours in diagnostic images. They could be easily integrated into endoscopic devices acting as big visual recognition databases to help surgeons in the detection of suspect tissues. The NEAR project was a step towards the development of such smart medical devices. The way to medical computer vision is now open.

Publications

M. Ciucci, L. A. Kahrs. J. Raczkowsky, R. Wirtz and H. Wörn. The introduction of augmented reality in neuroendoscopy: from surgeons' support to possible source of 3D intraoperative models. In: *MIMOS 2007, National Conference on Modeling and Simulation*, Proceedings, digital publication, Turin Italy.

M. Ciucci, L. A. Kahrs. J. Raczkowsky, R. Wirtz and H. Wörn. Setup and calibration of a navigated endoscope for virtual and augmented neurosurgery. In: *Tagungsband der 6. Jahrestagung der Deutschen Gesellschaft für Computer- und Roboterassistierte Chirurgie e.V.*, ISBN 978-3-86805-008-0, pages 299-302, 2007.

M. Ciucci, L. A. Kahrs. J. Raczkowsky, R. Wirtz and H. Wörn. A novel camera calibration model for augmented reality applications in neurosurgery. *"International Journal of Computer Assisted Radiology and Surgery*, 2008.

M. Ciucci, L. A. Kahrs. J. Raczkowsky, M.-E. Halatsch and H. Wörn. The NEAR Project: Active endoscopes in the operating room. In: *2009 IEEE VECIMS International Conference on Virtual Environments, Human-Computer Interfaces and Measurement Systems Proceedings*, pages 47-52, Hong Kong, 2009.

M. Ciucci, L. A. Kahrs. J. Raczkowsky and H. Wörn. Automatic point cloud generation using optical flow techniques for navigated endoscopy in neurosurgery. Accepted for publication in *13th International Conference on Human-Computer Interaction Proceedings*, S. Diego, 2009.

Curriculum vitae

Personal Data

Name	Matteo Ciucci
Birthdate	28.08.1975
Birthplace	Milan
Citizenship	Italian

Education

2006 - 2009	PhD research at the IPR, Karlsruhe, Germany Institute for process Control and Robotics
1994 - 2001	Studies of Theoretical Physics, Università degli Studi di Milano, Italy
1989 - 1994	Diploma di Maturità Scientifica, Liceo P. Frisi, Monza, Italy

Job Experience

2003 - 2006	Senior Developer at the CRCC informatica, Monza, Italy
2003	Research Assistant in Soft X-Rays, Department of Physics Politecnico di Milano, Italy and ESFR, Grenoble, France
2001 - 2003	Junior Developer at the CRCC informatica, Monza, Italy
2002 - 2003	Teaching Assistant for Experimental Physics, Politecnico di Milano, Italy

Bibliography

[1] A. Cuschieri. Whither minimal access surgery: Tribulations and expectations. *American Journal of Surgery*, 169(1):9–19, 1995.

[2] M. Scholz, M. Hardenack, W. Konen, B. Fricke, M. von Düring, L. Heuser, and A. G. Harders. Navigation in neuroendoscopy. *minimally Invasive Therapy and Allied Technologies*, 8(5):309–316, 1999.

[3] M. Zimmermann, R. Krishnan, A. Raabe, and V. Seifert. Robot-assisted navigated neuroendoscopy. *Neurosurgery*, 51(6):1446–1452, 2002.

[4] E. P. Westebring-van der Putten, R. H. Goossens, J.J. Jakimowicz, and J. Dankelman. Haptics in minimally invasive surgery – a review. *Minimimally Invasive Theraphy and Allied Technology*, 17(1):3–16, 2008.

[5] A. Perneczky and G. Fries. Endoscope-assisted brain surgery: Part 1-evolution, basic concept, and current technique. *Neurosurgery*, 42(2):219–224, 1998.

[6] O. Faugeras, Q-T. Luong, and T. Papadopoulou. *The Geometry of Multiple Images: The Laws That Govern The Formation of Images of A Scene and Some of Their Applications*. MIT Press, Cambridge, MA, USA, 2001.

[7] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, March 2004.

[8] Ronald Azuma, Yohan Baillot, Reinhold Behringer, Steven Feiner, Simon Julier, and Blair MacIntyre. Recent advances in augmented reality. *IEEE Computer Graphics and Applications*, 21(6):34–47, 2001.

[9] R. Shahidi, M.R. Bax, Jr. Maurer, C.R., J.A. Johnson, E.P. Wilkinson, Bai Wang, J.B. West, M.J. Citardi, K.H. Manwaring, and R. Khadem. Implementation, calibration and accuracy testing of an image-enhanced endoscopy system. *Medical Imaging, IEEE Transactions*, 21(12):1524–1535, Dec. 2002.

[10] F. Sauer, Ali Khamene, and S. Vogt. An augmented reality navigation system with a single-camera tracker: System design and needle biopsy phantom trial. In *MICCAI '02: Proceedings of the 5th International Conference on Medical Image Computing and Computer-Assisted Intervention-Part II*, pages 116–124, London, UK, 2002. Springer-Verlag.

[11] U. Bockholt, A. Bisler, M. Becker, W. Müller-Wittig, and G. Voss. Augmented reality for enhancement of endoscopic interventions, vr '03. *Proceedings of the IEEE Virtual Reality 2003*, 97, 2003.

[12] V. Lepetit and P. Fua. Monocular model-based 3d tracking of rigid objects: A survey. In *Foundations and Trends in Computer Graphics and Vision*, pages 1–89, 2005.

[13] R. U. Thoranaghatte, G. Zheng, F. Langlotz, and L.-P. Nolte. Endoscope-based hybrid navigation system for minimally invasive ventral spine surgeries. *Computer Aided Surgery*, 10(5):351–356, 2005.

[14] R. Shamir, L. Joskowicz, and Y. Shoshan. An augmented reality guidance probe and method for image-guided surgical navigation. In *International symposium on robotics and automation ISRA 2006*, august 2006.

[15] Tobias Sielhorst, Marco Feuerstein, and Nassir Navab. Advanced medical displays: A literature review of augmented reality. *J. Display Technol.*, 4(4):451–467, 2008.

[16] D. Dey, K. J. M. Gobbi, D. G. Surry, P. J. Slomka, and T. M. Peters. Mapping of endoscopic images to object surfaces via ray-traced texture mapping for image guidance in neurosurgery. In Seong K. Mun, editor, *SPIE proceedings series*, volume 3976, pages 290–300. SPIE, 2000.

[17] D. Dey, P. J. Slomka, D. G. Gobbi, and T. M. Peters. Mixed reality merging of endoscopic images and 3-d surfaces. In *MICCAI '00: Proceedings of the Third International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 796–803, London, UK, 2000. Springer-Verlag.

[18] D. Dey, D. G. Gobbi, P. J. Slomka, K. J. M. Surry, and T. M. Peters. Automatic fusion of freehand endoscopic brain images to three-dimensional surfaces: Creating stereoscopic panoramas. *IEEE Trans. Med. Imaging*, 21:23–30, 1 2002.

[19] W. Konen, M. Scholz, and S. Tombrock. The vn-project : Endoscopic image processing for neurosurgery. *Computer Aided Surgery*, 3(3):144–148, 1998.

[20] M. Scholz, K. Konen, S. Tombrock, B. Fricke, L. Adams, M. von Düring, A. Hentsch, L. Heuser, and A. G. Harders. Development of an endoscopic navigation system based on digital image processing. *Computer Aided Surgery*, 3(3):134–143, 1998.

[21] D. Bartz, O. Gürvit, D. Freudenstein, H. Schiffbauer, and J. Hoffmann. Integration of navigation, optical and virtual endoscopy in neurosurgery and oral and maxillofacial surgery. In *In 3rd Caesarium Computer Aided Medicine*, 2001.

[22] Fischer J., D. Bartz, and W. Straßer. Intuitive and lightweight user interaction for medical augmented reality. In *Proceedings of Vision, Modeling, and Visualization*, pages 375–382, 2005.

[23] A. Neubauer, S. Wolfsberger, M. T. Forster, L. Mroz, R. Wegenkittl, and Katja Bühler. Advanced virtual endoscopic pituitary surgery. *IEEE Transactions on Visualization and Computer Graphics*, 11(5):497–507, 2005.

[24] A. Fossati, M. Dimitrijevic, V. Lepetit, and P. Fua. Bridging the gap between detection and tracking for 3d monocular video-based motion capture. In *CVPR*. IEEE Computer Society, 2007.

[25] J. J. Caban and Brent W. Seales. Reconstruction and Enhancement in Monocular Laparoscopic Imagery. In *Proc. of Medicine Meets Virtual Reality*, volume 12, 2004.

[26] C. Wengert, J. M. Duff, C. Baur, G. Székely, and P. C. Cattin. Fiducial-free endoscopic vertebra referencing. In *CAOS 2007*, June 2007.

[27] T. Stehle, D. Truhn, T. Aach, C. Trautwein, and J. Tischendorf. Camera calibration for fish-eye lenses in endoscopy with an application to 3d reconstruction. In *Biomedical Imaging: From Nano to Macro, 2007. ISBI 2007. 4th IEEE International Symposium on*, pages 1176–1179, April 2007.

[28] R. A. Ketcham and W. D. Carlson. Acquisition, optimization and interpretation of x-ray computed tomographic imagery: applications to the geosciences. *Computers & Geosciences*, 27(4):381 – 400, 2001.

[29] G.L. Zeng. Nonuniform noise propagation by using the ramp filter in fan-beam computed tomography. *Medical Imaging, IEEE Transactions on*, 23(6):690–695, June 2004.

[30] W. E. Lorensen and H. E. Cline. Marching cubes: A high resolution 3d surface construction algorithm. In *SIGGRAPH '87: Proceedings of the 14th annual conference on Computer graphics and interactive techniques*, pages 163–169, New York, NY, USA, 1987. ACM.

[31] P. H. Schönemann. A generalized solution of the orthogonal procrustes problem. *Psychometrika*, 31(1):1–10, 3 1966.

[32] O. D. Faugeras and M. Hebert. The representation, recognition, and locating of 3-d objects. *Int. J. Rob. Res.*, 5(3):27–52, 1986.

[33] Berthold K. P. Horn, H. M. Hilden, and S. Negahdaripour. Closed-form solution of absolute orientation using orthonormal matrices. *J. Opt. Soc. Am. A*, 5(7):1127–1135, 1988.

[34] K. S. Arun, T. S. Huang, and S. D. Blostein. Least-squares fitting of two 3-d point sets. *IEEE Trans. Pattern Anal. Mach. Intell.*, 9(5):698–700, 1987.

[35] B. K. P. Horn. Closed-form solution of absolute orientation using unit quaternions. *J. Opt. Soc. Am. A*, 4(4):629–642, 1987.

[36] C. R. Jr. Maurer and J. M. Fitzpatrick. A review of medical image registration. In *Interactive imageguided neurosurgery*, pages 17–44, 1993.

[37] J. B. A. Maintz and M. A. Viergever. An overview of medical image registration methods. Technical report, In Symposium of the Belgian hospital physicists association (SBPH-BVZF, 1996.

[38] K. Schicho, M. Figl, R. Seemann, M. Donat, M. L. Pretterklieber, W. Birkfellner, A. Reichwein, F. Wanschitz, F. Kainberger, H. Bergmann, A. Wagner, and R. Ewers. Comparison of laser surface scanning and fiducial marker-based registration in frameless stereotaxy. technical note. *Journal of Neurosurg*, 106(4):704–9, 2007.

[39] Paul J. Besl and Neil D. McKay. A method for registration of 3-d shapes. *IEEE Trans. Pattern Anal. Mach. Intell.*, 14(2):239–256, 1992.

[40] B. Ma, E. R. Ellis, and D. J. Fleet. Spotlights: A robust method for surface-based registration in orthopedic surgery. In *Medical Image Computing and Computer-Assisted Intervention - MICCAI '99*, volume 1679, pages 936–944, 1999.

[41] R. R. Shamir, M. Freiman, L. Joskowicz, S. Spektor, and Y. Shoshan. Surface-based facial scan registration in neuronavigation procedures: a clinical study. 2009.

[42] Michael W. Walker, Lejun Shao, and Richard A. Volz. Estimating 3-d location parameters using dual number quaternions. *CVGIP: Image Underst.*, 54(3):358–367, 1991.

[43] A. Lorusso, D. W. Eggert, and R. B. Fisher. A comparison of four algorithms for estimating 3-d rigid transformations. In *BMVC '95: Proceedings of the 1995 British conference on Machine vision (Vol. 1)*, pages 237–246, Surrey, UK, UK, 1995. BMVA Press.

[44] J. M. Fitzpatrick, J. B. West, and C. R. Jr. Maurer. Predicting error in rigid-body point-based registration. *Medical Imaging, IEEE Transactions on*, 17(5):694–702, Oct. 1998.

[45] C. Y. Liu, M. Spicer, and M. L. Apuzzo. The genesis of neurosurgery and the evolution of the neurosurgical operative environment: Part ii-concepts for future development, 2003 and beyond. *Neurosurgery*, 52(1):20–35, 2003.

[46] K. W. Li, C. Nelson, I. Suk, and G. I. Jallo. Neurosurgery: past, present and future. *Neurosurgical focus*, 19, Dec. 2005.

[47] O. M. Skrinjar, D. Spencer, and J. S. Duncan. Brain shift modeling for use in neurosurgery. In *MICCAI '98: Proceedings of the First International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 641–649, London, UK, 1998. Springer-Verlag.

[48] A. Nabavi, P. Black, D. Gering, C. Westin, V. Mehta, R. Pergolizzi, M. Ferrant, S. Warfield, N. Hata, R. Schwartz, W. Wells, R. Kikinis, and F. Jolesz. Serial intraoperative magnetic resonance imaging of brain shift. 04 2001.

[49] M. M. Letteboer, P. W. Willems, M. A. Viergever, and W. J. Niessen. Brain shift estimation in image-guided neurosurgery using 3-d ultrasound. *IEEE Trans Biomed Eng.*, 52:268–276, 2005.

[50] M. Aron, G. Simon, and M.-O. Berger. Handling uncertain sensor data in vision-based camera tracking. In *Mixed and Augmented Reality, 2004. ISMAR 2004. Third IEEE and ACM International Symposium on*, pages 58–67, Nov. 2004.

[51] B. D. Allen and G. Welch. A general method for comparing the expected performance of tracking and motion capture systems. In *VRST '05: Proceedings of the ACM symposium on Virtual reality software and technology*, pages 201–210, New York, NY, USA, 2005. ACM.

[52] Martin Bauer, Michael Schlegel, Daniel Pustka, Nassir Navab, and Gudrun Klinker. Predicting and estimating the accuracy of n-occular optical tracking systems. In *ISMAR '06: Proceedings of the 2006 Fifth IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR'06)*, pages 43–51, Washington, DC, USA, 2006. IEEE Computer Society.

[53] A. D. Wiles, D. G. Thompson, and D. D. Frantz. Accuracy assessment and interpretation for optical tracking systems. *Proceedings of SPIE*, SPIE-5367:421–432, 2004.

[54] T. H. Tomkinson, J. L. Bentley, M. K. Crawford, C. J. Harkrider, D. T. Moore, and J. L. Rouke. Rigid endoscopic relay systems: a comparative study. *Applied Optics*, 35(34):6674–6683, 1996.

[55] Kazuhide Hasegawa and Yukio Sato. Endoscope system for high-speed 3d measurement. *Systems and Computers in Japan*, 32(8):30–39, 2001.

[56] W. Chenyu and B. Jaramaz. An easy calibration for oblique-viewing endoscopes. In *Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on*, pages 1424–1429, May 2008.

[57] P. Milgram and H. Takemura. Augmented reality: A class of displays on the reality-virutality continuum. *SPIE Proceedings: Telemanipulator and Telepresence Technologies*, 2351:282–292, 11 1994.

[58] P. Milgram and F. Kishino. A taxonomy of mixed reality visual displays. *IEICE Transactions on Information Systems*, E77-D(12):1321–1329, 12 1994.

[59] C. Wengert, P. C. Cattin, J. M. Duff, and G. Székely. Markerless endoscopic registration and referencing. In *Medical Image Computing and Computer-Assisted Intervention*, volume 4190, pages 816–823, october 2006.

[60] C. Wengert, L. Bossard, C. Baur, G. Székely, and P. C. Cattin. Endoscopic navigation for minimally invasive suturing. *Computed Aided Surgery*, 13:299–310, 5 2008.

[61] G. Bianchi, C. Wengert, M. Harders, P. C. Cattin, and G. Skékely. Camera-marker alignment framework and comparison with hand-eye calibration for augmented reality applications. *Mixed and Augmented Reality, IEEE / ACM International Symposium on*, 0:188–189, 2005.

[62] J. Fischer, M. Neff, D. Freudenstein, D. Bartz, and W. Straßer. Argus: Harnessing intraoperative navigation for augmented reality. In *38. DGBMT Jahrestagung Biomedizinische Technik (BMT)*, pages 42–43, September 2004.

[63] W. Konen, M. Scholz, and Tombrock S. The vn project: Endoscopic image processing for neurosurgery. *Computer Aided Surgery*, 3(3):144–148, 3 1998.

[64] C. Slama. *Manual of Photogrammetry*. American Society of Photogrammetry, 1980.

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

[66] J. Mallon and P. F. Whelan. Which pattern? biasing aspects of planar calibration patterns and detection methods. *Pattern Recognition Letters*, 28(8):921–930, June 2007.

[67] P. D. Carman. Photogrammetric errors from camera lens decentering. *Journal of the Optical Society of America*, 39:951–953, 1949.

[68] W. E. Smith, N. Vakil, and S. A. Maislin. Correction of distortion in endoscope images. *Medical Imaging, IEEE Transactions on*, 11(1):117–122, Mar 1992.

[69] R. Tsai. A versatile camera calibration technique for high-accuracy 3d machine vision metrology using off-the-shelf tv cameras and lenses. *Robotics and Automation, IEEE Journal of*, 3(4):323–344, Aug 1987.

[70] T. A. Clarke and J. G. Fryer. The development of camera calibration methods and models. *The Photogrammetric Record*, 16(91):51–66, 1998.

[71] G.-Q. Wei and S.D. Ma. A complete two-plane camera calibration method and experimental comparisons. In *Computer Vision, 1993. Proceedings., Fourth International Conference on*, pages 439–446, May 1993.

[72] R. Cipolla, T. Drummond, and D. Robertson. Camera calibration from vanishing points in images of architectural scenes, 1999.

[73] J. Salvi, X. Armangué, and J. Batlle. A comparative review of camera calibrating methods with accuracy evaluation. *Pattern Recognition*, 35(7):1617 – 1635, 2002.

[74] M. Hu, G. Dodds, Baozong Yuan, and Xiaofang Tang. Robust camera calibration with epipolar constraints. In *Signal Processing, 2004. Proceedings. ICSP '04. 2004 7th International Conference on*, volume 2, pages 1115–1118, Aug.-4 Sept. 2004.

[75] Ying Liu, Yuanxin Wu, Meiping Wu, and Xiaoping Hu. Planar vanishing points based camera calibration. In *Image and Graphics, 2004. Proceedings. Third International Conference on*, pages 460–463, Dec. 2004.

[76] Y. I. Abdel-Aziz and H. M. Karara. Direct linear transformation from comparator coordinates into object space coordinates in close-range photogrammetry. *Proceedings of the Symposium on Close-Range Photogrammetry*, pages 1–18, 1971.

[77] J. Heikkila and O. Silven. Calibration procedure for short focal length off-the-shelf ccd cameras. *Pattern Recognition, 1996., Proceedings of the 13th International Conference on*, 1:166–170, Aug 1996.

[78] K. Miyamoto. Fish eye lens. *J. Opt. Soc. Am.*, 54(8):1060–1061, 1964.

[79] J. Kannala and S. S. Brandt. A generic camera calibration method for fish-eye lenses. In *ICPR (1)*, pages 10–13, 2004.

[80] H. Hoppe, S. Daeuber, C. Kuebler, J. Raczkowsky, and H. Wörn. A new, accurate and easy to implement camera and video projector model. *Studies in Health Technology and Informatics*, pages 204–206, 2002.

[81] F. Devernay and O. Faugeras. Straight lines have to be straight. *Machine Vision and Applications*, 13:14–24, 2001.

[82] K. V. Asari, S. Kumar, and S. Radhakrishnan. A new approach for nonlinear distortion correction in endoscopic images based on least squares estimation. *IEEE Trans. on Medical Imaging*, 18(4), 1999.

[83] K. Daniilidis. Hand-eye calibration using dual quaternions. *International Journal of Robotics Research*, 18(18):286–298, 1999.

[84] R. Y. Tsai and R. K. Lenz. Real time versatile robotics hand/eye calibration using 3d machine vision. In *Robotics and Automation, 1988. Proceedings., 1988 IEEE International Conference on*, volume 1, pages 554–561, Apr 1988.

[85] J. C. K. Chou and M. Kamel. Quaternions approach to solve the kinematic equation of rotation, aaax=axab, of a sensor-mounted robotic manipulator. In *Robotics and Automation, 1988. Proceedings., 1988 IEEE International Conference on*, pages 656–662 vol.2, Apr 1988.

[86] F. C. Park and B. J. Martin. Robot sensor calibration: solving ax=xb on the euclidean group. *Robotics and Automation, IEEE Transactions on*, 10(5):717–721, Oct 1994.

[87] J. Shi and C. Tomasi. Good features to track. In *Computer Vision and Pattern Recognition, 1994. Proceedings CVPR '94., 1994 IEEE Computer Society Conference on*, pages 593–600, Jun 1994.

[88] Alper Yilmaz, Omar Javed, and Mubarak Shah. Object tracking: A survey. *ACM Comput. Surv.*, 38(4):13, 2006.

[89] C. Tomasi and T. Kanade. Detection and tracking of point features. Technical Report CMU-CS-91-132, Carnegie Mellon University, April 1991.

[90] C. Tomasi. Shape and motion from image streams under orthography: a factorization method. *International Journal of Computer Vision*, 9:137–154, 1992.

[91] L.G. de la Fraga and I. Vite-Silva. Direct 3d metric reconstruction from two views using differential evolution. In *Evolutionary Computation, 2008. CEC 2008. (IEEE World Congress on Computational Intelligence). IEEE Congress on*, pages 3266–3273, June 2008.

[92] A. Bartoli and J.-T. Lapresté. Triangulation for points on lines. *Image Vision Comput.*, 26(2):315–324, 2008.

[93] James C. Moore. Visualizing with vtk. *Linux J.*, page 5.

[94] G. Bradski. The OpenCV Library. *Dr. Dobb's Journal of Software Tools*, 2000.

[95] E. Anderson, Z. Bai, C. Bischof, S. Blackford, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney, and D. Sorensen. *LAPACK Users' Guide*. Society for Industrial and Applied Mathematics, Philadelphia, PA, third edition, 1999.

[96] G. Marti, V. Bettschart, J.-S. Billiard, and C. Baur. Hybrid method for both calibration and registration of an endoscope with an active optical tracker. In *Computer Assisted Radiology and Surgery*, 2004.

[97] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. *Proc. CVPR*, 1:511–518, 2001.

[98] C. Harris and M. Stephens. A combined corner and edge detection. In *Proceedings of The Fourth Alvey Vision Conference*, pages 147–151, 1988.