Computer Aided Medical Procedures
Prof. Dr. Nassir Navab

Dissertation

Computer Vision-Assisted Surgery: Real-Time Instrument Tracking with Machine Learning

Nicola Christin Rieke

Fakultät für Informatik
Technische Universität München

Technische Universität München
Fakultät für Informatik
Lehrstuhl für Informatikanwendungen in der Medizin

Computer Vision-Assisted Surgery: Real-Time Instrument Tracking with Machine Learning

Nicola Christin Rieke

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

Doktors der Naturwissenschaften (Dr. rer. nat.)

genehmigten Dissertation.

Vorsitzende(r): Prof. Dr. Nils Thürey

Prüfer der Dissertation: Prof. Dr. Nassir Navab
Prof. Dr. Raphael Sznitman

Die Dissertation wurde am 28.03.2018 bei der Technischen Universität München eingereicht und durch die Fakultät für Informatik am 16.09.2018 angenommen.

Nicola Christin Rieke
Computer Vision-Assisted Surgery:
Real-Time Instrument Tracking with Machine Learning
Dissertation

Technische Universität München
Fakultät für Informatik
Lehrstuhl für Informatikanwendungen in der Medizin
Boltzmannstraße 3
85748 Garching bei München

Abstract

The accurate tracking of surgical instruments in medical image sequences is a key component of various computer-assisted interventions. Determining the instrument position and monitoring its movements enables for example surgical gesture recognition, visual servoing of medical robots and interventional workflow analysis. Moreover, real-time tracking paves the way for intelligent decision support during the intervention: crucial information such as the identification of a potentially dangerous situation could be displayed close to the surgeon's visual attention area, which is usually close to the tip of the surgical tool. A further value of tracking lies in the alignment of an additional intra-operative modality to the tool movement and hence the possibility to observe in-depth instrument-tissue interaction. Such deployment can significantly reduce the burden on the surgeon and have a positive impact on the surgical outcome.

It is therefore not surprising that instrument tracking has been an important topic of research in the last decade. However, despite great advances, accurate and robust tracking of instruments in the intra-operative setting has not yet been resolved to a satisfactory extent. One of the main difficulties arises from the fact that the image data in such a setting captures only a very restricted field of view of the highly dynamic environment. Especially the non-static directional light source complicates the task by creating shadows, uneven illumination and specular reflections in the images. At the same time, a tracking algorithm for such a setting must be robust, accurate and real-time-capable. This combination of requirements and constraints poses a particularly challenging Computer Vision problem.

Prior work in this field has mainly relied on explicit modelling and just started to explore the potential of data-driven approaches. This dissertation follows the idea of learning from data and introduces several approaches that leverage machine learning techniques to overcome the aforementioned challenges. In the *feed-forward pipeline*, a two-step approach with specialized Random Forests (RF) is introduced: An intensity-based RF template tracker first limits the image search space before a gradient-based RF determines the instruments' 2D pose. Building on this dual RF, a *robust pipeline* is developed by adapting the offline model to online information and by "closing the loop" between the tracking and 2D pose estimation. Finally, a deep learning-based approach in the *end-to-end pipeline* is presented that simultaneously determines the segmentation and 2D pose of the instrument with a fully convolutional neural network. By reformulating the pose estimation task as a heatmap regression, the two objectives can leverage their spatial dependency and facilitate simultaneous learning. All presented methods achieve real-time performance and are evaluated in a cross-validation setup on *in-vivo* image sequences, demonstrating their applicability to various scenarios. The results demonstrate that machine learning-based instrument tracking has remarkable advantages with respect to the state of the art in terms of accuracy, robustness and generalization.

Zusammenfassung

Das präzise Tracking von chirurgischen Instrumenten in medizinischen Bildsequenzen stellt einen wesentlichen Bestandteil für eine Vielzahl computergestützter Eingriffe dar. So ermöglicht die Bestimmung der Instrumentenposition sowie die kontinuierliche Verfolgung ihrer Bewegungen beispielsweise eine chirurgische Gestenerkennung, Visuelles Servoing von Medizinrobotern und eine interventionelle Workflow-Analyse. Darüber hinaus ebnet das Echtzeit-Tracking den Weg für intelligente Entscheidungshilfen während des Eingriffs: wichtige Informationen wie die Identifikation einer potenziell gefährlichen Situation könnten in unmittelbarer Nähe des Aufmerksamkeitszentrums des Chirurgen angezeigt werden, das sich in der Regel dicht bei der Spitze des chirurgischen Instruments befindet. Ein weiterer Nutzen liegt in der möglichen Visualisierung einer tieferliegenden Instrumenten-Gewebe-Interaktion durch die Ausrichtung einer zusätzlichen intra-operativen Bildmodalität auf die Werkzeugbewegung. Auf diese Weise kann der Arbeitsaufwand für den Chirurgen deutlich reduziert und das Operationsergebnis positiv beeinflusst werden.

Es ist daher nicht verwunderlich, dass das präzise und robuste Tracking von chirurgischen Instrumenten seit einigen Jahren ein wichtiges Forschungsthema ist. Trotz großer Fortschritte ist diese Problemstellung im intra-operativen Bereich jedoch noch nicht zufriedenstellend gelöst. Eine der Hauptschwierigkeiten besteht darin, dass die Bilddaten in einer solchen hochdynamischen Umgebung lediglich ein sehr eingeschränktes Sichtfeld erfassen. Besonders die nicht-statische, gerichtete Lichtquelle erschwert die Aufgabe, da sie Schatten, ungleichmäßige Ausleuchtung und spiegelnde Reflexionen in den Bildern erzeugt. Gleichzeitig muss ein Tracking-Algorithmus für intraoperative Anwendungen stabil, exakt und echtzeitfähig sein. Diese Kombination von Einschränkungen und Anforderungen stellt ein besonders anspruchsvolles Problem im Bereich Computer Vision dar.

Bisherige Publikationen zu dieser Thematik beruhen hauptsächlich auf expliziter Modellierung und haben gerade erst begonnen, das Potenzial datengetriebener Ansätze zu erforschen. In dieser Dissertation wird der Ansatz untersucht, die oben genannten Herausforderungen mithilfe maschineller Lerntechniken zu meistern. Die vorgestellte *Feed-Forward-Pipeline* verwendet einen zweistufigen Ansatz mit spezialisierten Random Forests (RF): Ein intensitätsbasierter RF-Template-Tracker begrenzt zunächst den Bildsuchraum, woraufhin ein gradientenbasierter RF die 2D-Pose des Instrumentes bestimmt. Aufbauend auf diesem dualen RF wird eine *robuste Pipeline* entwickelt, indem das Offline-Modell eine Anpassung an Online-Informationen erfährt und sich somit 2D-Posenbestimmung und Tracking zusammenfügen. Schließlich wird mit der *End-to-End-Pipeline* ein deep learning-basierter Ansatz vorgestellt, welcher gleichzeitig sowohl die Segmentierung als auch die 2D Instrumentenpose mithilfe eines Fully Convolutional Neural Network ermittelt. Durch die Neuformulierung der Lagebestimmung des Instruments als Heatmap-Regression wird es beiden Zielen ermöglicht, ihre räumliche Abhängigkeit zu

nutzen und das gleichzeitige Lernen zu erleichtern. Alle vorgestellten Methoden erreichen Echtzeit-Performance und werden in einem Cross-Validierungs-Setup auf in-vivo Bildsequenzen evaluiert, um ihre Anwendbarkeit auf verschiedene Szenarien zu demonstrieren. Die Ergebnisse verdeutlichen, dass ein solches auf maschinellem Lernen beruhendes Instrumententracking bemerkenswerte Vorteile gegenüber modernsten Methoden in Bezug auf Präzision, Robustheit und Generalisierung aufweist.

Acknowledgments

The last couple of years have been an exciting journey for me and I am deeply grateful for all the support and encouragement that I have received during this time.

First of all, I wish to thank my advisor, Prof. Nassir Navab. You are an inspiring guide and mentor to me and I want to thank you for your continuous patience, motivation and incredible amount of trust that you have given over the years. I have learned far more from you than what is reflected in this dissertation. Also, I want to thank Prof. Raphael Sznitman. Your research in the area of instrument tracking has inspired many of my approaches. I really enjoyed the various discussions with you and I am very grateful that you evaluate my dissertation.

I would like to express special thanks to Dr. Federico Tombari for his guidance regarding Computer Vision and Machine Learning and to Dr. Maximilian Baust for being an amazing mentor and for the advice on how to "master a PhD". Moreover, I would like to thank the Graduate School of BioEngineering (GSB) for their support and TUM Diversity for the scholarship associated with the Laura Bassi Award which gave me freedom in my research.

Also, I am grateful to Dr. Corinna Maier-Matic and Dr. Abouzar Eslami from Carl Zeiss Meditec AG, Munich, for enabling such a great collaboration and for making it possible to carry out my research in their laboratory. And I would like to thank Prof. Mathias Maier, Prof. Chris Lohmann, Dr. Daniel Zapp, Dr. Ali Nasseri and Sabrina Bohnacker from Klinikum rechts der Isar for the many insightful discussions regarding the clinical need.

I would like to extend thanks to my colleagues and friends associated with the chair of Computer Aided Medical Procedures at TUM: David Tan, Jakob Weiss, Iro Laina, Christian Rupprecht, Josue Page Vizcaino, Hessam Roodaki, Vasileios Belagiannis, Chiara Amat di San Filippo, Mohamed Alsheakhali, Fausto Milletari, Diana Mateus, Tobias Lasser, Christoph Hennersperger, Ahmad Ahmadi, Stefanie Demirci, Julia Rackerseder, Hemal Naik, Ulrich Eck, Oliver Zettinig, Sailesh Conjeti, Benjamin Gutierrez Becker, Beatrice Demiray, Salvatore Virga, Anees Kazi, Shadi Albarqouni, Arianne Tran, Benjamin Busam and many many more. Thank you for all the discussions, the sleepless nights when we were working towards a deadline and all the fun we had in the last years. A special thanks goes to Martina Hilla!

Last but not the least, I would like to thank my German family, my Italian family and my friends outside of the CAMP chair. In particular, I am grateful to Rebecca Rittstieg for being such an amazing friend. And thank you, Marco Esposito, for believing in me and supporting me throughout the entire journey. Without you, I would not be where I am now.

Contents

Chronological List of Authored and Co-authored Publications

2018

[1] Jakob Weiss, **Nicola Rieke**, Ali M. Nasseri, Mathias Maier, Chris Lohmann, Abouzar Eslami and Nassir Navab. “Fast 5DOF Needle Tracking in iOCT”. *International Journal of Computer Assisted Radiology and Surgery (IPCAI / IJCARS), 2018, Germany*.

[2] Jakob Weiss, **Nicola Rieke**, Ali M. Nasseri, Mathias Maier, Chris Lohmann, Nassir Navab and Abouzar Eslami. “Injection Assistance via Surgical Needle Guidance using Microscope-Integrated OCT (MI-OCT)”. *Proceedings of Association for Research in Vision and Ophthalmology Annual Meeting (ARVO 2018), 2018, Honolulu, USA*.

2017

[3] Iro Laina*, **Nicola Rieke***, Christian Rupprecht, Josué Page Vizcaìno, Abouzar Eslami, Federico Tombari, and Nassir Navab. “Concurrent Segmentation and Localization for Tracking of Surgical Instruments.“ *Proceedings of the 20th International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI 2017), 2017, Quebec, Canada*.

[4] Josué Page Vizcaìno, **Nicola Rieke**, David Joseph Tan, Federico Tombari, Abouzar Eslami, and Nassir Navab. “Automatic Initialization and Failure Detection for Surgical Tool Tracking in Retinal Microsurgery”. *Proceedings of Workshop Bildverarbeitung für die Medizin (BVM 2017), 2017, Heidelberg, Germany*.

[5] **Nicola Rieke**, David Joseph Tan, Federico Tombari, Josué Page Vizcaìno, Chiara Amat di San Filippo, Abouzar Eslami, and Nassir Navab. “Abstract: Real-Time Online Adaption for Robust Instrument Tracking and Pose Estimation ”. *Proceedings of Workshop Bildverarbeitung für die Medizin (BVM 2017), 2017, Heidelberg, Germany*.

2016

[6] **Nicola Rieke**, David Joseph Tan, Federico Tombari, Josué Page Vizcaìno, Chiara Amat di San Filippo, Abouzar Eslami, and Nassir Navab. "Real-Time Online Adaption for Robust Instrument Tracking and Pose Estimation ". *Proceedings of the 19th International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI 2016), 2016, Athens, Greece*.

[7] **Nicola Rieke**, David Joseph Tan, Chiara Amat di San Filippo, Federico Tombari, Mohamed Alsheakhali, Vasileios Belagiannis, Abouzar Eslami, and Nassir Navab. "Real-time Localization of Articulated Surgical Instruments in Retinal Microsurgery". *Medical Image Analysis (Med. Image Anal.), 2016, vol. 34, pp. 82-100*.

[8] **Nicola Rieke**, Stefan Duca, Abouzar Eslami, and Nassir Navab. "Automatic iOCT Positioning During Membrane Peeling via Real-time High Resolution Surgical Forceps Tracking ". *Proceedings of International Society of Imaging in the Eye Conference, Association for Research in Vision and Ophthalmology (ARVO 2016), 2016, Seattle, USA*.

[9] Katharina Hofschen, Timo Geissler, **Nicola Rieke**, Christian Schulte zu Berge, Nassir Navab, and Stefanie Demirci. "Image Descriptors in Angiography". *Proceedings on Workshop Bildverarbeitung für die Medizin (BVM 2016), 2016, Berlin, Germany*.

2015

[10] **Nicola Rieke**, David Joseph Tan, Mohamed Alsheakhali, Federico Tombari, Chiara Amat di San Filippo, Vasileios Belagiannis, Abouzar Eslami, and Nassir Navab. "Surgical Tool Tracking and Pose Estimation in Retinal Microsurgery". *Proceedings of the 18th International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI 2015), 2015, Munich, Germany.* Young Scientist Award

2014 (Prior to doctoral study)

[11] **Nicola Rieke**, Christoph Hennersperger, Diana Mateus and Nassir Navab. "Ultrasound Interactive Segmentation with Tensor-Graph Methods". *IEEE International Symposium on Biomedical Imaging (ISBI 2014), 2014, Beijing, China*.

Part I

Introduction and Background

1 Introduction

1.1 Motivation and Main Objective

Image-guided interventions have become an integral part of modern clinical practice and can look back on a history of more than 25 years [12]. Over time, the possibilities and resources, yet also responsibilities of a surgeon have changed enormously [13]: in the *past*, a physician was considered a generalist who should be able to treat a wide range of diseases and only limited equipment was available for surgery. As a consequence, surgical interventions were usually open and invasive. With the introduction of antiseptics and anesthesia in the nineteenth century, surgical procedures on patients became more feasible. However, surgeons mainly relied on knowledge gained from medical books and their own clinical experience. The dissemination of advances in treatment was delayed due to the slow communication channels of the time and mainly took place during personal meetings of experts. With the transition into the *present*, advanced instrumentation and imaging modalities have emerged which allow less invasive interventions. In particular the development of surgical microscopes [14, 15, 16] and endoscopes [17, 18] as well as the availability of multi modal medical imaging [19, 20] and robot-assisted systems [21, 22] has enabled novel surgical treatments and revolutionized the clinical workflow. The surgeon can now base decisions and actions on a broad spectrum of devices and patient-specific information. However, this is not always an advantage: with the increase of concurrently available information, it becomes more and more difficult to extract and focus on the relevant information [23]. One of the major challenges nowadays consists in making the best use of the available data, as well as finding a balance between obtainable information and external constraints such as time, risk and cost. A further key requirement for surgeons is the mastering of novel cognitive tasks during interventions - such as mentally mapping the different information sources while maintaining the required handling precision of the intervention. So instead of being a generalist, today's physician is rather an expert in a specific medical field whose workflows require a high degree of multitasking and abstraction ability. In this context, the development of computer-based systems has been driven forward with the aim of supporting the physician to perform his tasks more safely, accurately and efficiently [13, 24].

With advances in computer science, medicine and physics, there is a growing number of complementary technologies which provide insights into the patient's anatomy and physiology. The aforementioned trend towards increasing availability of patient-specific information is likely to continue. Therefore, a consortium of leading researchers in the interdisciplinary link between medicine and computer science believe that the surgery of the *future* "will be based on automatic holistic processing of all the available data to facilitate, optimize and objectify care delivery" [25]. They envision surgical data science at the core of this progress, i.e. the extraction of valuable information from medical data to support and enhance interventional

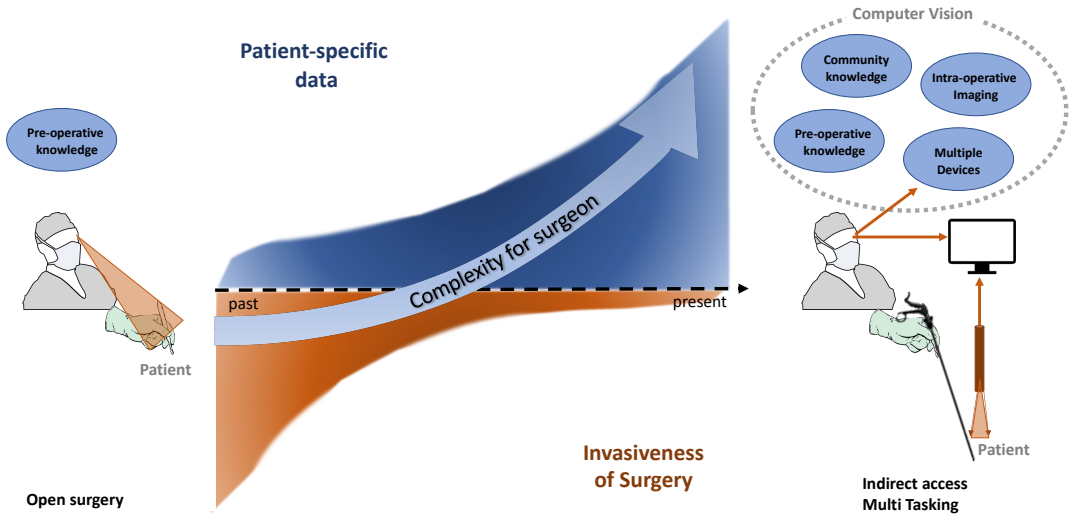


Fig. 1.1. **Motivation.** Advances in computer science, medicine and physics have led to a paradigm shift over the last decades: in the *past*, interventions were performed in an open surgery setup and the surgeon had to rely on his pre-operative knowledge. The development towards less invasive interventions and more patient-specific data provided by novel imaging modalities has enabled new surgical treatments but has led to increasing complexity for the surgeon. *Today*, the surgeon can base decisions and actions on a broad spectrum of patient-specific information and observes the surgical manipulations of the anatomical tissue indirectly. One of the main challenges is to mentally map the different information sources. Computer vision and surgical data science [25] allows to fuse this data to contextually useful information and augment the capabilities of the surgeon with computer-assisted surgical systems [26].

care. This new field of research is just beginning to emerge and has by no means reached its full potential yet.

One of the key components in this context is the visual tracking of surgical instruments in image data acquired during the intervention, which constitutes a particularly difficult and yet unsolved Computer Vision problem. The main objective of this dissertation is to explore novel approaches for **robust, real-time and accurate tracking of a surgical instrument that can overcome the challenges and fulfil the requirements of medical interventions**. To this end, we leverage the concept of learning from data and build on state-of-the-art machine learning techniques to overcome the difficulties posed by the peculiarities of interventional images. We believe that real-time instrument tracking bears great potential for improving clinical outcomes by enabling advanced surgical training as well as context-aware assistance and objective decision making during interventions. Machine learning has already led to tremendous progress in solving general Computer Vision problems on natural images. My aim is to transfer and tailor such techniques to the realm of instrument tracking in order to benefit the surgeon and hence ultimately also the patient.

1.2 Structure of this Dissertation

In the following, a brief outline of the remaining chapters of this dissertation is presented.

Chapter 2: Background. Instrument tracking can be used in a variety of medical applications, but is particularly advantageous if the operation is observed indirectly and recorded digitally. This chapter introduces the characteristics and challenges of the such medical interventions (Section 2.1). Furthermore, I will discuss how Computer Vision methods in general (Section 2.2) and instrument tracking in particular (Section 2.3) can assist the surgeon during these interventions.

Chapter 3: Surgical Instrument Tracking. This chapter contains the considerations and essentials on visual tracking, object representation and machine learning tools that build the backbone of the presented contributions. Moreover, I will explain how surgical instrument tracking methods can be evaluated and what are the particular challenges in contrast to general object tracking.

Chapter 4: Related Work. Surgical instrument tracking has been a very active field of research in the last years. This chapter will give an overview of recent approaches under the aspect of object representation.

Chapter 5: Contributions. This chapter briefly summarizes the main ideas, the advantages and the disadvantages of the proposed methods for real-time surgical instrument tracking. The original publications can be found in the Appendix A.

Chapter 6: Summary and Findings. The third part of this thesis outlines a summary and findings of our contributions.

Chapter 7: Limitations and Discussion. In the final chapter, the limitations of the proposed approaches will be explained and the possible future directions for bringing surgical instrument tracking into clinical practice will be discussed.

Appendix. Attached in appendix A are the original publications that contributed to this cumulative dissertation. Additionally, abstracts of contributions which are not covered by this thesis are listed in appendix B.

This is a publication-based dissertation and substantial parts are quoted from the respective publications [1, 2, 3, 4, 5, 6, 7, 8, 10]. Throughout this thesis, I will use the first-person plural form to highlight that many research efforts would not have been possible without such an amazing team. My personal contributions are stated for every publication in Appendix A.

2 Background

Advances in computational science have revolutionised the medical field in the last decades. Starting from image reconstruction technologies that allow a high resolution view into the patient anatomy, over automatic fusion of anatomical and functional image information all the way through intra-operative navigation to the targeted anatomy. A clinician nowadays has access to a very broad spectrum of available information for decision support and taking action. For interventional purposes, however, there is a trade-off between the amount of information and the surgeon's ability to process it in real-time. It is therefore not surprising that computer assistance has become an essential part of modern medical practice to, for example, reduce the data to contextually useful information.

This chapter will present an overview of the medical fields that are addressed in this thesis, as well as how Computer Vision methods in general and instrument tracking in particular can support the surgeon during these interventions.

2.1 Medical Field of Application

In this thesis, the main focus is on interventions in which the surgeon observes the surgical site without having a direct view onto the manipulated tissue. This mainly occurs in the following scenarios: First, due to the development towards applying less and less invasive approaches, the surgical point of access becomes smaller and does not allow an unobstructed view into the patient's body. The second case for an indirect monitoring is given when the manipulated anatomical structure itself is too small to observe with the naked eye. Instead, the surgical site is captured, magnified and visualised. The clinician manipulates the structures with special surgical instruments while observing his/her actions on the display. This kind of surgical setup poses several challenges, including:

Hand-eye coordination. The surgeon's gaze is not pointing towards his own hands anymore, but to a display. Consequently, his viewing direction is not necessarily consistent with his operating hand movements. While the surgical site may be recorded top down, the actual movements are in a different coordinate system. A horizontal movement in the real world may be observed as a tilted movement on the display, or a slight movement of μm or mm may be visualised as several cm on the display, depending on the magnification. Translating the visual input into the intended hand movement requires advanced coordination skills from the surgeon. Furthermore, the surgeon may have to physically control several devices at the same time, which complicates the coordination further.

Restricted movement and dexterity. The surgical working space within the patient's body is usually very small and only allows limited movements in terms of amplitude, degrees of freedom and operating space. In contrast to open surgery, where the surgeons can manipulate and feel the anatomy using their hands, there is no direct control or touch of tissue in this kind of surgery. All manipulations are performed with surgical instruments that are inserted into the cavity via trocars. Consequently, the instruments are restricted by at least one fixed point, which reduces the degrees of freedom and therefore the possible manipulation movements. Furthermore, tissue at risk such as important vessels and nerves might have to be avoided, which further limits possible approaches towards the targeted tissue. Although advanced surgical instruments with improved freedom of movement exist, they are not necessarily straightforward in terms of handling and operation.

Limited vision and perception. The surgeon relies on the information that is visualised on the mono or stereo 2D display. One of the main challenges here is that the clinician has to decide between a zoomed-in and a global view. As a consequence, while focusing on an anatomical structure, the global perception of the surgical site may be lost. Another main complication is that the visualisation is often affected by blur, occlusions or noise. The view of the manipulated tissue can for example be obstructed by the surgical instrument itself, detached tissue or blood, and the focus may have to be constantly adjusted to gain a clear view of the target. It is important to notice that direct perception of the manipulated tissue is not possible since the surgeon has to fully rely on the visual feedback on the screen and indirect haptic feedback via the utilised instruments. Furthermore, the depth perception is impaired due the high magnification and possibly also a monoscopic view of the surgical site.

As a consequence, the intervention is quite complex for the surgeon and requires years of training. However, the surgical view is in any case captured for visualization and can therefore be used without great effort for computer assistance during the procedure [27].

Two important surgical interventions of this field that share aforementioned characteristics are Vitreoretinal Surgery (Section 2.1.1) and Endoscopic Surgery (Section 2.1.2). The following section will describe their surgical setup, major components and particular challenges.

2.1.1 Vitreoretinal Surgery

In everyday life, we normally take our capability to see for granted. The visual acuity often only gradually decreases and many people notice it at an advanced age. In fact, however, the eye is a fragile organ and the various diseases or pathologies associated with the eye can range from benign to life-threatening [28].

From an anatomical point of view, the eye is a spherical body with a diameter of about $2.5\ cm$. Light enters from the frontal part of the eye and passes through a multitude of anatomical structures before it hits the retina. Important functional structures include the cornea, which protects the inner part of the eye, and the pupil, which regulates the amount of light that reach the eye lens. The eye cavity itself is filled with a transparent and gel-like liquid called vitreous humour that allows the light rays to pass through. The retina is located at the back of

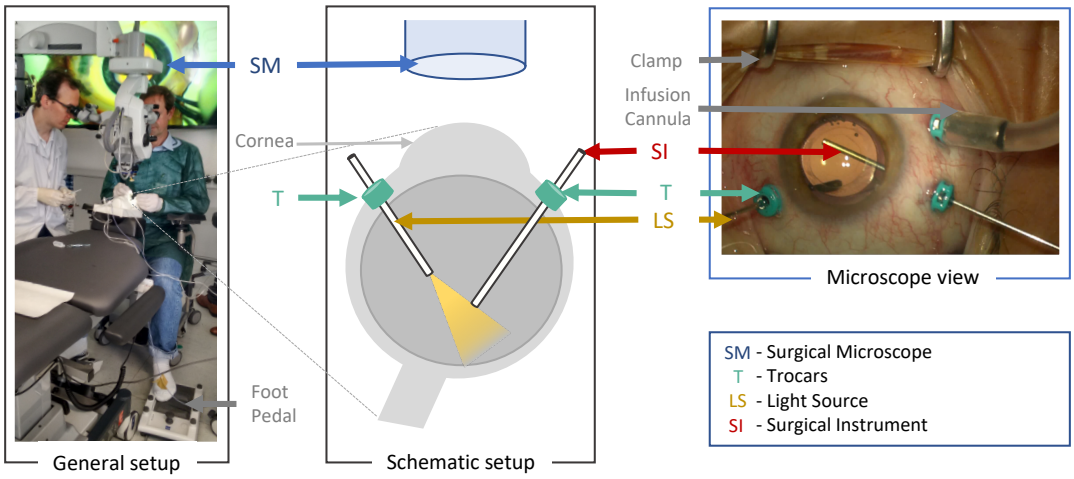


Fig. 2.1. **Surgical Setup of Vitreoretinal Surgery.** Left: In the general setup, the surgeon is seated at the head of the patient and observes his surgical movements through a microscope. The foot pedal allows to modify parameters of the microscope or position the microscope-integrated Optical Coherence Tomography. In the depicted experimental setup, a plastic head with a pig eye was used. Middle: Trocars are anchored into the sclera to provide a stable access to the eye cavity. A handheld, fibre-optic light source and a surgical instrument are inserted. Right: The image depicts the view captured by the microscope. The patient's pupil is dilated by medication and the eye is held open with a clamp to provide best possible access. An infusion cannula ensures constant intraocular pressure.

the eye and consists of multi-layered, specialised nerve tissue that converts the incident light into nerve impulses. The area in the central posterior region of the retina, through which the visual axis runs, is called macula. The cause of many significant visual impairments can be found in this anatomical area.

For example, if scar-like tissue is formed in front of the macula, the retinal layers contract and the patient's vision is severely impaired. Due to these properties, the disease is called macular pucker or epiretinal membranes (ERMs) [29]. ERMs have been associated with a vast number of ocular conditions and diseases [30], and cannot be corrected with glasses. With a prevalence of between around 4% and 29% [31, 32, 33], ERMs are one of the most common conditions currently treated by retina specialists. In order to improve the patient's vision, the membranes have to be removed from the retinal surface. This is a frequent and demanding Vitreoretinal Surgery called Membranectomy or epiretinal-membrane peeling [30]. Similar to the general setup of these kind of surgeries, it is performed by an ophthalmic surgeon, who manipulates anatomical structures within the cavity of the eye using an operating microscope and microsurgical instruments. Figure 2.1 depicts a schematic setup of a Vitreoretinal Surgery. The patient is placed on a table while his to-be operated eye is dilated by medication and held open with a clamp to ensure best possible access. Through small incisions, three trocars are anchored into the outer wall of the eye (sclera) to provide a stable access to the eye cavity. One port is used for the infusion cannula, which allows a constant intraocular pressure by pumping balanced saline solution (BSS) to replace the removed vitreous humour. The two remaining opposite trocars are for the instruments that the surgeon actively operates: a handheld, fibre-optic light source for illuminating the retinal surface from within the cavity and surgical instruments to manipulate the anatomical structures, such as intraocular forceps, diamond-dusted instruments or intraocular scissors.

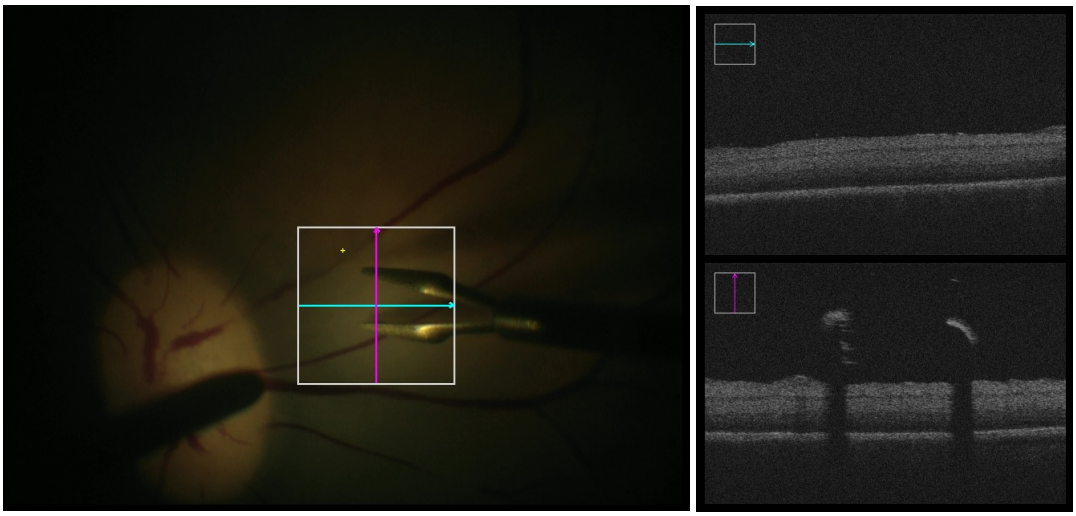


Fig. 2.2. **Surgeon's view during Vitreoretinal Surgery.** The intervention is observed through a microscope, which provides two different image sources that can be displayed next to each other, here exemplarily with porcine eyes. The surgeon has to mentally map the two image modalities. *Left*: the RGB en-face microscope image captures the direct view through the pupil of the eye. Distances to the retina are difficult to infer and the handheld light source imposes challenging illumination conditions. *Right*: the OCT image provides a cross-sectional view and therefore depth information. The image capture range is limited, as indicated by the coloured lines. The metallic instrument is opaque and occludes subjacent structures.

During a membrane peeling procedure, the surgeon first removes the vitreous humour from the eye during the so-called vitrectomy and then grasps and peels away the membrane from the retinal surface [34]. To that end, a suitable starting point such as a fold in the membrane has to be found which allows to grasp the membrane without injuring retinal structures by accidental intrusion with the instrument. It has been shown that the additional peeling of the internal limiting membrane (ILM) may result in reduced ERM recurrence [35]. The ILM is the structural interface between the retina and the vitreous with a thickness of only $1.5\mu m$ in the foveal area [36]. To avoid damaging the retinal tissue, the number of grasps should be minimal and the membrane should be peeled in a circular movement parallel to the retinal surface [37]. These maneuvers require an accuracy in the order of a few micrometers and cannot be observed with the naked eye. Instead, a microscope is placed above the patient's eye, which magnifies the top-down view through the eye lens and visualizes the surgical scene on a stereo ocular for the surgeon. Recently, an additional imaging modality was integrated into the microscope that complements the microscope en-face view with cross-sectional images (Figure 2.2): optical coherence tomography (OCT) [38]. It is a non-invasive imaging technique with near-infrared light and is based on low coherence interferometry. This means that for example sub-retinal structure information can be computed by measuring the magnitude and echo-time delay of backscattered light. The main components are a light source, a beam splitter, a reference mirror and a photo detector or spectrometer. Cross-sectional images, also called B-Scans, are reconstructed by transversely scanning the incident optical beam and measuring the echo time delay of the axial scans, also called A-Scans [39]. Originally, it was designed for diagnostic purposes and was first commercially available to surgeons in the year 2014 [40]. Its benefit in current and future ophthalmic interventions has been clearly stated by various surgeons and research groups [41, 42, 43].

However, in the currently available devices the surgeons are presented with the information of the two imaging modalities side-by-side and not combined, as depicted in Figure 2.2. Although they depict the same anatomy, the surgeon has to identify the exact spatial correlations himself by switching between the views. Another important complication is that the metallic surgical instruments are opaque for the OCT due to the underlying physics and cast shadows on the subjacent anatomical structures [44]. There have been advances in developing OCT-compatible surgical instruments, but in current practice the technology is limited to a "stop and image" approach [45]. For creating a cross-sectional view the surgeon usually interrupts the surgical manipulation and removes the surgical instruments from the respective area to eliminate the occlusions induced by the instrument. This, together with the fact that the spatial scan range of the OCT is limited, implies that the visualization of real tissue-instrument interaction with OCT is currently impossible: the surgeon would have to manually position the OCT scan location with a foot pedal close to the instrument at a correct angle while maintaining the required micro-precision in handling the instruments. In addition to aforementioned problems and the challenges described in Section 2.1, the surgeon has no or only very weak tactile feedback during the micron-scale manoeuvres [46]. The manipulated anatomical structures are so delicate that the surgeon mainly has to rely on the provided visual feedback and his experience. At the same time, the illumination of retinal structures is limited by the risk of iatrogenic phototoxicity [47] and the high magnification leads to a challenging depth perception. Since the surgical instruments are hand-held instruments, the surgeon must also ensure highest handling accuracy and absolute control over unintentional movements such as hand tremors while operating in a fairly rigid positioning for long periods of time. All this together makes Vitreoretinal Surgery extremely difficult to master.

2.1.2 Endoscopic Surgery

Endoscopic surgery refers to minimally invasive surgical techniques performed with surgical instruments and endoscopes. Depending on the targeted anatomy, the interventions are also known under different names. For example: minimally invasive procedures in the pelvic and abdominal area are called laparoscopic surgeries and operations within the thorax are thoracoscopic surgeries. The technology has become increasingly popular in the last years and has even completely replaced traditional and established techniques in many medical applications [48]. Emerging around 1980, laparoscopy has been one of the first types of endoscopic surgeries [49]. Nowadays, the minimally invasive approach is utilised for many different diseases, including pancreatic cancer, abdominal aortic aneurysms, gallbladder cancer, lung tumors, gynecological cancers, kidney disorders and prostate cancer.

In traditional open surgery, large incisions allow the surgeon to see and manipulate the target tissue directly. Although this is probably the most intuitive and direct way of handling the targeted tissue, it has many medical and cosmetic disadvantages due to the size of the required incision and the contact of tissue with air. The community therefore developed less and less invasive approaches that built the basis for current endoscopic surgery. By minimizing the incisions and introducing ports to insert instruments into the patients body, the access trauma and the blood loss was significantly reduced [50]. Furthermore, the minimally invasive approach lowers the risk of infection and the size of visible scar tissue. As a result, the patient suffers less pain and can be discharged from the hospital earlier. It also enables the use of

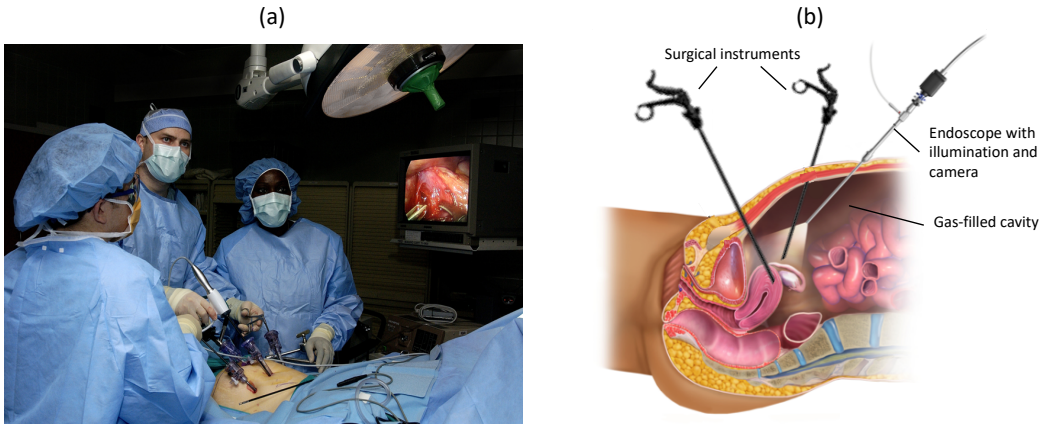


Fig. 2.3. **Endoscopic Surgery.** (a) Example of minimally invasive surgery in the abdomen around 2006. The surgeons observe their surgical actions on a display while manipulating surgical instruments. [Figure under common licence, link]. (b) Schematic setup of endoscopic surgery. The cavity is inflated with a sterile gas to create more space for the endoscope and the surgical instruments. [Figure by Blausen.com staff (2014). "Medical gallery of Blausen Medical 2014". WikiJournal of Medicine 1 (2). DOI:10.15347/wjm/2014.010. ISSN 2002-4436. link, Added and modified content]

previously unrealizable medical treatments and enlarges the potential patient group to older people or infants.

The usual surgical setup for endoscopic surgery is as follows: The anaesthetised patient is placed on an operating table in such a way that the surgical site is freely accessible. In some surgeries the patient is additionally positioned in a horizontally inclined position so that the soft tissue shifts out the target area. Trocars are anchored in the patient's skin and allow access to the surgical cavity. In order to increase the available workspace, the cavity is inflated with a sterile gas. This procedure is also known as insufflation. The devices used to observe and manipulate the tissue are inserted via the trocar tubes: one port is utilized for inserting the endoscope, which includes a Charge-Coupled Device (CCD) and a light source, and the other one is for the surgical manipulators. In conventional minimally invasive surgery, one surgeon operates the surgical working instruments and the assistant orients the endoscopic camera to the surgical site. Consequently, the surgeon usually has no direct control over the visual information that is forwarded from the laparoscope to an often monocular display next to the patient in the operation theatre. A typical endoscopic surgery setup is depicted in Figure 2.3. Alternative technologies reduce the access trauma even further by using natural openings like mouth, nostrils or vagina, as in the natural orifice transluminal endoscopic surgery (NOTES) [51], or requiring only one incision, as in the single port surgery [52]. Robotic systems, such as the da Vinci (Intuitive Surgical, Sunnyvale, CA) aim at supporting the surgeon via a master-slave concept. The manipulation of the instruments is remotely controlled by the surgeons from a console (master) and mapped via computer software into actual movements of the robotic arms (slave) that enter the patient's body.

What these approaches have in common is the fact that the surgeon can only observe the operation indirectly. As already mentioned in the introduction of this chapter (2.1), this leads to problems such as difficult hand-eye coordination, reduced navigation ability and limited perception. In contrast to Vitreoretinal Surgery, the surgeon does feel the applied

forces when manipulating the targeted anatomy. However, this haptic feedback is only indirect via the kinesthetic response of the surgical instruments and not as direct as in open surgery. Furthermore, the trocars define the available surgical workspace in the common endoscopic surgery, which may reduce the freedom of movement or result in an undesirable point of view on the anatomy when they are badly placed. Another major complication is that the surgeon is no longer able to see the entire anatomy as the endoscopic camera has a very limited field of view. Consequently, the surgeon looses the global perception of the surgical site and may for example not be able to see a bleeding in an area that is not captured by the endoscopic camera. Critical anatomy such as major vessels may also be hidden behind visible tissue and the view may further be impaired by surgical side effects such as smoke and blood or the surgical instruments itself. Furthermore, the 3D information of the patient anatomy is usually projected to a 2D display and the distance between lens and observed anatomy is small. As a result, the visualization cannot reflect the 3D structure of the anatomy and the depth perception is extremely challenging.

2.2 Computer Vision to Assist Surgeons

The enormous advances in Computer Vision (CV) research in recent years have led to a variety of mature and robust CV algorithms. At the same time, the digitalization of medical information has been pushed forward, so that today a lot of data is stored and processed digitally. CV algorithms can exploit this image data and enhance or extract abstract information to support the physician in his/her actions and decisions. In contrast to hardware-based solutions, the costs of deploying, developing and distributing software-based solutions are lower. Another major advantage is that CV solutions are mainly designed to support rather than replace sophisticated surgical routines. Consequently, the surgeon and therefore also the patient can benefit from improvements at an early stage.

The interventions addressed in this dissertation are already recorded in a video stream in current clinical practice. It therefore seems to be a “natural direction” to leverage the power of Computer Vision algorithms to overcome many challenges of these interventions [24]. The various possibilities range from low-level processing, such as noise filtering or contrast enhancement before visualizing the image data, up to mid- or high-level processing, e.g. understanding the video content in terms of object segmentation or surgical action recognition.

Computer Vision-based solutions may assist the surgeon in all operation steps:
In the *pre-operative* phase a 3D reconstruction of the patient’s anatomy model can help to accurately develop a patient-specific surgery plan. In the context of endoscopic surgery, this facilitates for example the estimation of an optimal trajectory to approach the targeted tissue without harming critical structures on the way such as nerves or major vessels. In Vitreoretinal Surgery, this detailed trajectory planning is not as crucial, as the eye ball is filled with the insensitive vitreous humour. However, a 2D segmentation of the anatomical layers [53, 54] in the high quality diagnostic OCT scans may help to identify pathological or critical structures that have to be addressed during the intervention. Creating a map of elevated epiretinal membrane could help identify suitable grasping points for membrane peeling [55].
The major potential of Computer Vision-based assistance is probably during the intervention itself (*intra-operative*). Early concepts mainly focused on providing pre-operative data during

interventions, such as additional information acquired by a different modality or predetermined models. An early work by Fleming *et al.* [56] registered preoperative OCT images to inraoperative microscope images. At this time, the intraoperative OCT was not available. But even with the new generation of surgical microscopes, the registration would be beneficial because the image quality of preoperative OCT is usually superior and covers a broader spatial area than the interaoperative version. In this way, annotated and processed pre-operative data could also be made available to the surgeon during the intervention, such as the map of elevated epiretinal membrane for safer membrane peeling. In endoscopic surgery, the visual information from the endoscope can be enhanced by pre-operative data such as Computed Tomography or Magnet Resonance Imaging [57]. In combination with augmented reality, this would allow the surgeon to see through occluded anatomy and directly see the planned cutting trajectories on the anatomy without mentally matching the different data sources. Also the problem of the limited field-of-view can be addressed with Computer Vision algorithms, both in Endoscopic Surgery [58] and Vitreoretinal Surgery [59]. The technique of video mosaicking enlarges the field of view of the surgeon by combining the image information of single frames into a global view. This gives surgeons a better global orientation and perception of the surgical site. Other possible applications include multispectral illumination [60] and image denoising [61] for improving the visualized image information. Newer concepts of computer-assisted interventions focus on context-awareness and abstract understanding of surgical procedures. This includes the automatic detection and classification of critical objects such as vessels or organs, as well as the recognition of surgical gestures. Risk situations can thereby be identified early and avoided. Another interesting direction is the 3D sensing from a single or stereo optical image. Estimating a partial 3D model of the observed anatomy facilitates the depth perception for the surgeon [24]. Especially in combination with augmented reality [62] this can be an intelligent, assistive system during the intervention.
In the *post-operative* phase, the hours of video footage of the surgeries are stored and analyzed. Instead of working through the large amount of data by hand, Computer Vision can help to classify and analyze the videos. For evaluating the surgical performance for example, the surgical gestures can be identified and compared to the expected surgical performance. Furthermore, surgical phases could be extracted that are useful for surgeon training.

Unfortunately, the application of Computer Vision algorithms is not straightforward. In contrast to the usual in- or outdoor scenario that is investigated by traditional Computer Vision algorithms, the surgical scene does not provide many reliable natural landmarks. In particular, the fact that the illumination is a directional, hand-held light source renders many methods inapplicable because they are based on the assumption of a constant light source. This difficulty will be discussed in a later chapter (3.5).

2.3 Potential Impact of Instrument Tracking

One of the key components for many computer-assisted approaches mentioned in the previous section is the real-time tracking of the utilized instrument. Examples of surgical instruments are depicted in Figure 2.4. Since the surgeon controls the surgical tool himself and knows his movements, the absolute position or the tracking information of the instrument is of course not of direct interest to him. The real value of instrument tracking lies in interpreting the instrument as an extension of the surgeon's hand. Based on a good understanding of the

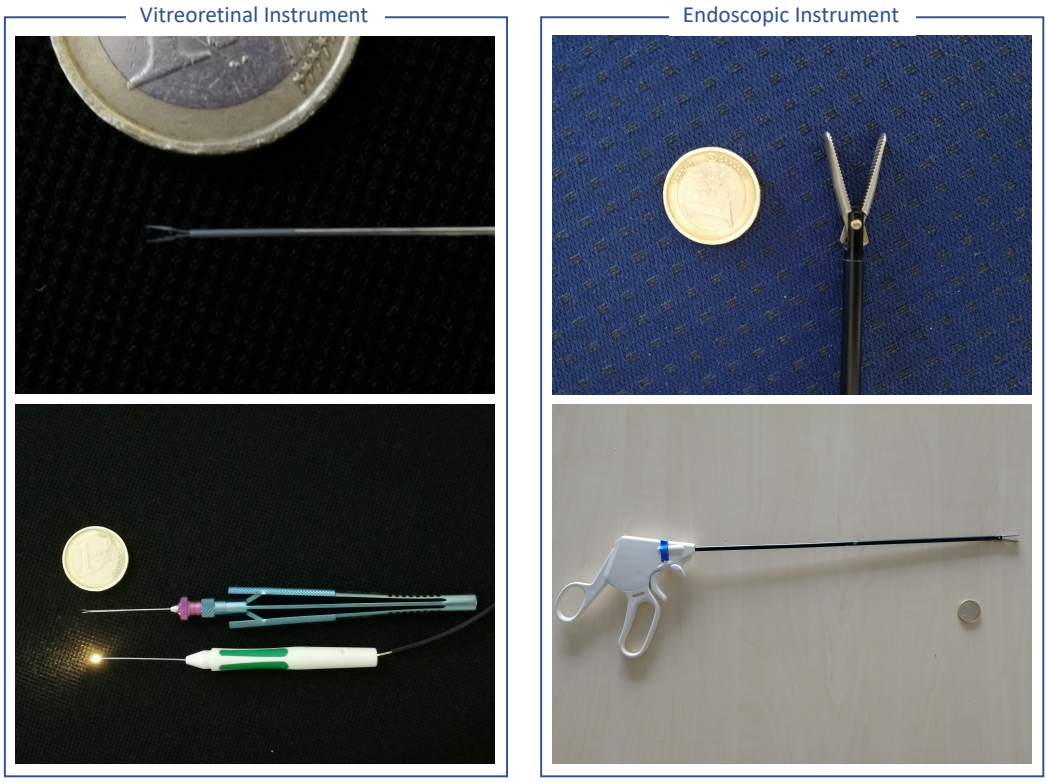


Fig. 2.4. **Surgical instruments.** The size and mechanical manipulation of surgical instruments differs considerably. In many cases, however, the instrument can be modelled as an articulated object. For vitreoretinal surgery, also the handheld light source is depicted.

physical relationship between the surgical instrument, the imaging sensors and the anatomy, it would for example be conceivable to synthesize a haptic feedback for the surgeon. The surgeon could virtually feel again the tissue through the surgical instrument. For this, however, it is necessary to know the exact position of the instrument relative to the tissue, as well as the tissue properties. In robotic systems, instrument tracking can even enable semi-automatic or automatic motion guidance towards desired anatomical targets. Together with the robotic compensation of hand tremor, this would have a major impact in the field of cell-based and gene therapy in Vitreoretinal Surgery: as discussed by MacLaren *et al.* [63], one of the main difficulties here is the safe positioning and delivery of drugs to the target area. But instrument tracking can also be used for the opposite intention: a tracked instrument may ensure a safety distance to a pre-defined vulnerable anatomical structure and thereby contribute to the patient's safety.

Understanding the instrument position and orientation can also serve as basis for high-level assessment such as task recognition [64, 65] or surgical phase recognition [66, 67]. Suturing or membrane peeling, for example, are characterised by a particular movement of the instrument. Utilizing the trajectory of surgical gestures could simplify the recognition. The activity and presence of instruments are already commonly used for surgical workflow analysis [66, 68, 69, 70]. Surgical gestures could also be beneficial here as the general, complication-free surgical procedure is often predefined. Following the same line of argumentation, instrument tracking can be employed for post-operative skill assessment [71, 72]. Since motion parameters of the

instrument such as trajectory length and occurrence of movement are directly related to the surgeon's expert level [71], instrument tracking could serve as objective parameter to measure or evaluate the performance, in terms of number of grasps or attempts for tissue cutting.

Instrument tracking could also pave the way for advanced augmented reality applications [62]. Since the shaft of the instrument is typically not of interest to the surgeon and already obstructs the view on the scene, it could be used as a suitable region for a graphical overlay of additional information. Another possibility is to visualize additional information close to the instrument manipulator, which is usually close to the surgeon's center of attention. The geometric information inferred from the instrument orientation, on the other hand, could give hints for creating perceptually correct visualization.

While the above-mentioned benefits apply to both Vitreoretinal and Endoscopic Surgery, there are also operation-specific advantages. As mentioned in Section 2.1.1, the usage of the intraoperative OCT in Vitreoretinal Surgery is currently limited to a "stop and image" approach. Ehlers *et al* [45] clearly state that "automated tracking is needed to minimise surgeon demand on OCT positioning during surgical manipulation." By knowing the spatial location of the instrument tips for example, the OCT scanning orientation and position can be automatically optimized in a way that there are no occluding shadowing artefacts and instrument-tissue interaction is visible in the cross-sectional view [8]. This would allow to understand the surgeon's interaction with tissue surfaces and even estimate the physical distance between the surgical instrument and the retina [73]. In endoscopic surgery, instrument tracking can for example guide the surgeon along an optimal path with minimal harm to sensitive tissue such as nerves according to a pre-operatively planned trajectory. For this purpose, the current position of the instrument as well as the visible anatomy must be registered to pre-operative data. Another important application for endoscopic surgery lies in visual servoing of robotic systems such as the daVinci: in a rigid setup, the instrument position can be estimated by the forward kinematics of the robot. Due to tendon-based connections of the arms, however, this projection is distorted and noisy. By tracking the instrument joints in the image directly, the relation can be recovered.

Part II

Methodology

3 Surgical Instrument Tracking

In order to achieve robust and accurate tracking of the surgical instrument, we will build on techniques from Computer Vision and Machine Learning. Since this is a publication-based dissertation, the chapter is not intended to give a comprehensive overview of existing methods, but rather to provide background and context to the presented contributions. This includes the basic concepts of visual tracking (Section 3.1), the possible representation of the tracked instrument (Section 3.2) and how machine learning algorithms can employed in this context (Section 3.3). Finally, I will explain how the performance of instrument tracking methods can be evaluated (Section 3.4) and what the difficulties and requirements are (Section 3.5).

3.1 Visual Tracking

The aim of object tracking is to locate a predefined target over a period of time. In general, there are several approaches for this task in the field of medical interventions, such as electromagnetic tracking [75], which is based on magnetic fields and triangulation, and acoustical tracking, which measures the time of flight of acoustic signals. Both techniques are not optimal for the medical applications addressed in this dissertation (see Section 2.1), as they may interfere with the devices used during the surgery and do not provide the required accuracy. An alternative approach is mechanical tracking, using the mechanical links of a surgical robot or a passive arm. If the tracked object is connected to the device and the kinematic chain is known, the spatial position of the object can be derived. However, this requires an exact calibration and is only applicable in surgeries with mechanical support (e.g. robot-assisted surgeries).

We aim at the general applicability of instrument tracking in aforementioned surgeries without the need of additional hardware and take advantage of the fact that the surgery is already digitally captured. Visual tracking requires image sequences as input and can therefore be

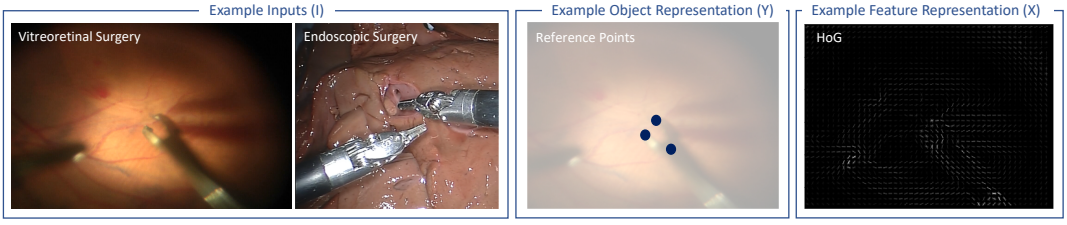


Fig. 3.1. **Elements of visual tracking.** Based on an image sequence the aim of visual tracking is to determine the instrument location for every frame. The tracked surgical tool is defined by an object representation Y. Marker-less tracking methods do not introduce artificial markers, but detect the instrument by its natural characteristics, which can represented by feature representation X. Endoscopic Surgery Images are from the Endovis Challenge 2015 [74].

integrated without great effort. It thereby allows measurements directly in the reference frame of the observing camera (endoscope or microscope) and does not need complex extrinsic calibrations for visualization applications, such as mentioned in Section 2.3. As defined by Forsyth and Ponce [76], visual tracking is "the problem of generating an inference about the motion of an object given a sequence of images". Common application domains include autonomous car driving, surveillance and sports. In our case, visual tracking corresponds to locating the surgical instrument in a sequence of endoscopic or microscope images.

Visual tracking approaches can be roughly divided into temporal tracking and tracking by detection: a temporal tracker associates information from previous frames with information from the current frame and thereby simplifies the tracking problem to a location update; a tracker based on detection only uses the information of the current frame to infer the location of the object. Furthermore, algorithms can be distinguished as either marker-based or marker-less. A visual marker can for example be an optical square marker or a predefined distinctive pattern that is attached to the object. Since we do not want to modify the surgical instrument, we focus on marker-less tracking methods. This means that we rely on the local, natural features that can characterize the instrument. For a more detailed overview on the general concepts, please refer to the survey by Yilmaz *et al.* [77].

Generally speaking, we aim at determining the instrument location, defined by an object representation Y, for frame t given an image or a sequence of images:

$$Y_t = f(\Phi\left(\{I_s\}_{s=t_1}^{t_2}\right)) = f(X_t), \tag{3.1}$$

where f is a function that maps the observations Φ from a set of images $\{I_s\}_{s=t_1}^{t_2}$ to the object representation Y. In case of tracking by detection the set of images contains only the current frame $\{I_s\}_{s=t}^{t} = I_t$, while for a temporal tracker one or several previous frames are considered, e.g. $\{I_s\}_{s=t-2}^{t} = \{I_{t-2}, I_{t-1}, I_t\}$. It should be noted that the choice of observation representation (also called feature representation) Φ and the object representation Y are highly application dependent. The feature representation consists of visual features for the marker-less tracker, such as for example Scale-Invariant Feature Transforms (SIFT) [78], Histogram of oriented Gradients [79] or the RGB value itself. Optimally, they are chosen in such a way that they are unique and the instrument can be easily distinguished from the background. The object representation is the final output of the tracker and can range from a rough localization to a precise outline of the object, will be presented in the next section.

3.2 Object Representation

The typical surgical instrument is a metallic, quite rigid object that can be represented by its 2D shape and appearance in the image. The required level of detail of this representation Y, however, is application dependent. Figure 3.2 depicts some examples that are inspired from [77]. For applications that require only basic understanding of the instrument movement, a centroid ($Y \in \mathbb{R}^2$) or a bounding box around the instrument is often sufficient. These geometric representatives already indicate the presence of the instrument and allow to calculate a rough trajectory. The bounding box would also simplify a subsequent classification of the instrument and therefore surgical action recognition. However, if the pixels occupied by the surgical instrument are for example intended as graphical overlay for augmented reality

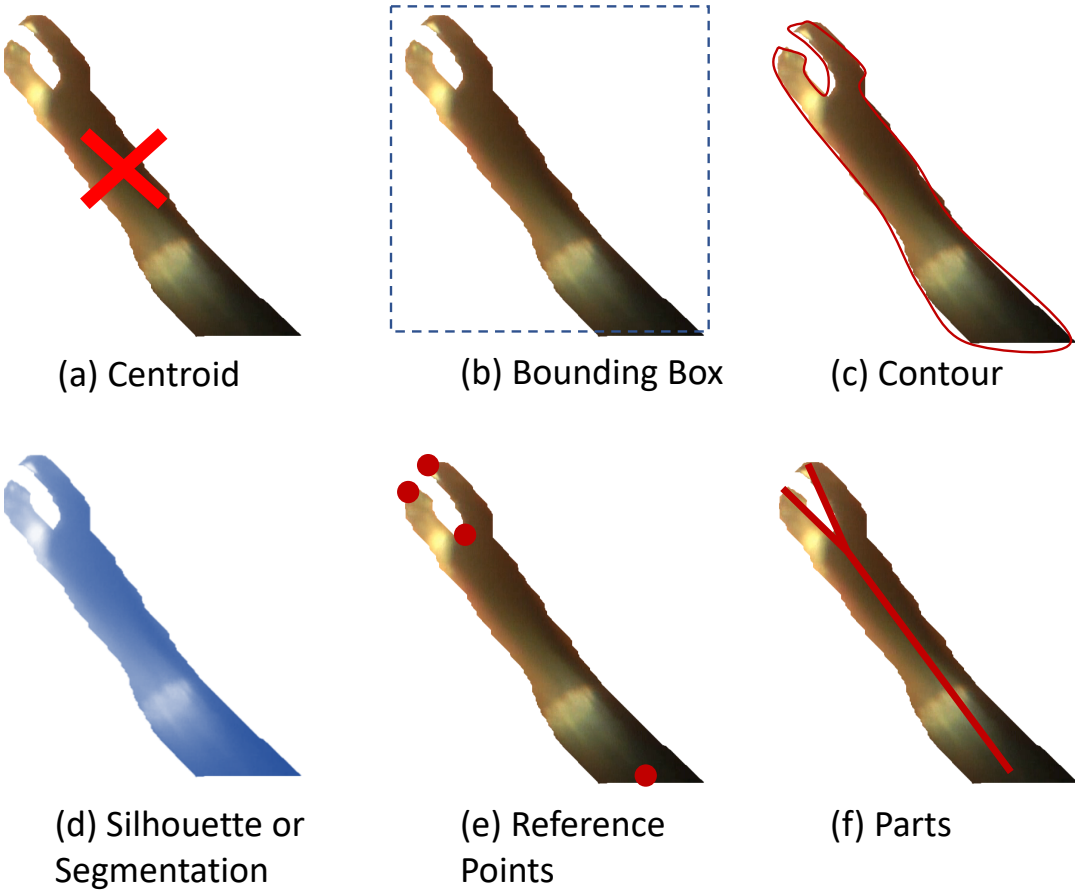


Fig. 3.2. **2D Object Representations.** The surgical tool can be tracked in terms of different object representations.

applications, such a coarse localization is not sufficient. In this case, it is more helpful if the image is divided into two disjoint sets: the 2D boundary of the instrument is called *contour* of the object ($Y \in \mathbb{R}^{n\times 2}$) and the region inside is the *silhouette* or *segmentation* ($Y \in \mathbb{R}^{w\times h\times c}$, where c denotes the number of labels and w,h denote the image width and height). In this way, every pixel of the image can be assigned to either instrument or background. The segmentation can also be used as intermediate representation and help to infer the 3D pose of the instrument, if the CAD model of the instrument is known [80]. For analysing surgical gestures however, the segmentation and the global instrument movement are not distinctive representations. A grasping gesture for example is characterized by opening and closing motion of the forceps. In this case, it is beneficial to define reference points ($Y \in \mathbb{R}^{n\times 2}$) such as the instrument tips as tracking objective. This also allows to interpret the instrument as an articulated object with parts that are connected by the reference points and paves the way for advanced applications including instrument-tissue interactions, as discussed in Section 2.3.

3.3 Learning from Data

There are numerous approaches to model a tracking function f that maps from image or feature space to the object representation as defined in equation 3.1. An overview of established methods is given in the survey by Yilmaz *et al.* [77]. In recent years, some of the most

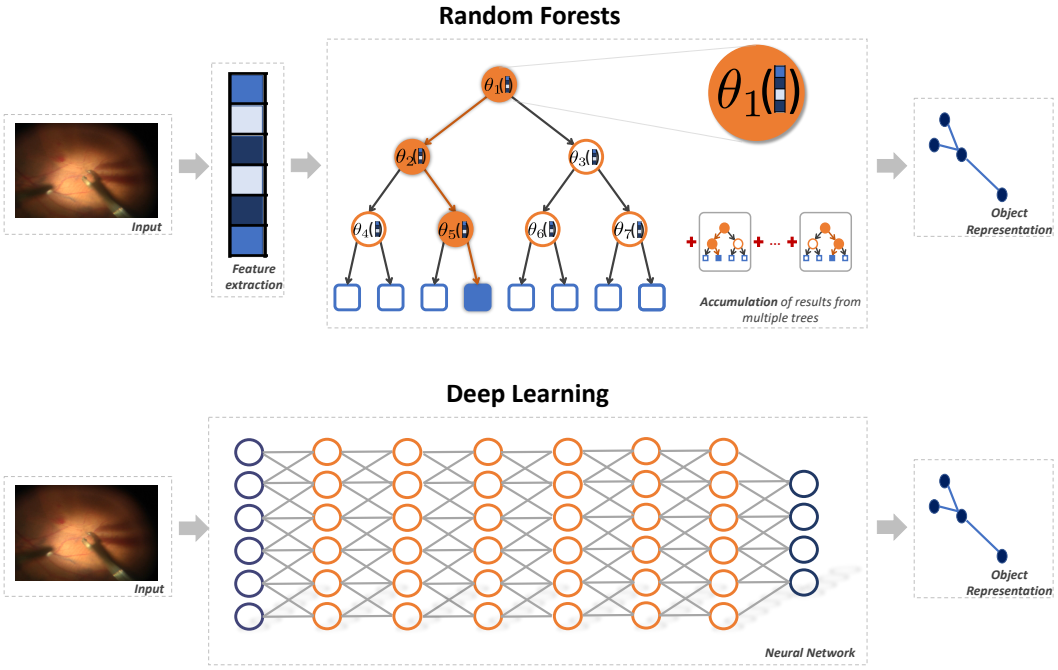


Fig. 3.3. **Learning strategies.** A Random Forest is an ensemble of independent decisions trees and addresses the problem by partitioning the input space using a set of binary decisions. These splitting functions are revisiting the input space and do not modify it. Consequently, Random Forest relies on a suitable choice for feature representation. Deep Learning is designed to learn representations inherently by abstracting image responses using a composition of nonlinear functions. Usually only the input layer has access to the image while the subsequent layers receive the resulting activations.

successful methods have been based on machine learning: it allows to statistically estimate complicated functions by building on knowledge from a set of observations. A machine learning system can thus learn from data and subsequently act as a predictive model, as opposed to rule-based systems. Depending on the availability and use of ground truth data in these observations, we can distinguish between *supervised*, *semi-supervised* and *unsupervised* methods. In the latter two cases only few or no data is labeled, respectively. In a *supervised learning framework*, we can build on a training set $\{(\mathbf{X}_d, \mathbf{Y}_d)\}_{d=1}^{n_d} = \{(\mathbf{X}_1, \mathbf{Y}_1), (\mathbf{X}_2, \mathbf{Y}_2), \ldots, (\mathbf{X}_{n_d}, \mathbf{Y}_{n_d})\}$ with available ground truth to find a prediction function f with

$$\mathbf{Y} = f(\mathbf{X}; \mathbf{w}) \quad \forall (\mathbf{X}, \mathbf{Y}) \in \{(\mathbf{X}_d, \mathbf{Y}_d)\}_{d=1}^{n_d}, \tag{3.2}$$

where $\mathbf{w}$ are the function parameters which can be learned. The prediction function is not perfect and can only be approximated, so that $f(\mathbf{X}; \mathbf{w}) = \hat{Y} \approx \mathbf{Y}$. To that end, the function parameters are optimized during training using the labeled training set according to a predefined objective function. In the *testing phase*, the trained function model can be employed for predicting $\mathbf{Y}$ for unseen input samples $\mathbf{X}$. In the following two subsections, I will summarize two important machine learning methods on which our contributions here are based. Both techniques can be represented by a graph, but differ fundamentally in their learning strategy (Figure 3.3): Random Forest (Section 3.3.1) addresses the problem by partitioning the input space using a set of binary decisions and solving the task in the created subspaces. It therefore relies on the suitable choice for feature representation Φ.

Deep Learning (Section 3.3.2) on the other hand is designed to learn suitable representations inherently by abstracting image responses using a composition of nonlinear functions.

3.3.1 Random Forest

Finding a suitable function f for the full feature space X is a complex task. Instead of solving it explicitly, the Random Forest approximates a solution by accumulating votes from "weak" learners that divide the problems into smaller problems. In other words, a Random Forest consists of an ensemble of t independent decisions trees $F = \{T_1, \cdots, T_t\}$ which can predict a discrete label (*classification*) or continuous values (*regression*) using a "divide" and "conquer" strategy. Given a training set $\{(\mathbf{X}_d, \mathbf{Y}_d)\}_{d=1}^{n_d}$, each tree T of a forest F attempts to divide the observations into partitions P_t by using a series of simple decisions in a hierarchical manner. A tree is composed of nodes that can either be a branch with two children (i.e., left and right) or a terminal node. The node at the top of a tree is called root, the nodes at a branch are called decision nodes and the terminal nodes are called leaves. An observation X can traverse the branches of the tree starting from the root. At each decision node i, it is sent either to the left or right child by the learned splitting function f_i, until it reaches a leaf.

During *training*, we split the data into two subsets at each node so that they pass down to its children (P_l, P_r), based on a splitting criterion θ. The splitting criterion is a pool of random tests and aims at finding the best split of the set at each step. In our contributions, we employ the information gain as objective function for evaluating the best split, which is given by

$$g(\theta) = H(P_n) - \sum_{i \in \{l,r\}} \frac{|P_i(\theta)|}{|P_n|} H(P_i(\theta)) \ , \tag{3.3}$$

where P_n is the set of samples that reached the node n, $|P|$ is the number of samples in the set P and $H(\cdot)$ evaluates the randomness of P. For regression tasks, $H(\cdot)$ can be estimated by standard deviation of the multi-variable Gaussian distribution. Starting from the root node, the dataset is iteratively split into two subsets and passed down to the node's children until one of the following stopping criteria is true:

1. the maximum depth of tree is reached;
2. the number of samples that reached the node is insufficient to split; or,
3. the information gain of the best split is too small.

Then, the leaf stores the distribution of the parameters of $\mathbf{Y}$ that typically employ a normal distribution with its mean and standard deviation. As a result of training, each branch of the tree stores the parameters of the splitting function with respect to the input $\mathbf{X}$ while each leaf stores a distribution of the output $\mathbf{Y}$. To enforce the independence of the trees in the forest, each random tree selects a random subset of elements in $\mathbf{X}$ or a random subset of the learning dataset.

During *testing*, a new sample of $\mathbf{X}$ traverses the tree. At each branch, it moves to the left or right child node depending on the learned splitting function, eventually ending up at a leaf node which contains the prediction to be associated with such sample. Finally, the

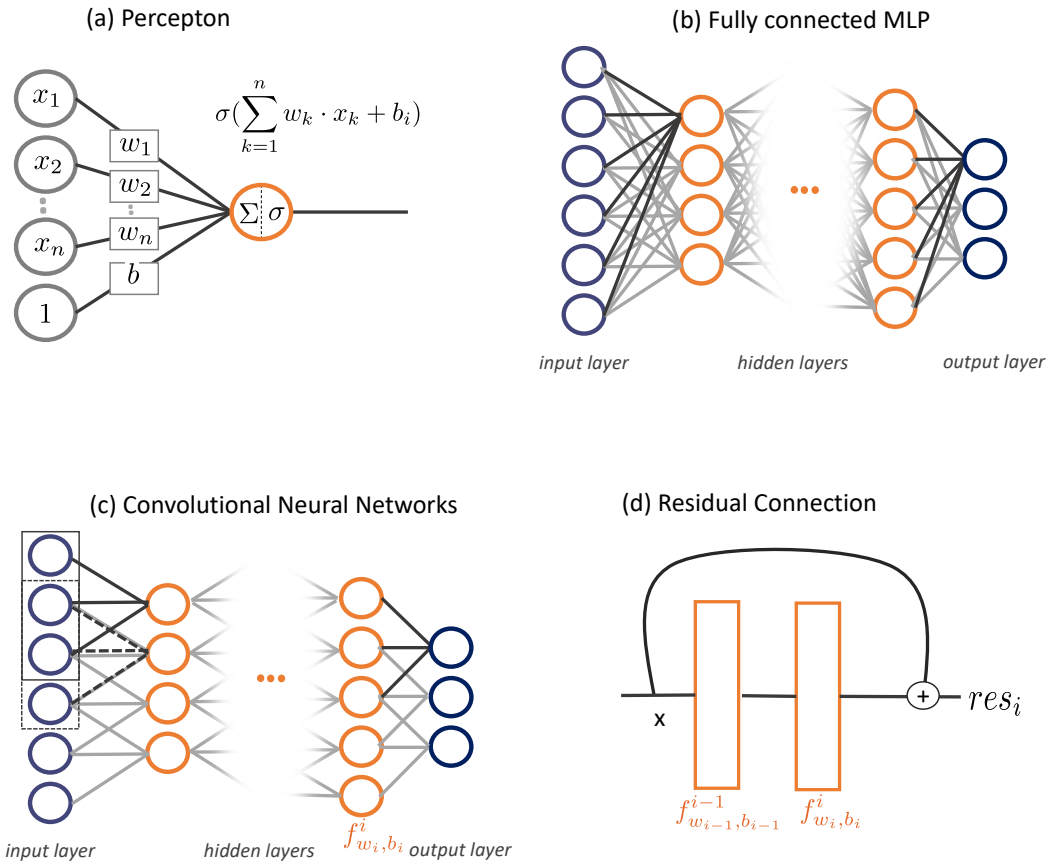


Fig. 3.4. **Neural networks.** Overview of different neural network components and feed-forward architectures.

results of the leaf nodes at different trees are aggregated in order to robustly obtain the final prediction.

The desired function f of equation 3.1 is therefore the final vote from an ensemble of binary decisions of the input space and the function parameters $\mathbf{w}$ are in this case defined by the splitting criterions.

Further information on random forest can be found in the original work of [81].

3.3.2 Deep Learning

Deep Learning does not make use of hand-crafted features, but discovers features that are relevant for tackling the problem at hand by approximating the mapping f with a composition of learned, nonlinear functions. Instead of revisiting the original input $\mathbf{X}$, the technique allows for more abstract connections and representations through its hierarchical, compositional architecture:

$$\hat{Y} = f(\mathbf{X}; \mathbf{w}) = f^l \circ f^{l-1} \circ \cdots \circ f^1(\mathbf{X}). \tag{3.4}$$

The function f^1 that receives the input values $\mathbf{X}$ is called the *first* or *input layer* of the network. The final layer f^l is the *output layer* and produces an approximation for $\mathbf{Y}$. Since we only specify the input and the desired output, i.e. do not have a direct influence on the mappings in-between, the intermediate layers $f^2 \cdots f^{l-1}$ are also called *hidden layers*.

The composition of these functions is called neural network, which were originally proposed with

$$f^i_{w_i,b_i}(x) = \sigma(\sum_{k=1}^{n} w_k \cdot x_k + b_i) = \sigma(w_i^T \cdot x + b_i), \tag{3.5}$$

where $\sigma(\cdot)$ is the activation function, w_i are the learned weights and b_i is the bias. It represents a fully-connected operation where all elements of x contribute to all elements of $f^i_{w_i,b_i}(x)$. Please note that x refers to the input of the function, e.g. $x = \mathbf{X}$ in case of f^1 and $x = f^1(\mathbf{X})$ in case of f^2. The nonlinearity is achieved by activation function $\sigma(\cdot)$ and is crucial, since the composition of the functions would otherwise result in a linear function that may not be suitable to approximate non-linear problems.

If the network consists of only one particular function f^1, it is called *Perceptron*. If the networks consists of several layers $f^1 \cdots f^l$ it is referred to as *multi-layer perceptrons (MLP)*. *Convolutional Neural Networks (CNNs)* usually also comprise several layers, but act more locally on the input signal by using of 2D convolution kernel instead of weighted sums:

$$f^i_{w_i,b_i}(x) = \sigma(w_i * x + b_i). \tag{3.6}$$

This offers several advantages [82]: the filter is translation invariant and the same weights w are repeatedly applied to cover the entire input signal (*parameter sharing*). Additionally, the dimensionality of w_i is lower than in in the fully connected case (equation 3.5) because of local-only connectivity. Consequently the number of free parameters and therefore the complexity of the network is reduced. In many CNN architectures, the convolutional layers are followed by weight-less pooling layers that change the dimensionality of the signal. If all learned layers are convolutional layers, the network is usually referred to as *Fully Convolutional Neural Network (FCN)* [83].

In the last years there has been a trend towards deeper networks to achieve better results: the AlexNet (2012) by Krizhevsky *et al.* consists of 8 layers, VGG (2014) by Simonyan *et al.* [84] of 19 layers and ResNet (2016) by He *et al.*[85] comprises already 50 layers and more. Unfortunately, creating deeper networks is not as straightforward as adding more layers, due to problems such as vanishing gradients and degradation [85]. Skip connections and residual blocks address this issue by allowing the information to flow through the networks via shortcuts. Signals from an early layer skip over one or several intermediate layers and are combined with one of the later layers, e.g. via summation [83] or concatenation [86]. A residual block [85] allows an identity mapping for every few stacked layers , e.g:

$$res_i = f^i_{w_i,b_i} \circ f^{i-1}_{w_{i-1},b_{i-1}}(\mathbf{x}) = \sigma(w_i * (\sigma(w_{i-1} * \mathbf{x} + b_{i-1})) + b_i + \mathbf{x}). \tag{3.7}$$

This enables deeper architectures by allowing the network an option to just skip subnetworks.

During *training*, the input $\mathbf{X}$ of a training sample $(\mathbf{X}, \mathbf{Y})$ is fed through the network with pre-initialized parameters and leads to a prediction $\hat{Y}$. The prediction error can be determined by an objective function as:

$$L(\mathbf{Y}, \hat{Y}) = \sum_{i=1}^{n} \delta(\mathbf{Y}_i, \hat{Y}_i) = \sum_{i=1}^{n} \delta(\mathbf{Y}_i, f(\mathbf{X}_i; \mathbf{w})), \tag{3.8}$$

where $\delta(\cdot)$ is a function that describes the distance of $\hat{Y}$ to $\mathbf{Y}$. A choice for δ is for example the L2 norm. According to the measure, the network weights are updated via backpropagation, so that the network output will be closer to the target output for the next sample.

During *testing*, a new unseen sample of $\mathbf{X}$ is fed forward through the network with learned weights and the final layer produces the prediction.

Further information on deep learning can be found in the book of Goodfellow *et al.* [87].

3.4 Evaluation Metrics

The performance of an instrument tracker can be evaluated depending on the object representation. In this section, a brief overview of the common evaluation metrics is presented for tracking an instrument via reference points (Section 3.4.1) or segmentation (Section 3.4.2). For machine learning-based approaches, the available dataset is usually split into independent, preferably sequence- or surgery-wise subsets to ensure that the model is does not simply memorize scenarios, but actually learns a generally applicable representation. During training, the parameters are learned on the training set and the model is selected using the validation set. The overall performance of the final model after the training process is evaluated on an independent testing set. In medical applications, the amount of data is rather limited and the validation set may be so small that it does not provide reliable results. Therefore it is very common to perform a cross-validation.

3.4.1 Reference Point

The following metrics can be used to evaluate accuracy for the reference points:

Keypoint Threshold (KT) [88] (or Threshold Score) addresses the quality of an estimated reference point location via a pixel distance measure. An estimated reference point location $j \in \mathbb{R}^2$ is evaluated as correct if the Euclidean distance to the ground truth annotation $\hat{j} \in \mathbb{R}^2$ is lower than a fixed pixel threshold $T \in \mathbb{R}$:

$$\|j - \hat{j}\| < T \, .$$

Notably, it yields a separate evaluation for every reference point $j \in J$.

Keypoint Threshold Bounding Box (KBB) [7] addresses the quality of an estimated reference point location via a pixel-distance measure that adapts to the pixel size of the instrument manipulator. The metric is based on the observation that setting one threshold for different image resolutions and instruments sizes makes the results of the KT metric unreliable when

averaged over several images. Instead, the threshold is scaled according to a tightly cropped, axis-aligned bounding box which contains all reference points of the instrument in the respective frame. A reference point j is located correctly if

$$\|j - \hat{j}\| < \alpha \cdot \max(h, w) \ ,$$

where $\hat{j}$ is the ground truth annotation of the to-be-evaluated reference point, w and h are the width and height of the bounding box around the instrument given by the ground truth, and $\alpha \in \mathbb{R}$. It should be noted that this metric is only computable if the ground truth of all reference points is given. However, the evaluation is also applicable if only one reference point is estimated.

Strict Percentage of Correct Pose (strict PCP) [89] addresses the quality of the prediction for a part of an articulated object. A prediction for a part connected by two reference points $j_1, j_2 \in \mathbb{R}^2$ is evaluated as correct only if both the Euclidean distances of the predicted reference points j_1, j_2 to its ground truths $\hat{j_1}, \hat{j_2}$ are lower than a threshold as a function of the ratio $\alpha \in \mathbb{R}$ times the ground truth length of the part, e.g. both of the following equations have to be fulfilled:

$$\begin{aligned} \|j_1 - \hat{j_1}\| < \quad & \alpha \cdot \|\hat{j_1} - \hat{j_2}\|, \\ \|j_2 - \hat{j_2}\| < \quad & \alpha \cdot \|\hat{j_1} - \hat{j_2}\| \ . \end{aligned}$$

3.4.2 Segmentation

If the instrument is tracked in terms of segmentation, we can consider a binary labelling of the image region Ω into two disjoint subsets $\Omega_{instrument}$ and $\Omega_{background}$. Let *TP* be the true positives, *FP* the false positives, *FN* be the false negatives and *FP* be the false positives, as exemplarily depicted in Figure 3.5 for the target class *instrument*. Then the the following metrics can be used for evaluating the performance of the tracker:

Precision is the proportion of correctly labeled target class pixels to all pixels assigned to the target class. It is defined as

$$\textit{Precision} = \frac{TP}{TP + FP}.$$

Recall is the proportion of correctly labeled target class pixels to all ground truth pixels of the target class. It is defined as

$$\textit{Recall} = \frac{TP}{TP + FN}.$$

Specificity is the proportion of correctly labeled not-target class pixels to all ground truth pixels of the not-target class. It is defined as

$$\textit{Specificity} = \frac{TN}{TN + FP}.$$

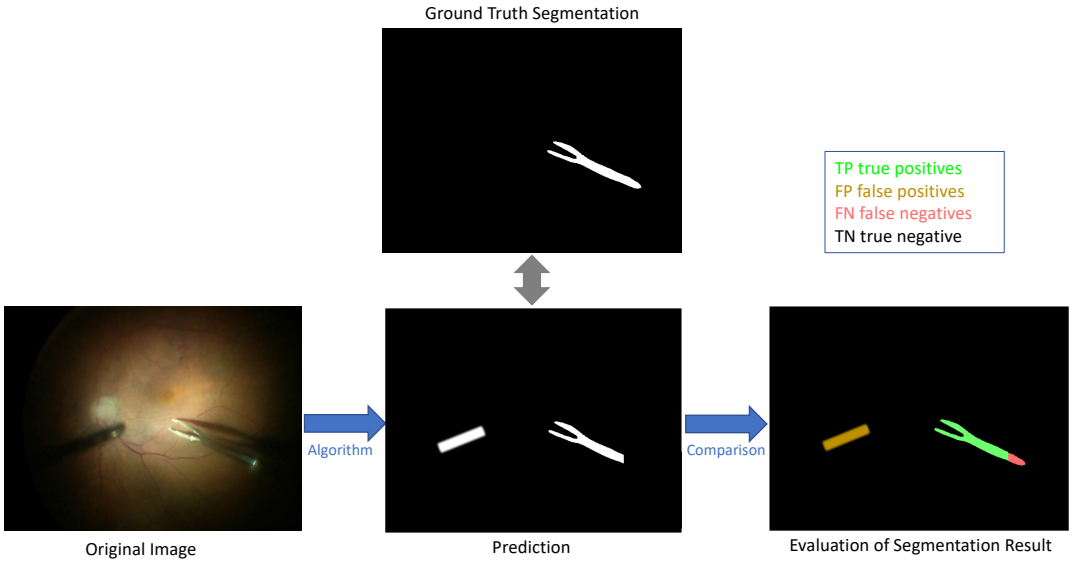


Fig. 3.5. **Segmentation Evaluation.** For the evaluation of a binary image segmentation, the segmentation result is compared to the ground truth segmentation. The resulting number of pixels for each category is used for computing segmentation metrics such as recall or specificity.

Balanced Accuracy (BACC) is the arithmetic mean of recall and specificity. It is defined as

$$BACC = \frac{1}{2} \cdot \left(\frac{TP}{TP + FN} + \frac{TN}{TN + FP} \right).$$

DICE coefficient (or F1 score) is the harmonic mean of recall and precision. It is defined as

$$DICE = \frac{2TN}{2TP + FP + FN}.$$

3.5 Requirements and Challenges

The tracking algorithms that are presented in this dissertation are intended to be employed in medical interventions. This means in particular that the methods must be adapted to the needs of the surgeons and the conditions of the surgical environment in order to provide reliable assistance. There are particular requirements that have to be considered, including:

Computational Complexity. As outlined in Section 2.2 and 2.3, instrument tracking can assist the surgeon pre-, intra- as well as post-operatively. During the intervention, it is essential that the algorithm runs in *real-time*. Already a delay of a second would make the tracker unsuitable for many applications, e.g. augmented reality or intra-operative guidance. Additionally, the *computational footprint* of the algorithm should be low as the computational resources in the operation theatre are limited. This is especially important if the instrument tracking is employed as an input for another algorithm. For the assistance before or after the surgery, these constraints are not as strict. Of course it is desirable for the physician or another algorithm to receive the tracking results as soon as possible. However, this can be in the range from seconds or even hours in case of reviewing for surgical training. Furthermore,

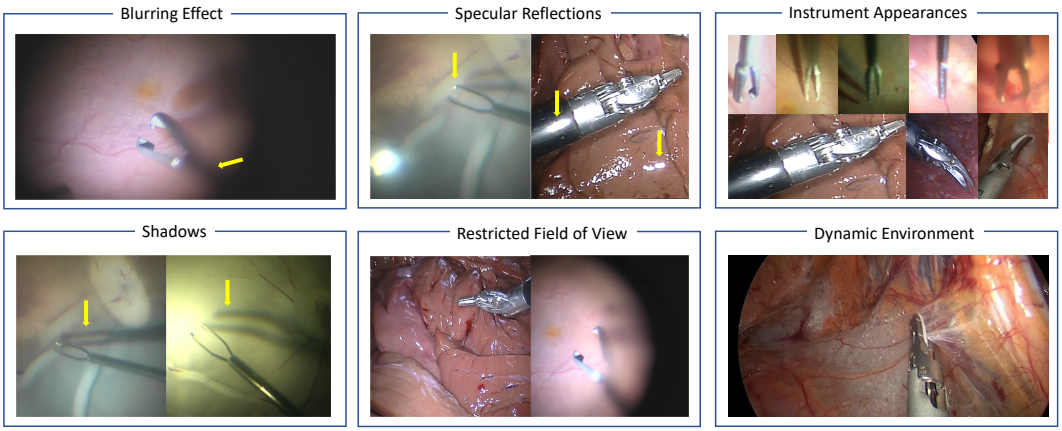


Fig. 3.6. **Examples of image peculiarities.** The surgical image data poses various challenges for image tracking. One of the main difficulties arises from the fact that the image data in such a setting captures only a very restricted field of view of the highly dynamic environment. Especially the non-static directional light source complicates the task by creating shadows, uneven illumination and specular reflections in the images. Endoscopic Surgery Images are from the Endovis Challenge 2015 [74].

the equipment is not constrained by the sterile conditions within the operating room and can be modified or extended.

Performance. The tracking algorithm has to meet the *precision and accuracy* requirements of the application. An error of several pixels for tracking a reference point would not impact the detection of a general instrument movement. But the same error would make it infeasible to reposition the iOCT to the instrument tips. Furthermore, the tracker has to be *robust* to the various peculiarities that can appear in the surgical images. The available labelled training data in the addressed medical fields is limited and usually does not cover all the variations that can occur during a surgery. Consequently, the tracker has to have a high generalization capability to unseen situations, so that the surgeon can rely on its performance.

Workflow compatibility. The aim is to assist the physician during the operation and smoothen the workflow. For this, the tracking algorithm has to be designed in a supportive and not a distractive way, e.g. a surgeon clicking on the screen for the initialization of the tracker is not acceptable. Consequently, the *user-interaction* has to be restricted to a minimum to modify the surgery workflow as little as possible. It would be desirable if the tracking algorithm is *autonomous*, including initialization at the beginning of the surgery and re-initialization in case of tracking failure.

Fulfilling the aforementioned requirements already excludes several of the traditional Computer Vision approaches [77]. The set of potential tracking solutions is further reduced by the challenges posed by the characteristics of the surgical image data. As explained in Section 2.1, the video only captures a restricted field of view of a highly dynamic environment with no reliable static components. In Endoscopic surgery, the camera is handheld by an assistant and does not retain a constant angle towards the surgical scene. The directional light source that is fixed on the endoscopic camera creates specular reflections on the wet soft tissue and the metallic surface of the instrument. The anatomical background is usually soft tissue and has therefore a highly deformable appearance that changes with the angle or the instrument manipulation. In Vitreoretinal Surgery, on the other hand, only small tissue deformations take

place and the camera angle is relatively fixed by the microscope and the pupillary opening. Although the movement of the patient's eye is inhibited, the background is not completely static. A complicating issue for instrument tracking in this surgery is that the illumination is not fixed to the camera. It is a handheld endoilluminator that the surgeon actively uses for estimating the depth of the observed scene. The instrument is between this light source and the retina and therefore casts shadows that change with the respective angle. Another major difficulty in this surgery is that the high magnification leads to an almost co-planar anatomy and usually only a small part of the observed scene is focused. Consequently, major parts of the image may show a severe blurring effect. For both types of surgeries, the instrument is metallic and therefore a texture-less object that is difficult to describe with feature representations Φ. The illumination variations, caused by either the moving camera or the light source, complicate the tracking task even further by creating uneven and specular reflections or violating the static light assumption of traditional Computer Vision approaches such as optical flow. Additionally, challenges that arise for all visual tracking approaches have to be taken into consideration, e.g. that the observed scene is only a 2D projection of the 3D world and therefore inherently contains a loss of information.

4 Related Work

Instrument tracking for medical interventions has been an active field of research in recent years and is still growing. This chapter provides an extract of published methods related to this dissertation. It should be noted that this includes publications that inspired our contributions as well as alternative approaches that were developed simultaneously to our work by other authors.

There are various ways to categorize methods of this field as summarized by Bouget *et al.* [90], e.g. feature representation, model learning strategy or use of prior knowledge. However, it is difficult to state that an approach is the overall best method since the requirements are highly application-dependent. Different object representations, for example, lead to a different complexity of the tracking problem. For some applications, a bounding box around the instrument tip may be sufficient, while for others a precise 2D pose estimation is necessary. Methods that target the same instrument representation and follow the same validation strategy on a dataset can be compared in terms of the metrics discussed in Section 3.4. However, better results according to the metric may result in higher computational cost, which would be a disadvantage if the visual tracking was intended as an input for a subsequent algorithm.

Following the arguments in the introductory chapters, the methods will be differentiated according to their instrument representation, i.e. their tracking objective. A chronological overview of the mentioned publications is depicted in table 4.1.

Localization (coarse):
A coarse localization of the instrument such as a bounding box or a single reference point can convey the general movement of the tool. In an early work, Reiter *et al.* (2010) [91] present a temporal tracker that provides a box around a user-defined interest point based on online learning. The tracker propagates information of previous frames to initiate a coarse segmentation in the current frame using an online generated database of tracked gradient-based features. This estimation is refined by current features that are integrated via a likelihood map to address the issue of dynamically-changing environment. A year later, Richa *et al.* (2011) [92] presented an alternative approach for regressing a bounding box that is based on weighted mutual information of stereo images. In the first step, the instrument tip position is approximated by a brute force search using the mutual color information of the stereo image. In the subsequent step, a gradient-based tracking iteratively optimises this estimation. One of the first data-driven approaches for coarse localization in terms of one reference point was presented by Sznitman *et al.* (2012) [88]. They suggested to combine a simple gradient-based tracking with a deformable detector. In the first step, the gradient-based tracker provides a rough location of the instrument, so that the detector only has to consider a reduced region for predicting the presence of an instrument. The detector itself is based on an AdaBoost learning that uses deformable features and was extended with a Gaussian Pyramid to address the multi-scale problem. The final instrument position is estimated by a spatial and

score weighting of the detector responses. In a work by Li *et al.* (2014) [93] an online learning approach was presented that combines temporal tracking and frame-to-frame detection of a user-defined reference point. A median flow tracker provides a bounding box around the tracked point to a three-stage detector. The detector is learned online and determines the precise reference point location via a combination of variance filter, Random Forest for pixel wise classification and 1-Nearest-Neighbour classifier. One of the latest methods regarding coarse localization in terms of bounding boxes was published for detecting the use of specific tools. Sarikaya *et al.* (2017) [94] proposed a Region Proposal Network (RPN) using the RGB image and the optical flow as input. The estimated bounding boxes are fed in a subsequent network for instrument classification.

Contour or Segmentation:
The extent of an instrument in the image can also be expressed by its contour or segmentation. One of the most common approaches consists of employing this representation as an intermediate step for estimating its 3D world coordinates [80, 95, 96, 97, 98]. Pezzementi *et al.* (2009) [95] estimated a binary segmentation for evaluating its consistency with a rendered 3D model of the instrument using color and texture features. A couple of years later, Baek *et al.* (2012) [96] employed the likelihood of an instrument contour measured by edge distance transform to evaluate its similarity of a projected 3D CAD model. The contour detection was realized by a combined edge distance generation using a canny detection algorithm on hue-saturation (H-S) color space and the intensity image. The particle filtering in the second step allowed to estimate the full state of the forceps. Shortly after, Reiter *et al.* (2013) [97] suggested a kinematic template matching method using gradient information of the instrument. Their idea was to learn from virtual renderings to match the kinematically-generated templates to the current image information. The template was hereby constructed by gradient information of a LINE detector. Instead of using sparse contour information, Allan *et al.* (2013) [98] suggested to employ Random Forest for supervised classification into binary instrument/background classes for every pixel. As a feature representation they chose several different colour spaces and structural descriptors. In a subsequent step they used this information to initialize a minimisation algorithm for 5DOF pose estimation of a level set framework. Reducing the parameter space, Zhou *et al.* (2014) [99] presented a temporal method based on standard Kalman filter and extended Kalman filter that estimates the contour and center line of the instrument. For every new frame, a Canny Edge filter provides candidate lines that are filtered by the previous Kalman state of the center line. In an alternative approach, Bouget *et al.* (2015) [100] detected the instrument contour via its local appearance and global shape. In the first step, the method assigns each pixel either to instrument or background based on the local appearance via a boosted decision forest. Subsequently, the global shape of the instrument is enforced via a tool-specific shape template. In the same year, Allan *et al.* (2015) [80] proposed to combine detection with temporal regularization. In their work, they segmented the image into multiple classes with random forest using using color based feature set of Hue, Saturation, Opponent 1 and Opponent 2. This semantic segmentation can then be used for statistical region based 3D pose estimation and is refined based on frame-to-frame tracking using the Lucas-Kanade method. However, also the improvement of segmentation accuracy itself has been of interest in recent years, especially in combination with deep learning. Garcia Peraza Herrera *et al.* (2016) [101] presented a FCN-based approach for binary segmentation of surgical instruments in minimally invasive surgery. To achieve the real-time requirement, they proposed to use the deep learning approach only for every couple

of frames. A continuous tracking is realised by propagating the segmentation with optical flow. In another work, Garcia Peraza Herrera *et al.* (2017) [102] reduced the computational complexity by including a multi-scale constraint inside the network architecture, either by cascaded aggregation of predictions or holistically nested architecture. Thereby, the number of parameters is reduced and real-time performance is enabled without optical flow. An alternative approach was presented by Pakhomov *et al.* (2017) [103] for multi-class instrument segmentation using deep residual learning with dilated convolutions.

Articulated object:
Another approach is to interpret the instrument as an articulated object with several reference points or parts. In case of a cylindric needle, the instrument can be parameterized as a line. Sznitman *et al.* (2011) [104] suggested to describe the tool in this case via the entry point of the instrument in the image, its angle to the image boundary and its length. The detection and tracking of the instrument can then be modelled as a single sequential entropy minimization problem, which they solve via Active Testing (AT). Reiter *et al.* (2012) [105] suggested to detect several reference points on an instrument with a manipulator by learning the appearances of natural landmarks with a multi-class classifier. The region of interest is first reduced to metallic parts of the image via a Gaussian Mixture Model. In the second step, the remaining pixels are classified with a Randomized Tree using the Region Covariance Descriptor. The final position of the reference points is determined for every frame via stereo matching of the feature tracks using normalized cross-correlation along the epipolar line and triangulation. However, this approach was not realizable in real-time at this point. To reduce the computational cost, Sznitman *et al.* (2014) [106] suggested to detect the instrument parts with an early stopping scheme. The reference regions are identified by a multiclass ensemble of gradient boosted regression trees. For every class, the early-stopping algorithm determines whether further computation is needed based on a probabilistic model. The relationship between the reference points on the instrument can also be represented by a Conditional Random Field, as presented by Alsheakhali *et al.* (2016) [107]. More recently, it was suggested to leverage the power of deep learning for the 2D pose estimation of the instrument. Kurmann [108] suggested to modify an established CNN architecture (U-Net) to simultaneously recognize the present instruments and regress their reference point positions. As in the original architecture, the feature layers are arranged in a U-shape by down- and upsampling operations and linked via skip connections. The instrument recognition classification-network separates from the lowest layer of the network. For the regression of the 2D locations, the features are upsampled to produce one 2D probability map per keypoint. In a single feed-forward pass both objectives are learned simultaneously by combining the respective losses. An alternative deep learning-based approach was presented by Du *et al.* (2018) [109]. They propose to combine a two-staged convolutional network with a graph-based parsing algorithm. First, a detection network segments the rough keypoint locations together with their associated connections. In the second step, this information serves as a spatial regularization in the subsequent regression network. The final prediction is obtained by eliminating outliers and bipartite matching.

Tab. 4.1. **Overview of related work.** Relevant recent research is listed chronologically and categorized according to their visual tracking algorithm, e.g. the object representation refers to the tracked object description in the image, which may not correspond to the overall target (3D coordinates). The horizontal, dashed line separates previous related work and publications that have been published simultaneously to the work presented in this dissertation. It should be noted that the real-time requirement is application and hardware dependent. Consequently, it should be interpreted as soft categorization (here: at least 10 fps). If a machine-learning algorithm was presented for the visual tracking, it can be distinguished whether it has been tested in a cross-validation setting. Abbreviations: *RM* = Vitreoretinal Microsurgery (Section 2.1.1), *Endo* = Endoscopic surgery (Section 2.1.2), *n.s* = not stated, *n.a* = not applicable, *RF* = Random Forest, *CNN* = Convolutional Neural Network, *FCN* = Fully Convolutional Network, *FCRN* = Fully Convolutional Residual Network.

Publication	Surgery	Object Representation	Tracking	In-Vivo Data	Real-Time	ML-Approach	Cross-Val.
Pezzementi *et al.* (2009) [95]	RM & Endo	segmentation (binary)	Detection	×	×	histogram matching or Gaussian mixture models	×
Reiter *et al.* (2010) [91]	Endo	segmentation (binary)	Temporal	✓	✓	no	n.a.
Sznitman *et al.* (2011) [104]	RM	line parametrization	Temporal & Detection	×	✓	Active Testing	×
Richa *et al.* (2011) [92]	RM	bounding box	Temporal	✓	n.s.	no	n.a.
Reiter *et al.* (2012) [105]	Endo	reference points (multiple)	Detection	×	×	Randomized Trees	×
Baek *et al.* (2012) [96]	RM & Endo	contour	Detection	×	✓	no	n.a.
Sznitman *et al.* (2012) [88]	RM & Endo	reference point (single)	Temporal & Detection	✓	✓	AdaBoost	✓
Reiter *et al.* (2013) [97]	Endo	gradient template	Detection	✓	✓	template matching	×
Allan *et al.* (2013) [98]	Endo	segmentation (binary)	Detection	✓	×	RF	×
Li *et al.* (2014) [93]	RM & Endo	reference point (single)	Temporal & Detection	✓	✓	online learning	n.a.
Sznitman *et al.* (2014) [106]	RM & Endo	reference point (multiple)	Detection	✓	✓	RF	×
Zhou *et al.* (2014) [99]	Endo	line parametrization	Temporal	✓	n.s.	no	n.a.
Bouget *et al.* (2015) [100]	Endo	contour	Detection	✓	✓	DF & SVM	×
Allan *et al.* (2015) [80]	Endo	segmentation (multi)	Detection & Temporal	✓	×	RF	×
Garcia Peraza Herrera *et al.* (2016) [101]	Endo	segmentation (binary)	Temporal & Detection	✓	✓	FCN	×
Alsheakhali *et al.* (2016) [107]	RM & Endo	reference points (multiple)	Detection	✓	✓	Conditional Random Field	×
Garcia Peraza Herrera *et al.* (2017) [102]	Endo	segmentation (binary)	Temporal & Detection	✓	✓	FCN	×
Sarikaya *et al.* (2017) [94]	Endo	bounding-box	Temporal & Detection	×	✓	CNN	✓
Pakhomov *et al.* (2017) [103]	Endo	segmentation (multi-class)	Detection	×	n.s.	FCRN	n.s.
Kurmann *et al.* (2017) [108]	RM & Endo	reference point (multiple)	Detection	✓	×	CNN	✓
Du *et al.* (2018) [109]	RM & Endo	reference point (multiple)	Detection	✓	×	FCN	n.s.

5 Summary of Contributions

Despite recent advances, the vision-based tracking of surgical tools in in-vivo scenarios remains challenging. The surgery-specific peculiarities such the high level of noise and the limited field of view create an unusual Computer Vision problem, as discussed in Section 3.5. The task is particularly difficult due to the hand-held directional light and the resulting difficulties, such as reflections on the metal instrument and strong shadows. Prior to the contributions discussed in this dissertation, most approaches relied on explicit modelling of the tracking problem (see Chapter 4). To approximate the function

$$Y_t = f(\Phi\left(\{I_s\}_{s=t_1}^{t_2}\right)) = f(X_t),$$

as defined in Chapter 3.1, we will build on machine learning approaches that have shown to successfully address related problems in general Computer Vision. Instead of projecting a 3D model or manually describing the instrument appearance, we will let the method learn from available data and statistically approximate the function. In all our contributions, we model the instrument as an articulated object with 2D reference points. For Vitreoretinal Surgery, the instrument representation is defined as $\mathbf{Y} = (C, L, R)^\top \in \mathbb{R}^{3\times 2}$, whereby C is the central natural landmark of the instrument and L or R are left or right tip of the tool, respectively. For endoscopic surgery, the tracked reference points are defined by the provided ground truth. All presented methods achieve real-time performance and are evaluated in a cross-validation setup.

In the following, a summary of the proposed Feed-Forward Pipeline (Section 5.1), Robust Pipeline (Section 5.2) and End-to-End Pipeline (Section 5.3) will be presented. For more details and results, please have a look at the original publications that are included in the Appendix A.

Tab. 5.1. **Overview of major contributions.** Abbreviations: *RM* = Retinal Microsurgery (Section 2.1.1), *Endo* = Endoscopic surgery (Section 2.1.2)

Publication	Surgery	Object Representation	Tracking	In-Vivo Data	Real-Time	ML-Approach	Cross-Val.
Rieke *et al.* (2015) [10]	RM	reference points (multi)	Temporal & Detection	✓	✓	RF (offline)	✓
Rieke *et al.* (2016) [7]	RM & Endo	reference points (multi)	Temporal & Detection	✓	✓	RF (offline)	✓
Rieke *et al.* (2016) [6]	RM	reference points (multi)	Temporal & Detection	✓	✓	RF (offline and online)	✓
Rieke and Laina *et al.* (2017) [3]	RM & Endo	reference points (multi) & segmentation	Detection	✓	✓	FCRN	✓

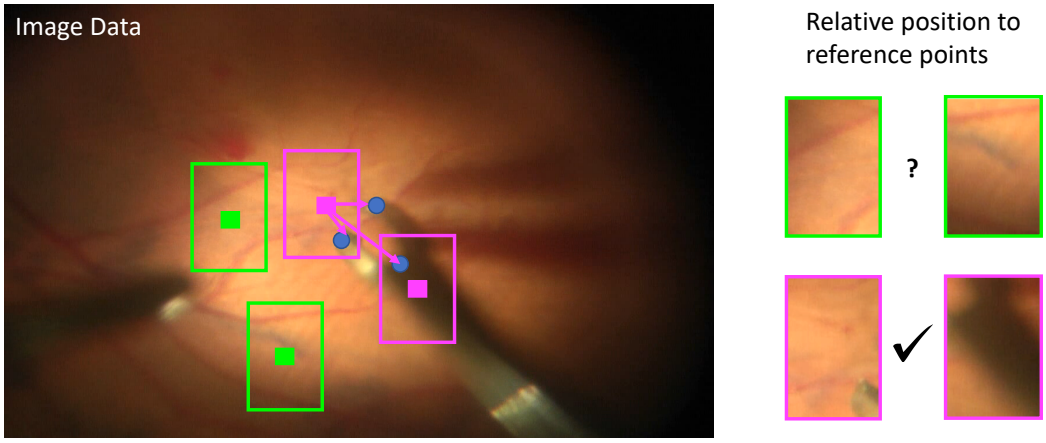


Fig. 5.1. **Need for template tracking.** Mainly the region around the tracked instrument provides reliable clues about the relative position to the 2D reference points.

5.1 Feed-Forward Pipeline

Given an image, the aim is to provide the 2D locations of the reference points that describe the instrument movement. The gradient of an image has proven to be a key information [88, 96, 97, 99, 106] for this task. However, computing advanced, gradient-based feature representations Φ on the entire image space is computationally expensive and may be misleading. As depicted in Figure 5.1, mainly the region around the tracked instrument provides reliable clues about the precise position of the 2D reference points. Therefore, we suggest a Feed-Forward Pipeline (Figure 5.2), in which we first reduce the search space by a template tracker before we estimate the instruments' 2D pose. For both steps, we build on the machine learning technique of Random Forests (RF) (Chapter 3.3.1), which have shown to be reliable for noisy data.

5.1.1 Surgical Tool Tracking and Pose Estimation in Retinal Microsurgery (MICCAI 2015)

In this work [10], we propose a method that breaks down the difficulty into two tasks: template tracking and 2D pose estimation. These two steps are combined into a Feed-Forward pipeline in order to achieve the overall goal of localizing the instrument reference points for every frame in real-time.

Temporal Tracking: The objective of the tracker is to determine the 2D translation vector $\delta\boldsymbol{\mu}$ that updates the location of the bounding box I_B around the instrument tip for the current frame I_t, given an image sequence $\{I_i\}_{i=0}^{t}$. In order to keep computational cost low for this step, the tracker relies on the readily available image intensity information as feature representation Φ. For this, n_s sample points $\{\mathbf{x}_t^s\}_{s=1}^{n_s}$ are randomly selected within the template, such that it can be described using the intensity vector $\mathbf{i}_t = [I_t(\mathbf{x}_t^s)]_{s=1}^{n_s}$. For determining the 2D translation vector $\delta\boldsymbol{\mu}$ we employ intensities in the current frame $\mathbf{i}_t = [I_t(\mathbf{x}_{t-1}^s)]_{s=1}^{n_s}$ using the location of sample points from the previous frame $\{\mathbf{x}_{t-1}^s\}_{s=1}^{n_s}$. Differently from [110], we propose to correlate multiple templates in order to compensate for the motion of the

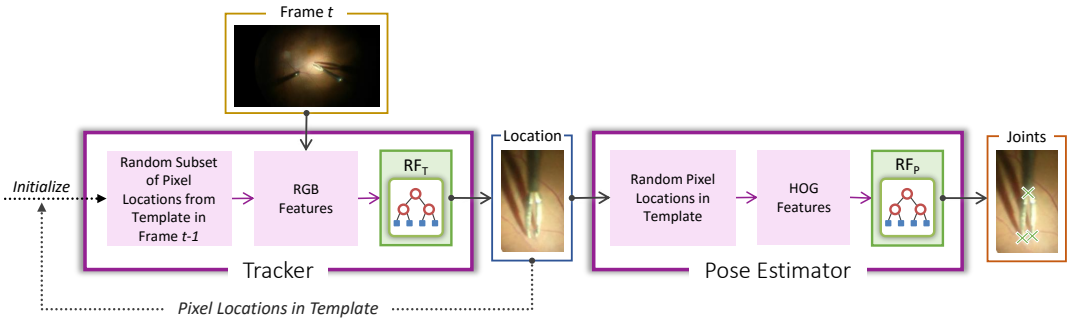


Fig. 5.2. **Feed-Forward Pipeline.** In this approach, we introduce a instrument tracking method based a dual Random Forest with a feed-forward connection. A multi-template tracker determines the region of interest around the instrument tip by relating the movement of the instrument to the induced changes on the image intensities. Within this bounding box, a gradient-based pose estimation infers the instrument reference points.

instrument tip as it opens and closes, as well as for the strong illumination changes and photometric distortions that are typically present during the surgery. Consequently, we set the input $\mathbf{X} = \delta\mathbf{i}$, the parameters $\mathbf{Y} = \delta\boldsymbol{\mu}$ and the function $H(\cdot)$ as the standard deviation for the Random Forest. The leaves store the mean and standard deviation of the parameters that are directed to that node.

2D Pose Estimation: Given a bounding box $I_B \subset I_t$ around the tool tip fed forward from the tracking step, the objective of the pose estimation algorithm is to determine the 2D positions of the reference points. As a feature representation Φ, and therefore as input space $\mathbf{X}$, we use the Histogram of Oriented Gradients (HOG) features of randomly selected image patches within the template. The output space $\mathbf{Y}$ are the patch-associated offsets in the template coordinate system. The binary split function θ divides the samples based on a threshold in one dimension of $\mathbf{X}$. The function $H(\cdot)$ is chosen to be the sum-of-squared-differences. During the prediction, every tree provides a vote for the offset from the patch location to all 2D reference point locations. In order to combine the votes and to find the most probable location of the joint, a greedy dense-window algorithm is applied, as in [111], and back projected to the image coordinate system. The final output of the pose estimation step are the 2D coordinates of each reference point in $\mathbf{Y}$.

At the time of publication, the proposed method outperformed state-of-the-art instrument tracking approaches in the field of Vitreoretinal Surgery and was honoured with the Young Scientist Award at the International Conference on Medical Image Computing and Computer Assisted Intervention in 2015.

5.1.2 Real-Time Localization of Articulated Surgical Instruments in Retinal Microsurgery (Med. Image Anal. 2016)

In this work [7], we extended and extensively evaluated the method [10] presented in Section 5.1.1. Instead of using only the grayscale information, as in [10], the tracker employs the RGB space, where the difference between the metallic instrument and the background is more visible. The dataset of *in-vivo* Vitreoretinal Surgery sequences was extended to 18 videos, which allowed to evaluate the performance of the algorithm regarding generalization

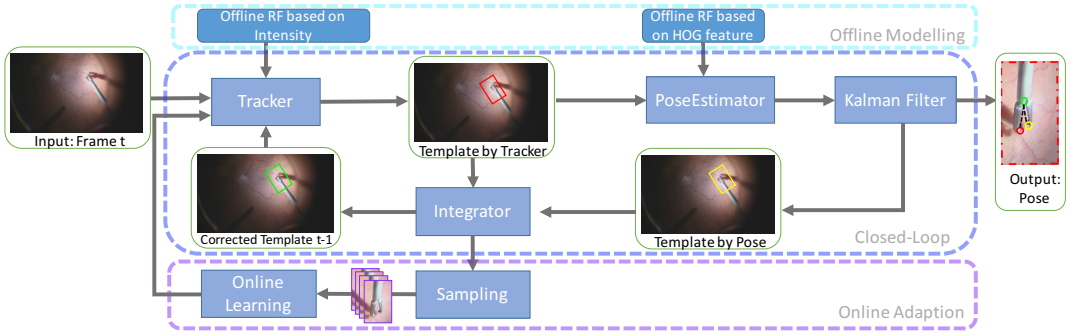


Fig. 5.3. **Robust Pipeline.** Building on the offline-learned dual RF of the feed-forward pipeline, we can develop a robust pipeline by adapting the offline model to online information while tracking and by "closing the loop" between the tracking and 2D pose estimation.

to different tool types and different background conditions. The results of the proposed method on this novel dataset are compared to the one of an online tracker TLD [112] and the offline learned FPBC [106]. Another main contribution is the introduction of a new metric (KBB, Section 3.4) that accounts for the variations in instrument size and image resolutions. Furthermore, we show that the proposed method is applicable to laparoscopic instrument sequences.

5.2 Robust Pipeline

The Feed-Forward Pipeline (Section 5.1) allows to track the instrument reference points with high accuracy in real-time on a CPU. However, both Random Forests are learned offline and therefore rely on the information provided in the training dataset. Another disadvantage is that the data is fed forward and therefore the pose estimation depends on the output of the tracker. The main idea of the robust pipeline is twofold: first, the two RF can benefit from each other's strength during testing and thereby increase the accuracy and robustness of the pipeline. By defining the template based on the reference points, we can fuse the complementary color-based and gradient-based predictions in a synergical way. Secondly, we suggest to adapt the RF online to incorporate the appearance changes learned by the trees with real photometric distortions witnessed at test time.

5.2.1 Real-Time Online Adaption for Robust Instrument Tracking and Pose Estimation (MICCAI 2016)

This work [6] advances the dual-Random Forest method presented in [10] to a robust pipeline by adapting the offline model to online information while tracking and by "closing the loop" between the tracking and 2D pose estimation. For the template tracking and 2D pose estimation components, we use the RF-based approach as described in Section 5.1. The *template tracking* employs the intensity information as feature representation. In addition, we assume a piecewise constant velocity from consecutive frames, so that the input to the forest is a feature vector concatenating the intensity values on the current location of the template $\mathbf{I}_t(\mathbf{x}_p)$ with a velocity vector $\mathbf{v}_{t-1}$. The *2D pose estimation* as described in Section 5.1

employs gradient information and considers every frame as a still image, although the surgical movement is usually continuous. Therefore differently from [10], we enforce a temporal-spatial relationship for the predicted reference point locations via a Kalman filter [113] using the 2D location in the frame coordinate system and their frame-to-frame velocity. Both feature representations are valuable in the context of Vitreoretinal Surgery: the utilized tool is metallic, and therefore many pixels on shaft and manipulator of the instrument share the same color. Consequently, the gradient information tends to be a more unique representation if we want to regress the location of a specific point on the instrument. However, if blur or motion artefacts are present in the image this gradient information vanishes and it becomes difficult to distinguish between the instrument and the background. In this case, the color information can still provide valuable information.

Closed Loop via Integrator: By defining the tracked template as a similarity transform on the reference points, the prediction of the gradient-based pose RF can directly be connected to the prediction of the intensity-based template tracker RF. Depending on the certainty of the separate Random Forests, we define the scale s_F and the translation t_F of the joint similarity transform as the weighted average

$$s_F = \frac{s_T \cdot \sigma_P + s_P \cdot \sigma_T}{\sigma_T + \sigma_P} \quad \text{and} \quad t_F = \frac{t_T \cdot \sigma_P + t_P \cdot \sigma_T}{\sigma_T + \sigma_P},$$

where σ_T and σ_P are the average standard deviation of the tracking prediction and pose prediction, respectively, and the t_F is set to be greater than or equal to the initial translation. If σ_T is higher than a threshold τ_σ, the tracker transmits the previous location of the template, which is subsequently corrected by the similarity transform of the predicted pose.

Online Adaptation: To address the issue of generalized modelling, we propose to perform an online learning strategy in addition to the offline learning of the tracker. The main aim is to stabilize the tracker by adapting its forest to the specific conditions at hand. Depending on the confidence evaluated by the aforementioned Integrator, the current template sample is forwarded to a separate thread. By imposing random synthetic transformations on the bounding boxes that enclose the templates, we can build an online learning dataset with pairs of feature and translation vectors. The resulting trees are incrementally added to the existing forest, so that the prediction for the succeeding frames include both the generalized and the environment-specific trees. The offline learning - online adaption leads to a substantial improvement regarding the generalization to scenarios that are not captured in the training dataset.

The performance of the proposed method is evaluated on two different *in-vivo* RM datasets and demonstrates remarkable advantages with respect to the state of the art in terms of robustness and generalization.

5.3 End-to-End Pipeline

For both the Feed-Forward (Section 5.1) and Robust Pipeline (Section 5.2) the feature representation Φ was explicitly modelled via either color information or HOG features. In the End-to-End Pipeline (Figure 5.4) we go one step further and allow the algorithm to learn a

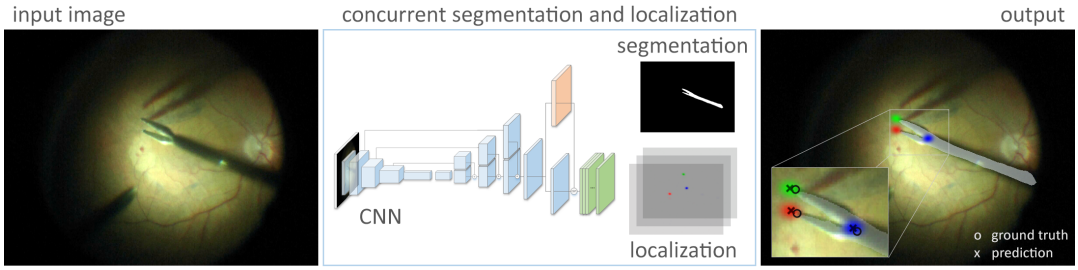


Fig. 5.4. **End-to-End Pipeline.** Instead of using explicit feature representations, we leverage deep learning techniques to simultaneously regress segmentation and 2D pose of the instrument. The network architecture is a fully convolutional neural network with skip and residual connections.

suitable feature representation by leveraging deep learning techniques. In other words, we learn the direct mapping $Y = f(I_t)$ instead of $Y = f(\Phi(I_t))$.

The targeted instrument representation $\mathbf{Y}$ is based on the following observation: As outlined in Section 4, two-step methods for reference point tracking can employ instrument segmentation either as pre- or as post-processing step. This suggests that tracking of an instrument landmark and its segmentation are not only dependent, but indeed inter-dependent. On the one hand, the instrument reference point can only be within the instrument segmentation. On the other hand, the reference points indicate the moving parts of the instrument and therefore the pixels that are most difficult to classify. So instead of carrying out the objectives as two subsequent pipeline stages, we propose to perform instrument segmentation and 2D pose estimation simultaneously, in a unified deep learning approach.

5.3.1 Concurrent Segmentation and Localization for Tracking of Surgical Instruments (MICCAI 2017)

Regressing the locations of the reference points corresponds to estimating a set of n 2D coordinates ($Y_{ref} \in \mathbb{R}^{n\times 2}$). A segmentation of the image with width w and height h into c classes, however, is represented by $Y_S \in \mathbb{R}^{w\times h\times c}$, and therefore lies in a solution space with completely different dimensionality. In this work [3], we show that reformulating the pose estimation task as a heatmap regression allows for representing semantic segmentation and localization with equal dimensionality. As a consequence the objectives can leverage their spatial dependency and facilitate simultaneous learning. Similar to established fully convolutional (FCN) architectures, it consists of a compressing or encoding path and an expanding or decoding path.

Network architecture:
Encoder: For the compressing part of the three proposed models, we employ ResNet-50 [85], a state-of-the-art architecture that achieves top performance in several Computer Vision tasks, such as classification and object detection. It is composed of successive residual blocks, as described in Section 3.3.2, and is pre-trained on ImageNet. Although deeper versions of ResNet exist, we use the 50-layer variant to allow real-time performance during testing.
Decoder: Predicting an absolute target location is arbitrary and ignores image context. Instead, we regress a heatmap for each tracked landmark in the proposed model. It is created by apply-

ing a Gaussian kernel to the reference point ground truth position and therefore represents a confidence of being close to the actual location of the tracked point. By having the same size as the segmentation, they can explicitly share weights over the entire network. The decoder part of the network consists of stacked up-sampling layers with residual connections [114].
Connections: We further enhance the encoder-decoder architecture with long-range skip connections that sum lower-level feature maps from the encoding into the decoding stage. Finally, we enforce a strong dependency of the two tasks by only separating them at the very end. The predicted segmentation scores are concatenated before the softmax operation to the last set of feature maps as an auxiliary means for guiding the location heatmaps.

Training: Given an image $X \in \mathbb{R}^{w \times h \times 3}$, we denote a training sample as (X, Y_S, Y_{ref}), where $Y_{ref} \in \mathbb{R}^{n \times 2}$ refers to the 2D coordinates of n tracked landmarks and $Y_S \in \mathbb{R}^{\frac{w}{2} \times \frac{h}{2} \times c}$ represents the semantic segmentation for c labels. The overall loss that combines both objectives can then be defined by:

$$l_{CSL} = \quad l_S(\tilde{Y_S}, Y_S) + l_{ref}(\tilde{Y}_{ref}, Y_{ref}) \tag{5.1}$$

with

$$l_S(\tilde{Y_S}, Y_S) = \quad -\frac{1}{wh} \sum_{x=1}^{w} \sum_{y=1}^{h} \sum_{j=1}^{c} Y_S(x,y,j) \log \left(\frac{e^{\tilde{Y}_S(x,y,j)}}{\sum_{k=1}^{c} e^{\tilde{Y}_S(x,y,k)}} \right) \text{ and} \tag{5.2}$$

$$l_{ref}(\tilde{Y}_{ref}, Y_{ref}) = \quad \frac{\lambda_H}{n} \sum_{i=1}^{n} \sum_{x=1}^{w} \sum_{y=1}^{h} || \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{||y_i - (x,y)^T||_2^2}{2\sigma^2}} - \tilde{y}^*_{x,y,i} ||_2^2 \tag{5.3}$$

Testing: The point of maximum confidence in each predicted heatmap $\tilde{y}_i^* \in \mathbb{R}^{\frac{w}{2} \times \frac{h}{2} \times n}$ is used as the location of the respective instrument reference point. Notably, a high variance in the predicted map can indicate a missing or misdetected reference point.

The resulting model is trained jointly as well as end-to-end for both tasks, relying only on contextual information. It is important to notice that it is capable of reaching both objectives efficiently without requiring any post-processing technique. At the time of publication, this was the first approach employing deep learning for surgical instrument tracking by predicting segmentation and localization simultaneously, and it is successful despite limited data. The method was evaluated for both Vitreoretinal and Endoscopic surgery and outperformed application-specific algorithms as well as other popular deep learning architectures.

Part III

Conclusions and Outlook

6 Summary and Findings

This dissertation explored techniques for visual tracking of surgical instruments and introduced novel approaches based on machine learning methods. In this chapter, I will summarize our work and point out some of the most important findings.

The first part of this dissertation provides an introduction to the scientific problem and motivates the need for instrument tracking. To that end, a detailed overview of the addressed surgeries including the surgical setup, the characteristics and the major challenges was provided. This allowed for an understanding for the respective difficulties and therefore the need of assistance for the surgeon. The fact that the interventions are digitally recorded in current clinical practice makes Computer Vision methods a natural direction for assisting the surgeon during these complicate procedures. Finally, I outlined how Computer Vision can support the surgeon in general and depicted in particular the potential impact of instrument tracking in this context.

In the second part, I moved forward to visual tracking itself. The first chapter of this part includes the essential techniques and considerations on which our contributions build upon. I explained the concept of visual tracking and how the instrument can be represented for this task. Furthermore, the fundamental principles for learning from data was provided, with focus on Random Forest (RF) and Fully Convolutional Networks (FCN), as well as evaluation methods for assessing the tracking performance regarding different object representations. Finally, the requirements and challenges were outlined that arise from the application of the developed method during the surgery as well as the distinctive image peculiarities which are present in the respective surgical image sequences. The second chapter of this part provided an overview of related work for visual instrument tracking, categorised according to the instrument representation of choice. At the time of the first work presented in this dissertation, various methods addressed the tracking problem either by explicit mathematical or rather low-level statistical modelling. Only few approaches suggested a data-driven solution [88, 98, 106]. Building on the idea of statistically approximating the desired tracking function by learning from data, we developed several machine-learning based approaches. In the first approach, we suggested a *Feed-Forward pipeline* based on a dual RF. A temporal tracking algorithm employs intensity values at random locations as features for the RF and regresses a bounding box around the instrument manipulator. The subsequent 2D pose estimation detects the reference points of the instrument by HOG features of random patches within this bounding box. The method was extensively tested on both Vitreoretinal and Endoscopic surgery sequences and outperformed state of the art in terms of reference point accuracy. The performance on additional, not annotated *in-vivo* sequences was also qualitatively acceptable. We pushed the method forward from the offline testing of surgical sequences on the computer to real integration in the microscope of our industrial partner (Zeiss Meditec, Munich, Germany). As an application we investigated the automatic positioning of the iOCT [8]. However, when surgeons tested our method in an experimental setup with porcine eyes, the algorithm was

not as robust as the evaluation on the annotated human eye dataset suggested. One of the reasons is that the Random Forests of the method are trained offline and therefore completely rely on the information available in the training dataset.
To address this issue, we suggested an online adaptation in the *robust pipeline*. In this approach we still make use of the available offline information to build the aforementioned dual RF, but enhance the tracking forest online by incrementally adding new trees that are trained on the current conditions. Another main contribution of this work was the combination of two different RF outputs in a synergic way via a closed loop. By defining the tracked template based on the reference points, we can feed back the 2D pose information and correct the temporal tracking based on the confidence of the RF. This cooperative prediction in combination with online adaptation lead to significant improvement in tracking accuracy when evaluated on two different *in-vivo* Vitreoretinal datasets. As both feed-forward and robust pipeline require an initialization of the temporal tracker, we further improved the methods by developing a fast bounding-box localization and failure detection algorithm [4] that transforms the pipelines into a fully automatic framework. It should be noted that to this end only an approximate template detection is needed. For the feed-forward pipeline, the requirement is that the reference points are included. For the robust pipeline, the template will be corrected after the first frame by the RF of the pose estimation. The resulting framework was integrated in the Zeiss microscope and tested on porcine eyes. Compared to the feed-forward pipeline, the tracking was more reliable and was also able to withstand unseen illumination scenarios. An inherent limitation of these pipelines lies in the statistical abstraction capacity of the Random Forest and the need of explicit feature representation.
Therefore, we moved forward to more advanced machine learning methods and suggested an *end-to-end pipeline* which allows to immediate tracking results based on a RGB image. By leveraging recent deep learning techniques, suitable cues are learned indirectly and render an explicit feature representation redundant. We further suggested to take advantage of the interdependency between the reference points and segmentation of the instrument. To this end, we reformulate the 2D pose estimation as a heatmap regression. The resulting deep learning architecture is a fully convolutional neural network with residual and skip connections that simultaneously regresses segmentation and reference points of the instrument. We evaluate the performance of both instrument representations on a Vitreoretinal and a Endoscopic Surgery dataset. Throughout the experiments our approach outperforms aforementioned pipelines as well as other state-of-the-art methods. It is important to note that this pipeline is a pure detection approach and does not employ any temporal information. Furthermore, it represents an end-to-end approach which takes the image as an input and does not require an initialisation. Consequently, this fully automated framework could directly be integrated in the image processing pipeline of the surgery. However, in contrast to the feed-forward and robust pipeline, it requires a GPU to achieve real-time performance.

In conclusion, we demonstrated both quantitatively and qualitatively that we can accurately track surgical instruments based on machine learning methods. Our experiments and evaluations are performed on *ex-vivo* and *in-vivo* datasets, including a variety of different instruments and illumination conditions. All presented pipelines have also been evaluated in a cross-validation setting to illustrate the generalization power and run in real-time. Every approach comes with its own advantages and disadvantages, as discussed in this chapter. Although these methods can achieve excellent accuracy, there are several possible extensions and future directions that will be discussed in the following chapter.

7 Future Work and Discussion

The topic of visual instrument tracking has been an active field of research for many groups in the past years and will become even more important in the future. With the expected future evolution of surgery [13], instrument tracking will be a key component for enabling active decision-support and computer-based surgical assistance. In this dissertation, I have presented several real-time approaches that build on state-of-the-art machine learning techniques and offer a multitude of possible future extensions.

To address the problem of limited diversity in the training data, we proposed a technique to actively adapt to the illumination conditions during the surgery. To this end, we changed the offline-learned Random Forest by incrementally adding online-learned trees to incorporate the experienced appearance changes. Although not a major limitation, one of the disadvantages of this technique is that the model thereby becomes wider and more complex with time. A natural direction would be to investigate the potential of effective convergence testing. However, this extension is not straightforward: By dropping the trees of the ensemble that do not contribute to the prediction, we could keep the model to a predefined size. At the same time, this could lead to an overfitting to the current scenario. The early-stopping criteria for efficient online learning would have to consider both aspects.

In the end-to-end pipeline, we present a deep learning-based approach for instrument tracking, which is computationally more expensive to train. Due to the real-time requirement, the learning process is static and relies on the quality of the training set. A future direction for this method could be to transfer the idea of online-adaption and develop a dynamic learning approach for the networks [115, 116, 117, 118]. The problem of the limited training data set could also be addressed by pre-training on simulated and rendered instrument configurations and adapting to real surgery sequences by fine-tuning the network on a few samples. Another direction would be to investigate the potential of temporal information for complexity reduction of the tracking problem [101]. Our end-to-end pipeline is a detection algorithm and does not make use of the fact that we are actually processing image sequences and not still images. Furthermore, the stochastic problem could be simplified by integrating knowledge about the instrument. In the presented approaches, we do not explicitly use the fact that the instrument is usually a rigid object. By integrating the kinematic constraints into the pipeline, the search space could be limited. Another possibility consists of leveraging the fact that modern surgical setups provide stereo image systems. If we can track the instrument in both images and establish their correspondence, we could estimate its 3D pose or enhance the depth perception for the surgeon.

A further issue to be mentioned is that the level of abstraction of deep neural networks leads to an impressive accuracy and applicability in many fields, but comes with the disadvantage that the learned features or the reasoning are not directly interpretable by humans. Indeed, the deep learning-based methods appear to be a black box [119]. Several works aimed at understanding and visualizing the intrinsics of convolutional networks [120, 121, 122, 123].

This understanding is in particular important for medical applications. Not only because physicians want to have an influence on the outcome, especially regarding diagnosis: in the context of medical applications an unexpected failure can lead to severe harm for the "end-user", i.e. the patient. In the case of instrument tracking, however, the situation is slightly different. For the surgeon, the method should be a black box in the sense that it should not require any interaction and the surgeon also does not need to understand why the instrument position was estimated at this location. As outlined in Section 2.3, the tracking information itself is not of direct interest to the surgeon and is rather employed as intermediate step for assistance. The algorithms should run in the background without any input from the surgeon that would interrupt the surgical workflow. One essential requirement, however, is that the method is a *reliable blackbox*. Depending on the application, the surgeon has to be able to trust the result of the tracking, e.g. if the iOCT is positioned according to the instrument location during membrane peeling. Consequently, the tracking quality has to reach a point where the method is robust to various scenarios or provide a backup-solution in case of failure. For a machine learning approach it is difficult to analyze its limits and determine the failure situations. Even if a proposed method follows the same evaluation strategy on the same dataset as another method, we can not necessarily deduce that a better reported performance corresponds to a better robustness in general. Recent works have shown that networks can be fooled even with modification of only a single pixel while achieving similar accuracy on the same datasets [124, 125, 126]. These so-called adversarial attacks are of course constructed and may not occur in a real surgeries. But they help to understand the behaviour and limits of neural networks.

Currently, the U.S. Food and Drug Administration (FDA) states that a machine learning-based method can only be considered for medical applications, if an extensive testing has statistically proven the repeatability of the results [127]. A cross-validation, as performed in this dissertation for all experiments, is an essential part of such an evaluation. However, it can only verify the performance of a method based on the information that is captured by the dataset. Unfortunately, the current publicly available datasets are quite limited and do not reflect the entire range of possible illumination scenarios, instruments, etc. To really leverage the power of deep learning and to provide statistical proof of general performance, an extensive, broad-ranging dataset is necessary. One major step into this direction was achieved in the field of Endoscopic Surgery with the publication of the dataset associated with the EndoVis Challenge 2015 [74] and 2017 [128]. In the field of Vitreoretinal Surgery, a dataset of comparable size is currently not available. I believe that an introduction of such a shared database would considerably push forward future research for instrument tracking in Vitreoretinal Surgery. The published datasets, such as the three sequences provided by [88], are unfortunately already quite saturated in terms of the tracking performance, which makes it difficult to demonstrate significant performance improvements of novel approaches.

I hope that the considerations and approaches presented in this dissertation will inspire the development of new methods and promote further research. However, transferring these technologies into clinical practice is challenging. I believe that future advances, not only in the field for instrument tracking, are most fruitful in active exchange and collaboration with both industrial and clinical partners. Surgical data science [13] is only at the beginning and I believe that our work has contributed to a promising direction.

Part IV

Appendix

Contributions

A

Surgical Tool Tracking and Pose Estimation in Retinal Microsurgery

Nicola Rieke[1], David Joseph Tan[1], Mohamed Alsheakhali [1], Federico Tombari[1], Chiara Amat di San Filippo[1], Vasileios Belagiannis[1], Abouzar Eslami[3], Nassir Navab[1,2]

[1] Chair for Computer Aided Medical Procedures (CAMP), TU Munich, Germany.
[2] Johns Hopkins University, Baltimore, USA.
[3] Carl Zeiss MEDITEC München, Germany.

Contribution. The main contributions of this publication, including the main idea of combining instrument tracking and pose estimation via specialized Random Forests, creation of the appearance dataset, tests, validation, and writing of the manuscript were done and coordinated by the author of this thesis. David Joseph Tan and Mohamed Alsheakhali were responsible for the implementation, description and validation of the template tracker. Annotations of the appearance dataset as well as the revision of the publication was done jointly together with the co-authors.

Surgical Tool Tracking and Pose Estimation in Retinal Microsurgery

Nicola Rieke[1], David Joseph Tan[1], Mohamed Alsheakhali[1], Federico Tombari[1,3], Chiara Amat di San Filippo[2], Vasileios Belagiannis[1], Abouzar Eslami[2], and Nassir Navab[1]

[1] Computer Aided Medical Procedures, Technische Universität München, Germany
[2] Carl Zeiss Meditec AG, München, Germany
[3] DISI, University of Bologna, Italy
Nicola.Rieke@tum.de

Abstract. Retinal Microsurgery (RM) is performed with small surgical tools which are observed through a microscope. Real-time estimation of the tool's pose enables the application of various computer-assisted techniques such as augmented reality, with the potential of improving the clinical outcome. However, most existing methods are prone to fail in *in-vivo* sequences due to partial occlusions, illumination and appearance changes of the tool. To overcome these problems, we propose an algorithm for simultaneous tool tracking and pose estimation that is inspired by state-of-the-art computer vision techniques. Specifically, we introduce a method based on regression forests to track the tool tip and to recover the tool's articulated pose. To demonstrate the performance of our algorithm, we evaluate on a dataset which comprises four real surgery sequences, and compare with the state-of-the-art methods on a publicly available dataset.

1 Introduction

Retinal Microsurgery (RM) is a delicate medical operation which requires extremely high handling precision of the utilized surgical instruments. Usually, the visual control for the surgeon is restricted to a limited 2D field of view through a microscope. Problems such as lens distortions and the lack of depth information or haptic feedback complicate the procedures further. Recent research aimed at assisting the surgeon by introducing smart imaging such as the Optical Coherence Tomography (OCT) [1], which visualizes subretinal structure information. In the current workflow, these devices have to be manually positioned on the region of interest, which is usually close to the tool tip. The ability of extracting the position of the surgical tool tip in real-time allows to carry out this positioning automatically. Other applications that require tool tracking include surgical motion analysis and visual servoing. The estimation of the articulated pose of the tool rather than its position alone allows us to measure the size of the anatomical structures in the video sequence. Additionally, it paves the way for advanced augmented reality applications which provide, for example, proximity

information of the tool tips to the retina. Despite recent advances, the vision based tracking of the tool tip's location in *in-vivo* is still challenging, mainly due to lighting variation and variable instrument appearances. Moreover, tracking has to be real-time capable in order to be employed during a surgical procedure. These challenges have been addressed with different approaches, including color-based [2] and geometry-based methods [3–5]. Other relevant works [6–8] focus on a specific tool model (e.g. vitrectomy or closed forceps). The learning-based approach from Sznitman et al. [9] introduces the combination of a tool detector, which relies on deformable feature learning, and a simple gradient tracker. Li et al. [10] present an instrument tracking method based on online learning. Both methods [9, 10] achieve accurate results for *in-vivo* RM sequences and their implementation runs at video frame-rate. However, the tracking is restricted to the center joint of the surgical forceps and does not localize the two tips of the instrument, which are extremely important for the surgeon. In contrast, the learning-based method published by Pezzementi et al. [11] yields the pose of the surgical tool, but relies on a high amount of labeled data in order to capture the appearance changes of the instrument.

This paper introduces an alternative visual tracking approach that goes beyond phantom data. It can handle incomplete and noisy data by building on regression forests to yield the positions of the tool tips as well as the center point of the forceps in *in-vivo* RM sequences in real-time. First, the tracking algorithm finds a bounding box around the tool tip by estimating the relation between the instrument motion and changes induced in the image intensities. Successively, the pose estimation localizes the points of interest within this region by evaluating a learned mapping between image patches and the articulated pose. To the best of our knowledge, modeling the localization of articulated surgical forceps as simultaneous tracking and pose estimation, is a novel approach in real *in-vivo* microscopic surgery. Throughout experimental results, we demonstrate how the proposed method is able to withstand challenging environments characterized by variable illumination and noise, as well as to yield the pose of various forceps types in real-time. A comparison with the state of the art on a public benchmark demonstrates further the performance of our method.

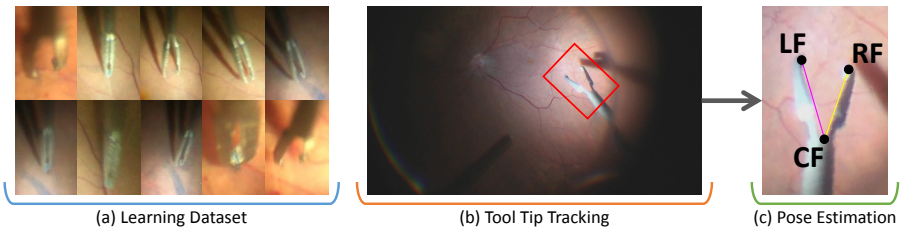


Fig. 1: Left: Learning set with various templates including open and closed forceps. Right: Tracking in the frame and pose estimation within the reduced region.

2 Proposed Method

The overall goal is to localize the three joint points of the forceps for every frame in real-time. As the image quality is poor in general, tracking the three points independently is prone to fail. We propose a method that breaks down the difficulty into two separate tasks. Our pipeline begins with a template tracking algorithm (Sec. 2.2) that estimates a bounding box around the tool tip in the current frame. In the second step, a pose estimation algorithm (Sec. 2.3) localizes the three points of interest within this region. Both algorithms are based on regression forests (Sec. 2.1). In case of tracking, the objective is to regress the location of the bounding box, while during pose estimation the task is to regress the points of interest. The typical input and output data of the entire algorithm is shown in Fig. 1.

2.1 Regression Forest

From a generic input $\mathbf{X}$ and output $\mathbf{Y}$, the regression forest is used to learn the relation of $\mathbf{X}$ and $\mathbf{Y}$ such that, given the input $\mathbf{X}$, the forest can predict the output $\mathbf{Y}$. Every tree of the forest is defined by a set of branches and leaves. At each branch, a binary splitting function θ determines if a training sample of $\mathbf{X}$ goes to the left P_l or right P_r subset of samples. During *training* the splitting functions are selected in a way that maximizes the information gain $g(\theta)$ by optimally splitting the training samples of the node. The information gain is given as:

$$g(\theta) = H(P) - \sum_{i \in \{l,r\}} \frac{|P_i(\theta)|}{|P|} H(P_i(\theta)), \tag{1}$$

where $H(\cdot)$ is the entropy. Splitting aims to divide the learning dataset into smaller subsets through $\mathbf{X}$ while optimizing for $\mathbf{Y}$ by making the parameters of the individual subsets more coherent. Based on the above rule, the tree grows by iteratively applying the same splitting process and stops when the number of samples $|\mathcal{P}|$ is less than a threshold, the best information gain is less than a threshold or the maximum depth is reached. All final nodes are considered as leaves and store the prediction, i.e. a statistical representation of all $\mathbf{Y}$ that reached this node. The same scheme is followed for a number of trees in order to build a forest. During *prediction*, the branches look at the splitting function to navigate a sample of $\mathbf{X}$ towards the left or right child until it reaches a leaf that gives the corresponding prediction. After taking the predictions from different trees, an average prediction is computed as the final output.

2.2 Tracking

Given a sequence of images $\{I_t\}_{t=0}^{n_t} = \{I_0, I_1, \ldots, I_{n_t}\}$, the proposed tracking approach learns from the initial image I_0 — where we assume to be given a rectangular template around the tool tip — and then propagates the learned model of the template in the following frames. In more detail, within such template,

n_s sample points $\{\mathbf{x}_0^s\}_{s=1}^{n_s}$ are randomly selected such that the template can be described using the intensity vector $\mathbf{i}_0 = [I_0(\mathbf{x}_0^s)]_{s=1}^{n_s}$. The objective of frame-to-frame tracking is to find the 2D translation vector $\delta\boldsymbol{\mu}$ that updates the location of the tool based on the intensities in the current frame $\mathbf{i}_t = [I_t(\mathbf{x}_{t-1}^s)]_{s=1}^{n_s}$ using the location of sample points from the previous frame $\{\mathbf{x}_{t-1}^s\}_{s=1}^{n_s}$. The tracker [12] learns the relation of the changes in the intensities $\delta\mathbf{i} = \mathbf{i}_t - \mathbf{i}_0$ and the transformation parameters $\delta\boldsymbol{\mu}$. To model the structure of the forest based on Sec. 2.1, we set the input $\mathbf{X} = \delta\mathbf{i}$, the parameters $\mathbf{Y} = \delta\boldsymbol{\mu}$ and the function $H(\cdot)$ as the standard deviation, while the leaves store the mean and standard deviation of the parameters that arrive on that node. It is noteworthy to mention that since $\mathbf{i}_0$ is constant and the branch only compares an index of the vector $\delta\mathbf{i}$, we can simplify the forests by learning the relation of $\mathbf{i}$ and $\delta\boldsymbol{\mu}$ instead of $\delta\mathbf{i}$ and $\delta\boldsymbol{\mu}$. This enables the tracker to directly look at the intensities $\mathbf{i}$, without explicitly taking into account the intensities of the template $\mathbf{i}_0$. This brings in an important benefit, since it allows the tracker to alleviate from the restriction of tracking a single template, and to use multiple templates within the same forests. Therefore, differently from [12], we propose here to correlate multiple templates to compensate for the motion of the tool tip as it opens and closes, as well as for the strong illumination changes and photometric distortions that are typically present in such working conditions. Nevertheless, the proposed approach is still able to yield an efficiency of less than 2 ms per frame, using only one CPU core. To compensate for a possible loss in tracking, we impose a confidence measure using the average standard deviation of the predicted leaves from different trees of the forest. During learning, a tree recursively splits the dataset into two subsets such that the parameters in each subset have a lower standard deviation. Henceforth, a confident prediction must have an average standard deviation less than a threshold τ_σ. If the prediction is confident, the location of the template is updated using $\delta\boldsymbol{\mu}$; otherwise, its previous location is propagated to the next frame.

2.3 Pose Estimation

Given the bounding box $I_B \subset I_t$ around the tool tip from the tracking, the pose estimation algorithm localizes three joints within this region. By considering the surgical tool as an articulated object, we can transform the problem of localizing the tool parts into a task that was successfully addressed in the area of human pose estimation [13], which can predict the pose in very challenging scenarios with occlusion and noisy data. In order to integrate this method in the tool pose estimation, we define the set of joints as the tip of the left part of the fork (LF), the tip of the right part of the fork (RF) and the connecting center part of the fork (CF) (compare Fig. 1): $\mathbf{Y} = \{LF, RF, CF\} \subset \mathbb{R}^2 \times 3$. As the input space for the tree, we set $\mathbf{X}$ to be the HOG features of randomly selected image patches with associated joint offsets. The binary split function θ divides the samples based on a threshold in one dimension of $\mathbf{X}$. The function $H(\cdot)$ is chosen to be the sum-of-squared-differences. As a result, the offsets of all instrument joints $y \in \mathbf{Y}$ are stored in each leaf of each tree. During the prediction, image patches

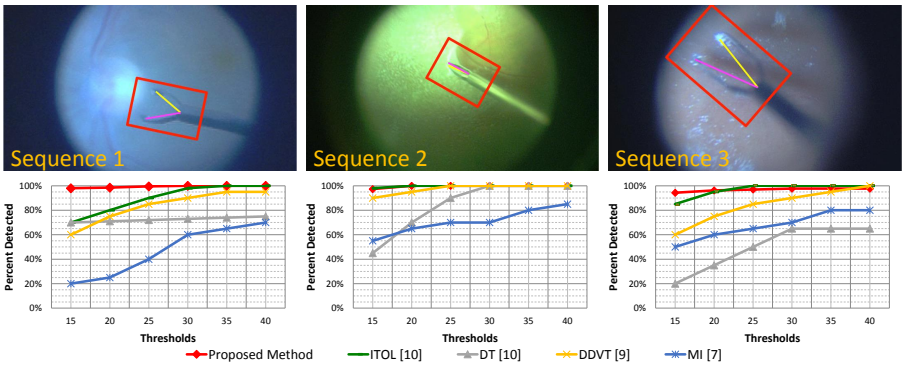


Fig. 2: Results for each sequence of the *public* dataset, when learned and tested on separate sequences, in terms of Threshold Score, as performed in [10]. Top: a qualitative example of detected bounding box and tool pose for each sequence.

are extracted from random pixel positions within the bounding box I_B. In order to combine the votes of the different trees and to find the most probable location of the joint, a greedy dense-window algorithm is applied, as in [13]. The final output of the pose estimation step are the 2D coordinates of each joint in $\mathbf{Y}$.

3 Experiments and Results

The experimental validation of the proposed algorithm is carried out on two different Retina Microsurgery (RM) datasets: the first one, referred to as the *public* dataset [9], is a fully annotated dataset of three different sequences of *in-vivo* vitreoretinal surgery. It comprises 1171 images with a resolution of 640x480 pixels each. The main difficulty of this dataset consists in the presence of noise and shadow as well as variable illumination conditions. The forceps type is the same in all sequences. The second one, referred to as the *appearance* dataset, is a new dataset comprising four real *in-vivo* RM surgeries with 200 manually annotated consecutive images at 1920x1080 pixels of resolution each. This dataset is challenging since it includes different types of forceps, as well as different illumination conditions and microscope zoom factors.

The performance of the algorithm on these datasets was evaluated by means of two different metrics: the strict Percentage of Correct Parts (strict PCP) [14] and the Threshold Score used by Sznitman et al. [9]. The Threshold Score recovers the pixel-wise aspect of the quality for the predicted localizations, i.e. a prediction for the position of a joint is evaluated as correct if the pixel distance to the ground truth is smaller than a threshold. The strict PCP score is a standard metric in human pose estimation and addresses the length of the connected joints of the model. Considering two connected joints, a prediction is evaluated as correct if the distances between the predicted localization and the ground truth for the joints are both smaller than $\alpha \in \mathbb{R}$ times the corresponding ground

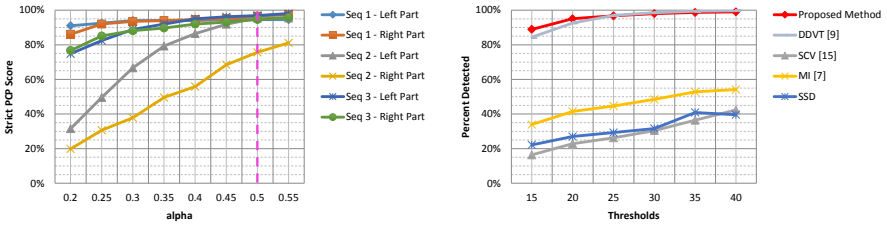


Fig. 3: Left: Strict PCP scores for testing on separate sequences of the *public* dataset. The vertical line indicates the standard α value in human pose estimation [14]. Right: Threshold Score comparison to other methods when tested on full dataset. Results for referenced methods are given by [9].

truth length of the connection (in the field of human pose estimation, usually $\alpha = 0.5$ [14]). Our method is implemented in C++ and runs at 30 fps on an off-the-shelf computer. For the tracker we used 500 sample points and 90 trees for the *public* dataset and 100 trees for the *appearance* dataset. Based on the results in [13], we have set the number of trees for the pose estimation to 15, the HOG features bin size to 9 and the patch size resolution to 50x50 pixels.

3.1 Public Dataset Evaluation

We compare our method with the state-of-the-art methods DDVT [9], MI [7], SCV [15] and ITOL [10]. Analogously to results presented on such works for this dataset [9, 10], we evaluated the pixel-wise measure for the center joint for thresholds between 15 and 40 pixels. First, we evaluated the algorithm for every sequence separately by training the regression trees on the first half of the sequence and testing on the second half. Our method outperforms the baseline methods, reaching over 94% prediction rate in every sequence (Fig. 2). Even the inclusion of all the first halves of the sequences in one training dataset results in higher detection rates of our method than the state-of-the-art methods when tested on the unseen halves (Fig. 3a). In terms of strict PCP score, it can be observed that the length of the forceps parts, thus also the tool tip joints, are predicted correctly even for α values below the standard measure. In contrast to the other methods, our algorithm is able to reliably track the instrument over the entire sequence without the need of reinitialisation.

Table 1: Strict PCP for Appearance Dataset for $\alpha = 0.5$.

	Set 1	Set 2	Set 3	Set 4
Left Part	69.70	93.94	94.47	46.46
Right Part	58.58	93.43	94.47	57.71

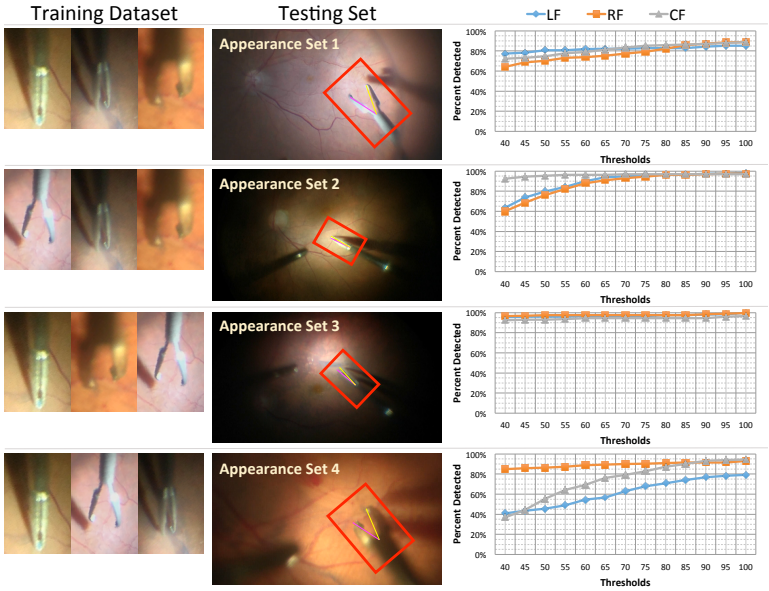


Fig. 4: Results for the cross-validation experiment on the *appearance* dataset. The first column shows examples of the respective training dataset and the second shows one image from the testing set. The threshold score on the right indicates the percentage of correctly predicted locations for the different joints.

3.2 Appearance Dataset Evaluation

Learning-based methods tend to fail on data which comprises image content that is not seen in the training dataset. In this section we show that the proposed method can generalize from different illumination conditions, zoom factors and noise levels. In contrast to the *public* dataset, this dataset includes various forceps types. The experiment was performed in a 4-fold leave-one-out fashion, i.e. training the forests each time on three sequences and testing on the remaining one. Since the average tool shaft diameter is 50 pixel for this dataset, we evaluated the threshold measure for values between 40 and 100 pixels. The results are summarized in Fig. 4. For every sequence, the tracker only had to be reinitialized once. The strict PCP results for $\alpha = 0.5$ are depicted in Table 1 with a mean strict PCP score of 76% for both the left and right part of the forceps.

4 Conclusions

We presented a novel approach that simultaneously predicts location and pose of surgical forceps in *in-vivo* RM sequences at 30 fps. This paper demonstrates the algorithm's capability to estimate the correct locations even in challenging situations as well as to generalize to unseen tools. Moreover, our experimental results indicate that our approach outperforms state-of-the-art methods.

References

1. Balicki, M., Han, J.H., Iordachita, I., Gehlbach, P., Handa, J., Taylor, R., Kang, J.: Single fiber optical coherence tomography microsurgical instruments for computer and robot-assisted retinal surgery. In: Yang, G.-Z., Hawkes, D., Rueckert, D., Noble, A., Taylor, C. (eds.) MICCAI 2009. LNCS, vol. 5761, pp. 108–115. Springer, Heidelberg (2009)
2. Allan, M., Ourselin, S., Thompson, S., Hawkes, D.J., Kelly, J., Stoyanov, D.: Toward detection and localization of instruments in minimally invasive surgery. In: IEEE Transactions on Biomedical Engineering 60, pp. 1050 – 1058 (2013)
3. Baek, Y.M., Tanaka, S., Kanako, H., Sugita, N., Morita, A., Sora, S., Mochizuki, R., Mitsuishi, M.: Full state visual forceps tracking under a microscope using projective contour models. In: Proc. of IEEE ICRA, pp. 2919–2925 (2012)
4. Reiter, A., Allen, P.K., Zhao, T.: Feature classification for tracking articulated surgical tools. In: Ayache, N., Delingette, H., Golland, P., Mori, K. (eds.) MICCAI 2012, Part II. LNCS, vol. 7511, pp. 592–600. Springer, Heidelberg (2012)
5. Wolf, R., Duchateau, J., Cinquin, P., Voros, S.: 3d tracking of laparoscopic instruments using statistical and geometric modeling. In: Fichtinger, G., Martel, A., Peters, T. (eds.) MICCAI 2011, Part I. LNCS, vol. 6891, pp. 203–210. Springer, Heidelberg (2011)
6. Sznitman, R., Basu, A., Richa, R., Handa, J., Gehlbach, P., Taylor, R.H., Jedynak, B., Hager, G.D.: Unified detection and tracking in retinal microsurgery. In: Fichtinger, G., Martel, A., Peters, T. (eds.) MICCAI 2011, Part I. LNCS, vol. 6891, pp. 1–8. Springer, Heidelberg (2011)
7. Richa, R., Balicki, M., Meisner, E., Sznitman, R., Taylor, R., Hager, G.: Visual tracking of surgical tools for proximity detection in retinal surgery. In: IPCAI, pp. 55–66 (2011)
8. Sznitman, R., Becker, C., Fua, P.: Fast part-based classification for instrument detection in minimally invasive surgery. In: Golland, P. et al. (eds.) MICCAI 2014. LNCS, vol. 8673, pp. 692–699. Springer, Heidelberg (2014)
9. Sznitman, R., Ali, K., Richa, R., Taylor, R.H., Hager, G.D., Fua, P.: Data-driven visual tracking in retinal microsurgery. In: Ayache, N., Delingette, H., Golland, P., Mori, K. (eds.) MICCAI 2012, Part II. LNCS, vol. 7511, pp. 568–575. Springer, Heidelberg (2012)
10. Li, Y., Chen, C., Huang, X., Huang, J.: Instrument tracking via online learning in retinal microsurgery. In: Golland, P. et al. (eds.) MICCAI 2014. LNCS, vol. 8673, pp. 464–471. Springer, Heidelberg (2014)
11. Pezzementi, Z., Voros, S., Hager, G.D.: Articulated object tracking by rendering consistent appearance parts. In: ICRA 2009, pp. 3940–3947 (2009)
12. Tan, D.J., Ilic, S.: Multi-forest tracker: A chameleon in tracking. In: CVPR 2014, pp. 1202–1209 (2014)
13. Belagiannis, V., Amann, C., Navab, N., Ilic, S.: Holistic human pose estimation with regression forests. In: In: Perales, F.J., Santos-Victor, J. (eds.) AMDO 2014. LNCS, vol. 8563, pp. 20–30. Springer, Heidelberg (2014)
14. Ferrari, V., Marin-Jimenez, M., Zisserman, A.: Progressive search space reduction for human pose estimation. In: CVPR, pp.1–8 (2008)
15. Pickering, M.R., Muhit, A.A., Scarvell, J.M., Smith, P.N.: A new multi-modal similarity measure for fast gradient-based 2d-3d image registration. In: EMBC 2009,pp. 5821–5824 (2009)

Real-Time Localization of Articulated Surgical Instruments in Retinal Microsurgery

Nicola Rieke[1], David Joseph Tan[1], Chiara Amat di San Filippo[1], Federico Tombari[1,3], Mohamed Alsheakhali[1], Vasileios Belagiannis[1], Abouzar Eslami[4], and Nassir Navab[1,2]

[1] Chair for Computer Aided Medical Procedures (CAMP), TU Munich, Germany.
[2] Johns Hopkins University, Baltimore, USA.
[3] DISI, University of Bologna, Italy
[4] Carl Zeiss MEDITEC München, Germany.

Contribution. The main contributions of this publication, including the resolution independent evaluation metric, extension of the appearance dataset, tests, validation, and writing of the manuscript were done and coordinated by the author of this thesis. David Joseph Tan was responsible for the description and validation of the template tracker. Annotations of the appearance dataset as well as the revision of the publication was done jointly together with the co-authors.

Contents lists available at ScienceDirect

Medical Image Analysis

journal homepage: www.elsevier.com/locate/media

Real-time localization of articulated surgical instruments in retinal microsurgery

Nicola Rieke[a,*], David Joseph Tan[a], Chiara Amat di San Filippo[a,c], Federico Tombari[a,d], Mohamed Alsheakhali[a], Vasileios Belagiannis[b], Abouzar Eslami[c], Nassir Navab[a]

[a] Computer Aided Medical Procedures, Technische Universität München, Germany
[b] Visual Geometry Group, Department of Engineering Science, University of Oxford, Great Britain
[c] Carl Zeiss Meditec AG, München, Germany
[d] DISI, University of Bologna, Italy

ARTICLE INFO

Article history:
Received 14 January 2016
Revised 3 May 2016
Accepted 3 May 2016
Available online 13 May 2016

Keywords:
Pose estimation
Visual tracking
Retinal microsurgery

ABSTRACT

Real-time visual tracking of a surgical instrument holds great potential for improving the outcome of retinal microsurgery by enabling new possibilities for computer-aided techniques such as augmented reality and automatic assessment of instrument manipulation. Due to high magnification and illumination variations, retinal microsurgery images usually entail a high level of noise and appearance changes. As a result, real-time tracking of the surgical instrument remains challenging in *in-vivo* sequences. To overcome these problems, we present a method that builds on random forests and addresses the task by modelling the instrument as an articulated object. A multi-template tracker reduces the region of interest to a rectangular area around the instrument tip by relating the movement of the instrument to the induced changes on the image intensities. Within this bounding box, a gradient-based pose estimation infers the location of the instrument parts from image features. In this way, the algorithm does not only provide the location of instrument, but also the positions of the tool tips in real-time. Various experiments on a novel dataset comprising 18 *in-vivo* retinal microsurgery sequences demonstrate the robustness and generalizability of our method. The comparison on two publicly available datasets indicates that the algorithm can outperform current state-of-the art.

1. Introduction

Retinal Microsurgery (RM) is a delicate surgical procedure, which requires high handling precision for the utilized instruments under limited visual feedback. In routine surgery such as membrane peeling, the surgeon has to manipulate anatomical features (*layers*) which are less than 10 \mu m thick. One of the main difficulties is caused by the fact that the surgeon can only observe the procedure in an indirect way through a microscope. The interpretation of the perceived depth then becomes quite challenging and the high magnification leads to lens distortions such that, in most cases, only a portion of the observed scene is focused. Furthermore, the haptic feedback is weak. All these problems limit the possibilities for the surgeon to identify or grasp surgical targets and consequently may increase operating time and the risk for retinal damage.

Recently, new microscopes introduced on the market provide additional intraoperative imaging information to the surgeon, by visualizing subretinal structure information via Optical Coherence Tomography (OCT). OCT imaging has become widespread in ophthalmology over the past 20 years because of its ability to visualize ocular structures at high resolution (Gabriele et al., 2011). Its new intraoperative version (iOCT) has opened up new research fields and paved the way to new applications (Ehlers et al., 2014), thanks to the fact that depth information within the tissue can be obtained in real-time during the procedure. In the current workflow, these devices have to be manually positioned on the region of interest which further increases the complexity of the device handling for the surgeons, who already have to manipulate the surgical tools, the manual light source and the microscope.

Recent works have been aiming at introducing specific computer vision algorithms in order to overcome the current technical limitations and support the surgeon in a direct or indirect way. For instance, the visual data acquired from the microscope can be processed for stitching different frames together, in order to create a wider field of view of the retina, e.g., by Cattin et al. (2006);

* Corresponding author.
E-mail address: nicola.rieke@tum.de (N. Rieke).

http://dx.doi.org/10.1016/j.media.2016.05.003
1361-8415/

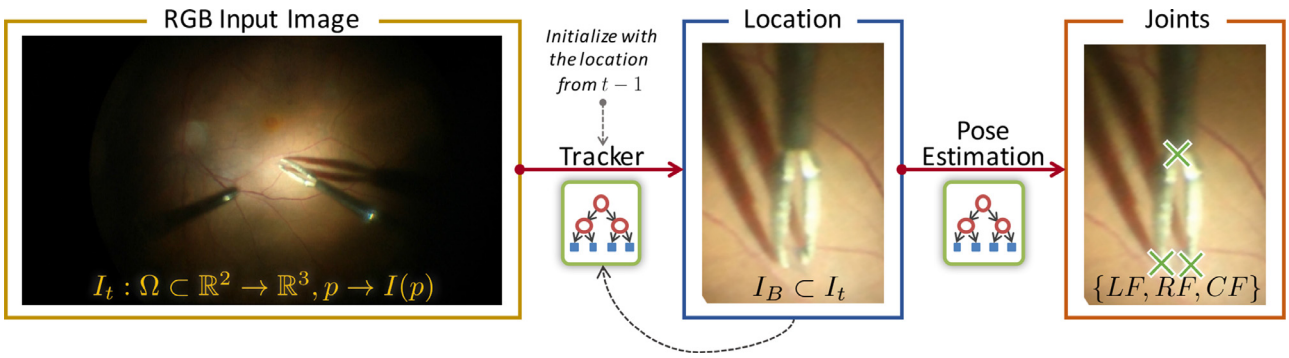


Fig. 1. Pipeline: The figure illustrates an overview of the algorithm. Given a frame of a video sequence at time t, the temporal tracker determines the region of interest around the tool tip. It takes the bounding box from the previous frame and iteratively refines its location. Thereafter, the bounding box is used as input to the pose estimation to find the positions of the three joints – left joint (LF), right joint (RF), central joint (CF).

or, for providing landmarks on the retina, but also simultaneously tracking surgical tool and its shadow as presented by Yigitsoy et al. (2015). Within this field, the capability of robustly tracking the surgical instrument at each frame represents a key component for various applications.

There are different applications that can benefit from efficient and robust tool tracking such as the one proposed in this paper. First, the algorithm allows tracking the position of the tools and estimate its trajectory, in order to compare the movements of an expert physician to a non-expert for training and for assessing the quality of the surgeries, as introduced by Blum et al. (2007). Secondly, the pose estimation of the articulated object is a particularly crucial step for further applications as automatic grasp-counting. Note that the number of grasps is fundamental in retinal surgery (Pavlidis et al., 2015) and should be minimized because the tissue can be easily damaged. Third, we consider also the usability of the microscope. Since the physician has to operate with both ocular and food pedal to navigate the microscope options, smoothing the workload can lead to quicker and better surgical result, as specific movements of the tool can be linked with the (de-)activation or modification of functionalities such as zooming, lighting or iOCT automatic positioning (Rieke et al., 2016). Moreover, the position of the tool is also the missing link to advanced augmented reality applications, such as providing the surgeon with additional information regarding the proximity of the tool tip to the retina. In Roodaki et al. (2015), the distance of the instrument from the retina can be calculated with the help of a tool tracker, and can visually inform the operator in case of risk of retina damage. Importantly, for the integration into the surgical workflow, it is necessary that the tracking algorithm achieves real-time efficiency.

1.1. Benefit of the instrument pose in addition to the instrument position

The position of the instrument in real-time is a valuable information during RM. However, the surgeon's center of attention is usually close to the surgical tool tips, which are more challenging to detect due to its opening and closing movement. Providing the position of the instrument's tips in real-time rather than the position of the central joint (e.g., as done in the work of Li et al., 2014 or Sznitman et al., 2012) can pave the way for advanced computer-aided support. One example is the positioning of the intraoperative OCT (iOCT) during membrane peeling. Usually, the surgeon needs the distance and depth information of the layers of the retina at the point where the tool tips grasp the membrane. In the current workflow, the iOCT position has to be adjusted manually, which is time consuming and also increases the complexity of handling the various instruments during a procedure. The ability of extracting the location of the surgical tool tips allows carrying out the positioning of the iOCT in an automatic way (Rieke et al., 2016). Furthermore, additional information such as proximity information of the tool tips to the retina can be visualized close to the instrument tips, so that the surgeon does not have to switch between different visualization modalities. By knowing the pixel distance of the tool tips to the joint point of the forceps in an image, the physical distance can be inferred and characteristics of anatomical structures can be measured directly in the visual data. The location of the instrument parts relative to each other can also provide important information about the state of the surgical workflow. All these aims are not achievable by measuring only the location of the center joint of the forceps.

1.2. Contributions

Despite the importance of estimating the location of the tool tips, most existing methods recover only the center joint of the instrument and are tested on synthetic data or only on a small dataset of RM sequences. In this work, we go beyond phantom data and propose a method for real-time tracking and pose estimation of surgical instruments in *in-vivo* microsurgery images, which estimates not only the position of the forceps central joint, but also the position of the instrument tips. Preliminary results of this work appeared in Rieke et al. (2015). The algorithm is inspired by state-of-the-art computer vision approaches and handles the aforementioned difficulties by modelling the problem as two different tasks: tracking and pose estimation (see Fig. 1). First, the tracking algorithm reduces the considered image information to a rectangular region containing the tool tip. In the second step, the pose estimation algorithm estimates the location of the instrument parts inside the bounding box. Both algorithms employ random forest in order to cope with noisy and incomplete data which result from the various appearance and illumination changes, but rely on different image information. By combining these two different algorithms, we use both color and gradient information for predicting the positions of the instrument parts.

In contrast to the work of Rieke et al. (2015) where only the grayscale information was used, the tracker in this work is now based on the entire RGB space. Another main contribution is the introduction of a novel, extended dataset of *in-vivo* RM sequences, which allows us to perform various detailed experiments evaluating the performance of the proposed algorithm regarding generalization to different tool types and different background conditions. These experiments could not be carried out extensively with the original dataset presented in Rieke et al. (2015), as it included only one sequence per tool type. In addition, we compared the performance of our method on this novel dataset to two methods: the online tracker TLD and the offline learned FPBC for retinal microsurgery by Sznitman et al. (2014). Due to the variations in instrument size in our new dataset, the performance of the algorithm is no longer comparable across sequences via the standard

performance measure which is based on pixel distances. Therefore, we also introduce a new metric for evaluating the prediction for the forceps joint which takes into account the variation of instrument shapes and different image resolutions. Furthermore, an additional contribution is the extension of the annotations on a public available laparoscopic dataset to pose information. Thereby, we can compare our algorithm to state-of-the-art methods on two published dataset – the RM sequence dataset and the laparoscopic instrument sequence.

2. Related work

Despite recent advances, the vision-based tracking of *in-vivo* sequences remains challenging – the strong illumination changes and the noise level of the images are the most prominent difficulties while the various appearances changes of the surgical instrument complicate the task further.

Prior work addressing these challenges has considered the use of geometric models such as (Baek et al., 2012) proposed an approach to track the forceps by generating a database of the projected contours of a 3D CAD model of the robotic forceps. The likelihood to the projected contour of the microscopic image is measured and finally the full state of the forceps is estimated via particle filtering. They evaluate this approach and demonstrate its robustness and efficiency on synthetic data of a simulated surgical environment. Reiter et al. (2012) presented a tracking method, which relies on the appearance of natural landmarks. They trained an efficient multi-class classifier and as the location of the natural landmarks are known in the tool's CAD model, they are used to compute the final pose. The algorithm was tested on five endoscopic sequences. Color-based approaches were presented by Allan et al. (2013, 2015) for the related field of laparoscopic tool tracking. Other relevant work (e.g. Richa et al., 2011) presents results on both phantom and *in-vivo* data, using a two stage procedure: brute-force search of the tool tip in the surroundings of the instrument coordinates in the previous frame; and, weighted mutual information to optimize the initial guess.

Most recent works build on learning based approaches like (Chen et al., 2013) who use natural features of surgical instruments for tracking and adopt a spiking neural network to recognize the instrument tip in laparoscopic surgeries. (Li et al., 2014) proposed an online learning approach for tool tracking in RM. The system starts with a manual initialization and gradually builds the database for tracking by adding new positive and negative tool samples, which are collected by a filtering process. The algorithm provides an accurate bounding box around the forceps' central point, but does not localize the two tips of the instrument. In Pezzementi et al. (2009), a phantom is employed, using a half-sphere, painted to resemble the retinal surface. This learning-based method is based on creating a model by hand-segmenting the instrument, where experiments have shown that usually one or two frames are sufficient. Rigid tool tracking is performed over two steps: first via appearance modelling, which computes a pixel-wise probability of class membership (foreground/background), then filtering, which estimates the current tool configuration. The proposed method of Sznitman et al. (2011) utilizes a parametrization of the surgical tool considering the following three criteria: the location of the insertion point, the angle between image boundary and tool, and the tool length. Afterwards, tracking is considered as a Bayesian filtering estimation problem. To compute the necessary posterior distribution, they use a strategy based on active testing (ATF). Their dataset consists in two sequences on a retinal phantom using a needle. In their most recent work, (Sznitman et al., 2014) proposed a robust and efficient algorithm which uses a multi-class classifier based on boosted regression trees. Each class represents a different part of the instrument (e.g. center, insertion point, shaft) or background. In order to provide both accuracy and good frame rate, an early stopping method was also implemented using a probabilistic model, which evaluates the reliability of current classification, and stops in case no more computation is necessary.

Many different works have been proposed in the RM field, but most of them are constrained and cannot uphold to real-world applications. For instance, some methods (Baek et al., 2012; Sznitman et al., 2011; Pezzementi et al., 2009) are only evaluated on synthetic data. In others, the (CAD-) model of the surgical instrument was given (Reiter et al., 2012). But this is not possible in RM because the tool are often changed and 50+ models are available on the market. Moreover, in Pezzementi et al. (2009), the instrument needs to be hand segmented in the first frame, and we fear this approach can increase surgery time. The works of Sznitman et al. (2011); 2014) use the insertion point as parameter to define the tool. However, in many cases when evaluating on our novel dataset (see Section 5.2), the point is not well-defined and their performance showed it to be the most challenging identifiable point of all classes. Allan et al. (2015) utilize the optical flow technique, which did not provide reliable results on the novel dataset due to the strong illumination changes in RM. Moreover, most of the works are not focusing on the exact coordinates of the tool tips (e.g. Richa et al., 2011; Sznitman et al., 2012; Li et al., 2014; Sznitman et al., 2014), which is a crucial step as discussed in Section 1.1.

3. Method

In this section, we present details of the proposed algorithm. The overall goal is the location of the instrument parts for every frame of an *in-vivo* RM sequence in real-time. As previously mentioned, due to the challenging nuisances normally present on the images, the independent tracking of the tool parts proves to be difficult. Furthermore, appearance changes of the instrument and strong illuminations variations result in incomplete and noisy data. Random forests (see Section 3.1) have shown to be able to handle these problems and even generalize to unseen situations. Therefore, we propose an algorithm relying on random forests, but tackling the task by means of two separate steps, so to focus the algorithms in exploiting different available information that are essential to solve the distinct problems in tracking and pose estimation. The overall pipeline is as follows (see Fig. 2): we assume, as input, an RGB-valued image $I : \Omega \subset \mathbb{R}^2 \rightarrow \mathbb{R}^3, p \rightarrow I(p)$ along with the initial localization of the tool. For every new frame I_t, the tracking algorithm (see Section 3.2) estimates the transformation that updates the location of a bounding box $I_B \subset I_t$ containing the entire instrument tip. As a RGB-based frame-to-frame tracker, it exploits spatio-temporal information of the template in the previous frame and relies on the contrast between the instrument and the retina. The pose estimation (see Section 3.3) then regresses the locations of the points of interest within this region I_B, which define the articulated pose of the instrument. In contrast to the tracker, it employs gradient information and is completely independent from the results of previous frames. The specific details of our algorithm will be given in the following sections.

3.1. Random forest

A crucial part of our method is the random forest, which is a machine learning method used in the tracking and in the pose estimation stage of our algorithm. Considering an input **X** and output **Y**, the random forest is used to learn the relation of **X** and **Y** such that, given the input **X**, the forest can predict the output **Y**. A forest itself consists of an ensemble of decision trees which can output a class (classification) or real numbers (regression) as in our case. In particular, random forests are used to correct the tendency of trees

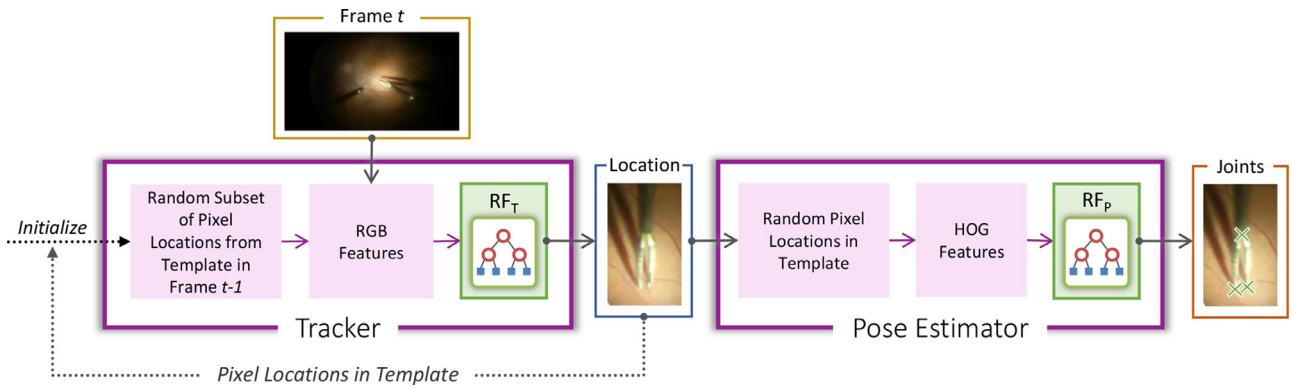


Fig. 2. Information processing: The figure illustrates the information processing of the algorithm. The algorithm relies on random forest and addresses the problem in two stages: tracking and pose estimation. The tracking random forest employs RGB intensity information whereas the pose estimation forest uses HOG features for estimating the locations of the joints.

to overfit the training set. Moreover, while predictions of a single tree are highly sensitive to noise in the training set, the average of many trees is not, under the hypothesis that they are independent. Each classifier at each node of the tree represents a "weak learner", while the ensemble of all such weak classifiers make the forest more confident to predict **Y**. Further information on random forest can be found in the original work of Breiman (2001).

A tree is composed of nodes that can either be a branch which has two children (i.e., left and right) or a leaf which is the terminal node. Training a tree requires a learning dataset $P = \{(\mathbf{X}_d, \mathbf{Y}_d)\}_{d=1}^{n_d} = \{(\mathbf{X}_1, \mathbf{Y}_1), (\mathbf{X}_2, \mathbf{Y}_2), \ldots, (\mathbf{X}_{n_d}, \mathbf{Y}_{n_d})\}$ with the input $\mathbf{X}_d$ and its corresponding ground truth observation $\mathbf{Y}_d$. During *training* at each node, we split the data into two subsets to be passed down to its children (P_l, P_r), based on a splitting criterion θ. The splitting criterion is a pool of random tests and has the goal, at each step, to find the best split of the set. In our work, we employ the information gain for evaluating the best split, which is given as

$$g(\theta) = H(P_n) - \sum_{i\in\{l,r\}} \frac{|P_i(\theta)|}{|P_n|} H(P_i(\theta)), \tag{1}$$

where P_n is the set of samples that reached the node n, $|P|$ is the number of samples in the set P and $H(\cdot)$ evaluates the randomness of P. Since we consider regression forest, $H(\cdot)$ can be estimated by standard deviation of the multi-variable Gaussian distribution. Starting from the root node, the dataset is iteratively split into two subsets and passed down to the node's children until one of the following stopping criteria is true:

1. the maximum depth of tree is reached;
2. the number of samples that reached the node is insufficient to split; or,
3. the information gain of the best split is too small.

Then, the leaf stores the distribution of the parameters of **Y** that typically employ a normal distribution with its mean and standard deviation. As a result of learning, each branch of the tree stores the parameters of the splitting function with respect to the input **X** while each leaf stores a distribution of the output **Y**. To enfoce the independence of the trees in the forest, each random tree selects a random subset of elements in **X** or a random subset of the learning dataset.

During *testing*, a new sample of **X** traverses the tree. At each branch, it moves to the left or right child node depending on the splitting function, eventually ending up at a leaf node which contains the prediction to be associated with such sample. Finally, the results of the leaf nodes at different trees are aggregated in order to robustly obtain the final prediction.

3.2. Tracker

In this work, a template is defined as a rectangular region that encloses the three keypoints at the tip of the tool which are used for the pose estimation. The region is axis-aligned to the shaft and the tool tip is on the upper third of the bounding box. In this way, the rectangular region is large enough such that all the keypoints are visible for the pose estimation. In practice, the tracker is initialized by enclosing a bounding box around the tool on the first frame of a given video sequence. The objective then is to propagate the transformation parameters from one frame to the next in order to keep track of the region of interest.

Mathematically, a template is described by the RGB intensity values at n_s sample points, written as $\{\mathbf{x}_s\}_{s=1}^{n_s}$, which are 2D points that are randomly selected within the rectangular region. Thus, given a sequence of images $\{I_t\}_{t=0}^{n_t}$, the tracker determines the transformation $\mathbf{T}_t$ of the sample points and, in effect, locates the rectangular region such that the tool tip is enclosed in the bounding box. Based on this, the tracker uses the random forest to learn the relation of the RGB intensity vector at the sample point locations of the previous frame $\mathbf{X} = [I_t(\mathbf{T}_{t-1} \cdot \mathbf{x}_s)]_{s=1}^{n_s}$ and the corresponding translation vector $\mathbf{Y} = \delta\boldsymbol{\mu}$ that aligns the bounding box at the position of the tool tip through

$$\mathbf{T}_t = \mathbf{T}_{t-1}\mathbf{T}(\delta\boldsymbol{\mu}). \tag{2}$$

The algorithm is inspired by the work of Tan and Ilic (2014). Similar to them, our tracker runs at less than 2 ms per frame with a single CPU core. The fast tracking time with a very small computational cost is the primary advantage of this tracker.

In general, the cost function of most template tracking algorithms (e.g. Baker and Matthews, 2004; Jurie and Dhome, 2002; Holzer et al., 2012; Tan and Ilic, 2014) follow the pixel-wise difference

$$E(\mathbf{x}_s) = \|I_t(\mathbf{T}_{t-1} \cdot \mathbf{x}_s) - I_{t_0}(\mathbf{x}_s)\| \,\forall\, \mathbf{x}_s, \tag{3}$$

derived from image registration where $I_{t_0}(\mathbf{x}_s)$ is the given template. Based on Tan and Ilic (2014), they converted Eq. 3 as the feature vector $\mathbf{X} = [I_t(\mathbf{T}_{t-1} \cdot \mathbf{x}_s) - I_{t_0}(\mathbf{x}_s)]_{s=1}^{n_s}$ of a random forest. In tracking, each node of the tree thresholds one element of **X** to determine whether to traverse to the left or right child until a leaf is reached. Due to the given $I_{t_0}(\mathbf{x}_s)$, both the cost function and feature vector illustrate the limitation of these methods to track a single template. However, we observed that, when using pixel-wise splitting, $I_{t_0}(\mathbf{x}_s)$ is always constant for each element of the feature vector. Thus, we can incorporate $I_{t_0}(\mathbf{x}_s)$ as part of the threshold and simplify the feature vector as $\mathbf{X} = [I_t(\mathbf{T}_{t-1} \cdot \mathbf{x}_s)]_{s=1}^{n_s}$. In this formulation, the tracker learns and tracks based on the intensity values instead of the intensity difference. As a consequence, we can

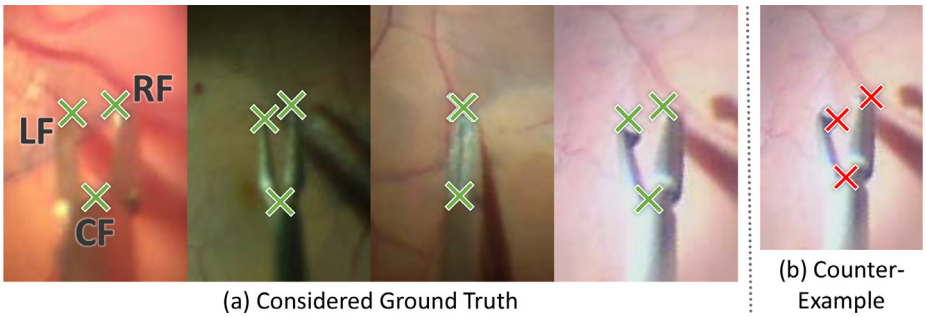


Fig. 3. Considered ground truth in instrument dataset: Due to the variation of instrument shapes, the ground truth has to be set in more detail in order to be well-defined. (a) shows examples for the considered ground truth. The tool tips (*LF* and *RF*) are set as the points on the tip which are closest to the microscope and would touch in case of a closed forceps. The center joint (*CF*) is the point connecting the two parts of the forceps. In (b), the annotation is not selected as the top visible part the tool tip points but on the part which is closer to the retina.

take a step further to alleviate the limitation of learning only one template and learn based on the intensity values of multiple templates. Therefore, in contrast to other template tracking algorithms where they learn and track an individual template, we generalize our algorithm to utilize multiple correlated templates to handle the visual changes due to the articulated deformation of the tool and different instrument structures, and to be robust against various environmental factors such as illumination changes and photometric distortions that are common in such working conditions as shown in Fig. 3.

Learning. Considering that the algorithm is a temporal tracker, it predicts the update parameters that refines the location of the tool from the previous frame to the current frame through Eq. 2. The forest then learns to predict the movement from an erroneous position of the tool to its ground truth. To create the learning dataset of the forest, we enforce this movement by randomly transforming the template from its ground truth location. Given n_i templates to learn and the individual ground truth transformation $\mathbf{T}_i$ of the ith template, we impose n_r random transformations on the ith template by transforming the sample points by $\mathbf{T}_i\mathbf{T}_r^{-1}$ for all $\{\mathbf{T}_r\}_{r=1}^{n_r}$ where $\mathbf{T}_r = \mathbf{T}(\delta\boldsymbol{\mu}_\mathbf{r})$. The objective of introducing $\mathbf{T}_r^{-1}$ is to mimic the transformation from the previous frame such that a transformation of $\mathbf{T}_r$ updates the position of the template from an erroneous location $\mathbf{T}_i\mathbf{T}_r^{-1}$ to its ground truth location as $(\mathbf{T}_i\mathbf{T}_r^{-1})\mathbf{T}_r = \mathbf{T}_i$.

As a consequence, each random transformation generates a combination of samples and labels written as $(\mathbf{X}_r, \mathbf{Y}_r) = ([I_i(\mathbf{T}_i\mathbf{T}_r^{-1}\cdot\mathbf{x}_s)]_{s=1}^{n_s}, \delta\boldsymbol{\mu}_r)$, which are accumulated as the set $P = \{(\mathbf{X}_d, \mathbf{Y}_d)\}_{d=1}^{n_i\cdot n_r}$ that are used to learn one forest per transformation parameter. Our goal in learning is to divide the learning dataset P, as the tree gets deeper, into subsets with similar transformation parameters such that, in tracking, the subset is located and uses the mean of the parameters as the prediction.

When learning a tree, each node splits the the learning dataset into two subsets which are inherited by the left and right child. Given the subset of the learning dataset P_n that arrived on the node n, an index β of the vector $\mathbf{X}_r$ is selected to threshold the values across the dataset. In order to find the best split, multiple indices and thresholds are tested and the selection of the best pair is measured based on the information gain in Eq. 1 where

$$H(P) = \frac{1}{|P|}\sqrt{\sum_{\mathbf{Y}_d\in P}\|\mathbf{Y}_d - \delta\bar{\boldsymbol{\mu}}(P)\|^2} \tag{4}$$

is the standard deviation of the transformation parameter and

$$\delta\bar{\boldsymbol{\mu}}(P) = \frac{1}{|P|}\sum_{\mathbf{Y}_d\in P}\mathbf{Y}_d \tag{5}$$

is the mean vector of all parameters in P. This implies that the result of the split is two subsets with a more homogeneous transformation parameter.

With the splitting of P enforced in each node, the tree continuously grows until one of the stopping criteria in Section 3.1 is satisfied. Consequently, the node is considered as a leaf and stores the mean and standard deviation of the parameters based on the subset of the learning dataset that arrives on the node, which is similar to Eqs. 4 and 5. Here, the mean from Eq. 5 is the predicted transformation parameter in tracking while the standard deviation from Eq. 4 acts as a weight that measures the homogeneity of the parameters within the subset.

Tracking. When evaluating the trees in tracking, $\mathbf{X}$ is computed using the transformation from the previous frame. Through the splitting function at the nodes of a tree, $\mathbf{X}$ manoeuvres from the root node to a leaf where the mean and standard deviation of the predicted parameter is stored. Considering the possibility that, when learning the tree, some subsets of the learning dataset does not converge to a homogeneous transformation parameters, only the 15% of the best predictions multiple trees of the forest with the lowest standard deviation are aggregated in finding the parameters. Using Eq. 5, the best predictions are aggregated by computing the average parameter $\delta\bar{\boldsymbol{\mu}}$. Thereafter, the predictions constructs $\mathbf{T}(\delta\bar{\boldsymbol{\mu}})$ and updates the transformation with Eq. 2 from $t-1$ to t. Notably, the forest is used iteratively to refine the previous estimate.

Since the standard deviation measures the confidence of the predictions, we deem tracking successful (i.e., the tracker converges to a confident solution) if the average standard deviation in the final iteration is less than a threshold; otherwise, we avoid updating the transformation parameters and utilize the previous location of the tool for the succeeding frames.

3.3. Pose estimation

The final task of the presented method is the localization of the instrument parts in every frame, which similarly to the tracking stage, also has to be carried out in real-time. The main idea behind our approach is to interpret the surgical tool as an articulated object and to employ parametric models similar to those successfully proposed in the field of human pose estimation (as can be seen in (Belagiannis et al., 2014). Specifically, by defining the set of joints as the left tip of the instrument (LF), the right tip of the instrument (RF) and the center joint (CF) connecting these two parts, we can integrate this methodology in our approach. Since we investigate the performance of the algorithm on different instrument shapes, we have to emphasize that the points on the tips are defined as the inner-most and top visible point of the part, which

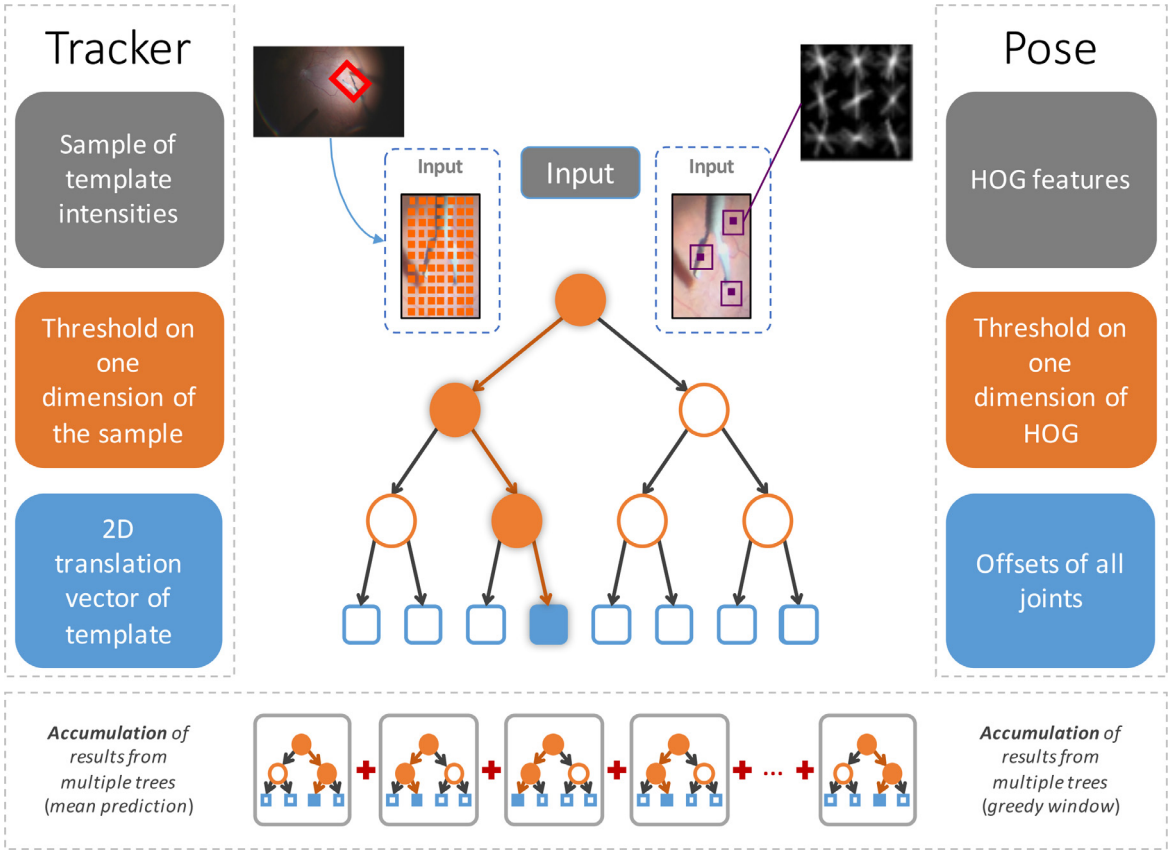


Fig. 4. Overview random forests: The input **X** for *Tracker* are the intensity values of the current frame I_t at the sampling positions x_s from previous frame I_{t-1}. The binary split function θ divides the samples by thresholding on one dimension of **X**. The output **Y** is the 2D translation of the template I_B. Finally, the results are accumulated by taking the mean of the predictions with the smallest standard deviation from the forest of a parameter. For the *Pose Estimation*, the input **X** consists of the HoG features extracted at randomly selected points within the bounding box $q \in I_B$ and the binary test θ is performed on one dimension of the HoG feature. The output **Y** is the offset of the joints of the instrument. The final estimation is aggregated by a greedy dense-window algorithm.

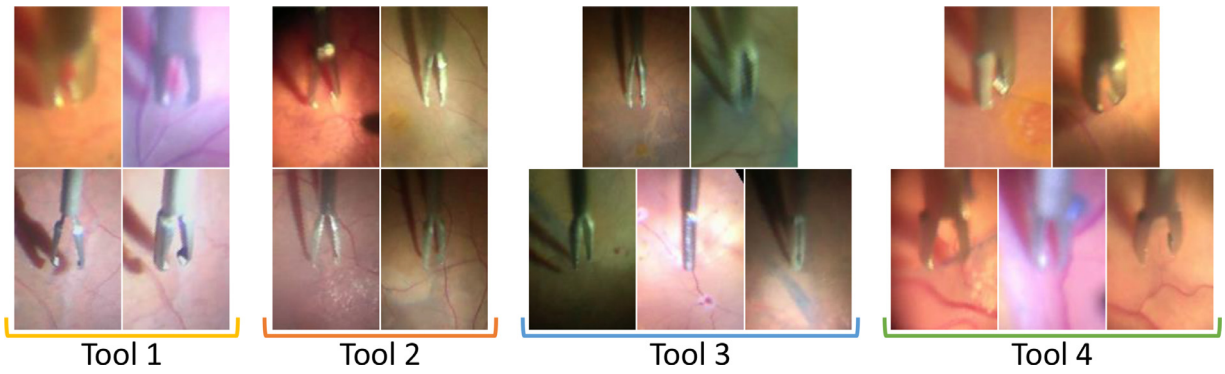


Fig. 5. Instrument dataset representative frames: Different types of instruments are present in our dataset. They show different shapes, for example **Tool 1** and **Tool 4** are bulky, the first being more rounded close to the center joint. **Tool 2** and **Tool 3** are smaller, with Tool 3 showing a extremely thin attachment to the shaft. In addition, different illuminations, reflections, blur and colouring can be observed.

tend to touch each other in case of a closed forceps (see Fig. 3), in order to have a well-defined ground truth.

The goal is now to infer the location of the instrument parts from the extracted image features. In contrast to part-based methods, the holistic approach aims at predicting the joints at one step. Instead of considering the image information of the entire frame, the tracker simplifies the problem by limiting the region of interest to a bounding box $I_B \subset I_t$, and therefore drastically reduces the computational cost. This is an important observation since in this second step of the pipeline, we make use of the computationally more expensive gradient information, which tends to be highly reliable in these kind of challenging scenarios. More precisely, we employ Histogram of Oriented Gradients (HoG) features (Dalal and Triggs, 2005), which have shown their robustness in fields such as object detection (Felzenszwalb et al., 2010), image retrieval (Eitz et al., 2011) and classification (Nilsback and Zisserman, 2008). Here, a key aspect is that the tracked template as defined in Section 3.2 yields a bounding box around the tool tip which is aligned with the direction of the tool shaft at the time of the initialization. During the tracking, only the translation parameters are updated and I_B is not necessarily aligned with the instrument shaft any more. However, the insertion point of the

Table 1
Overview of the novel dataset introduced in this work. Representatives of the Tool Types are shown in Fig. 5. Example frames of every sequence are displayed in Figs. 8, 9, 10, 11. For determination the average RGB color, all pixels in the range (0,0,0) to (10,10,10) were excluded from the computation. Significant scaling is present, if the ratio of minimum size to maximum size of instrument tip is higher than 0.5. Translation is considered as significant, if the overall translation distance of the central joint is higher than 1000 pixels.

Sequence	Resolution (pixel)	Tool type	# open-close	Average RGB color	# frames	Average (*h*, *w*) for KBB (pixel)	Lightsource in focused area	Significant Scaling	Significant Translation
1	1920 × 1080	Tool 1	2	[103, 59, 35]	200	(53,70)	✓	✓	✓
2	1920 × 1080	Tool 1	3	[74, 50, 70]	200	(83,43)	✓	✓	✓
3	1920 × 1080	Tool 1	2	[68, 57, 55]	200	(67,82)	×	×	✓
4	1920 × 1080	Tool 1	4	[84, 68, 66]	200	(100,115)	✓	×	✓
5	1920 × 1080	Tool 2	2	[91, 41, 31]	200	(98,178)	✓	✓	✓
6	1920 × 1080	Tool 2	2	[71, 55, 31]	200	(131,79)	✓	✓	✓
7	1920 × 1080	Tool 2	3	[26, 29, 37]	200	(87,33)	✓	✓	×
8	1920 × 1080	Tool 2	3	[32, 45, 61]	200	(126,69)	×	✓	✓
9	1920 × 1080	Tool 3	2	[65, 48, 34]	200	(226,74)	×	×	✓
10	1920 × 1080	Tool 3	2	[52, 53, 30]	200	(94,31)	×	×	×
11	1920 × 1080	Tool 3	2	[49, 48, 27]	200	(121,60)	✓	×	✓
12	1920 × 1080	Tool 3	1	[100, 68, 67]	200	(173,109)	✓	×	✓
13	1920 × 1080	Tool 3	3	[60, 44, 36]	200	(104,91)	✓	✓	✓
14	1920 × 1080	Tool 4	2	[133, 77, 55]	200	(104,173)	✓	✓	✓
15	1920 × 1080	Tool 4	1	[83, 50, 30]	200	(77,141)	✓	✓	✓
16	1920 × 1080	Tool 4	3	[123, 62, 46]	200	(76,98)	✓	✓	✓
17	1920 × 1080	Tool 4	2	[96, 57, 74]	200	(93,61)	✓	✓	✓
18	1920 × 1080	Tool 4	1	[115, 73, 54]	200	(93,130)	✓	×	✓

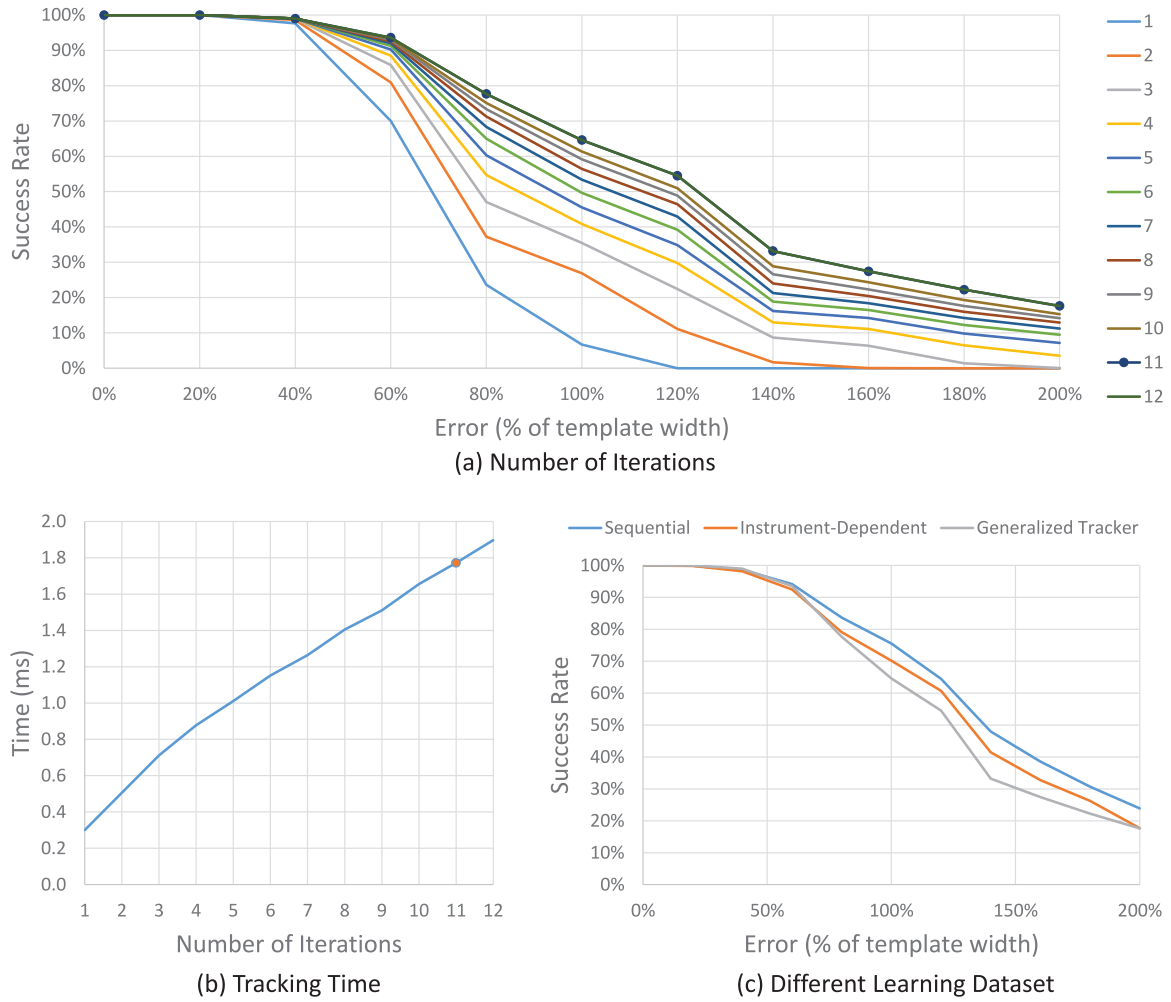


Fig. 6. Parameter evaluation for the tracker: The experiment was evaluated on every sequence of the Instrument Dataset (IDS). The depicted results show the success rate of the tracker and the timings associated with them. With 100 trees, we evaluate the number of iterations required to achieve convergence in (a) with its corresponding tracking time in (b). In addition, we compare the success rate when learning an increasing number of templates (i.e., 100 for sequential, 400-500 for instrument-dependent and 1800 for the generalized tracker).

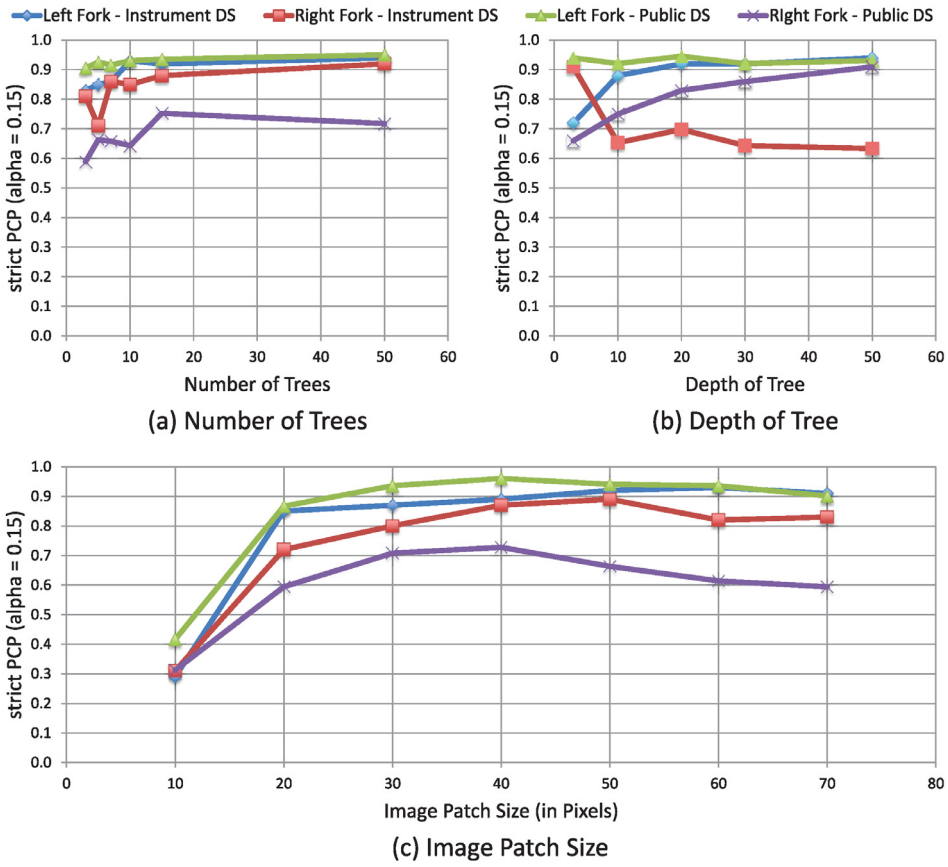


Fig. 7. Parameter evaluation for pose estimation: The experiment was evaluated on one sequence each of the Instrument Dataset (IDS) and Public Dataset (PDS). The depicted results are the strict PCP scores for the alpha value of $\alpha = 0.15$. It should be considered that in human pose estimation, a common choice is $\alpha = 0.5$. Therefore, the low α value in our case constrains that only very precise predictions are accepted.

instrument to the eye is fixed by trocars during a procedure and consequently the orientation of the tool shaft remains in a limited range. Therefore, we consider the set of templates as defined in Section 3.2 together with the ground truth annotation of the joints as base learning dataset and augment the data by applying n random similarity transformations with parameters in the range of $\pm$ 0.3 for the scale, $\pm$ 30 pixel for the translation in x and y direction and $\pm$ 30 degrees for the rotation from the ground truth homography. All images are rescaled to a fixed pixel size. This yields an extended dataset which improves the robustness of the pose estimation and tackles the problem that HoG features are not rotation invariant.

Within the bounding box I_B, the HoG features are computed on image patches around randomly selected points and are employed as an input $\mathbf{X}$ for the trees. The binary split function θ divides the input sample regarding a threshold on one dimension of the HoG feature.

The function $H(\cdot)$ is based on the Sum-of-Squared-Distances (SSD)

$$H(P) = \sum_{i \in I} \sum_{j} \| o_{i,j} - \mu_j \|_2^2, \tag{6}$$

where I denotes the image patch, the 2D vector $o_{i,j}$ contains the offset of the joint $j \in J$ from the image patch center and μ_j is the mean for each joint offset. The leaves store the corresponding offsets $o_j = \mathbf{Y} \subset \mathbb{R}^2$ of all instrument joints $j \in J = \{LF, RF, CF\} \subset \mathbb{R}^2$. In order to find the most probable outcome, the votes of the separate trees are accumulated by a greedy dense-window algorithm, similar to the work of Belagiannis et al. (2014). For this purpose, the 2D predictions for every joint $j \in J$ are discretized on a fixed grid, whereas the grid cells contain the number of votes that lie within it. To aggregate its votes, an integral matrix is created for every cell and all the cells form an integral image. Then, the final estimation corresponds to the region with the maximum number of points, which is found by sliding a window over the integral image.

In this way, a direct mapping between the extracted HoG features and the location of the instrument joints is modelled. Due to the characteristics of the random forest, this relation can also be inferred for unseen instrument poses or varying lighting conditions. An overview of the random forests is visualized in Fig. 4.

4. Material

4.1. Description of the datasets.

The experimental validation of the proposed algorithm is carried out on two different RM datasets: a new dataset, in the following called *Instrument Dataset* and the dataset published by Sznitman et al. (2012), in the following referred to as *Public Dataset*. For comparison, the performance of the algorithm was also evaluated on a published laparoscopic instrument sequence.

Instrument dataset: This consists of 18 sequences of *in-vivo* retinal surgery and is an extended version of the *appearance* dataset presented in the work of Rieke et al. (2015), which only contained 4 of the 18 sequences. The images are acquired by a Carl-Zeiss Lumera 700® operating microscope with a resolution of 1920 $\times$ 1080 pixels at 25 fps progressive scans with 24-bit RGB color format. For each sequence, we selected 200 subsequent frames in which the instrument is always visible and at least one

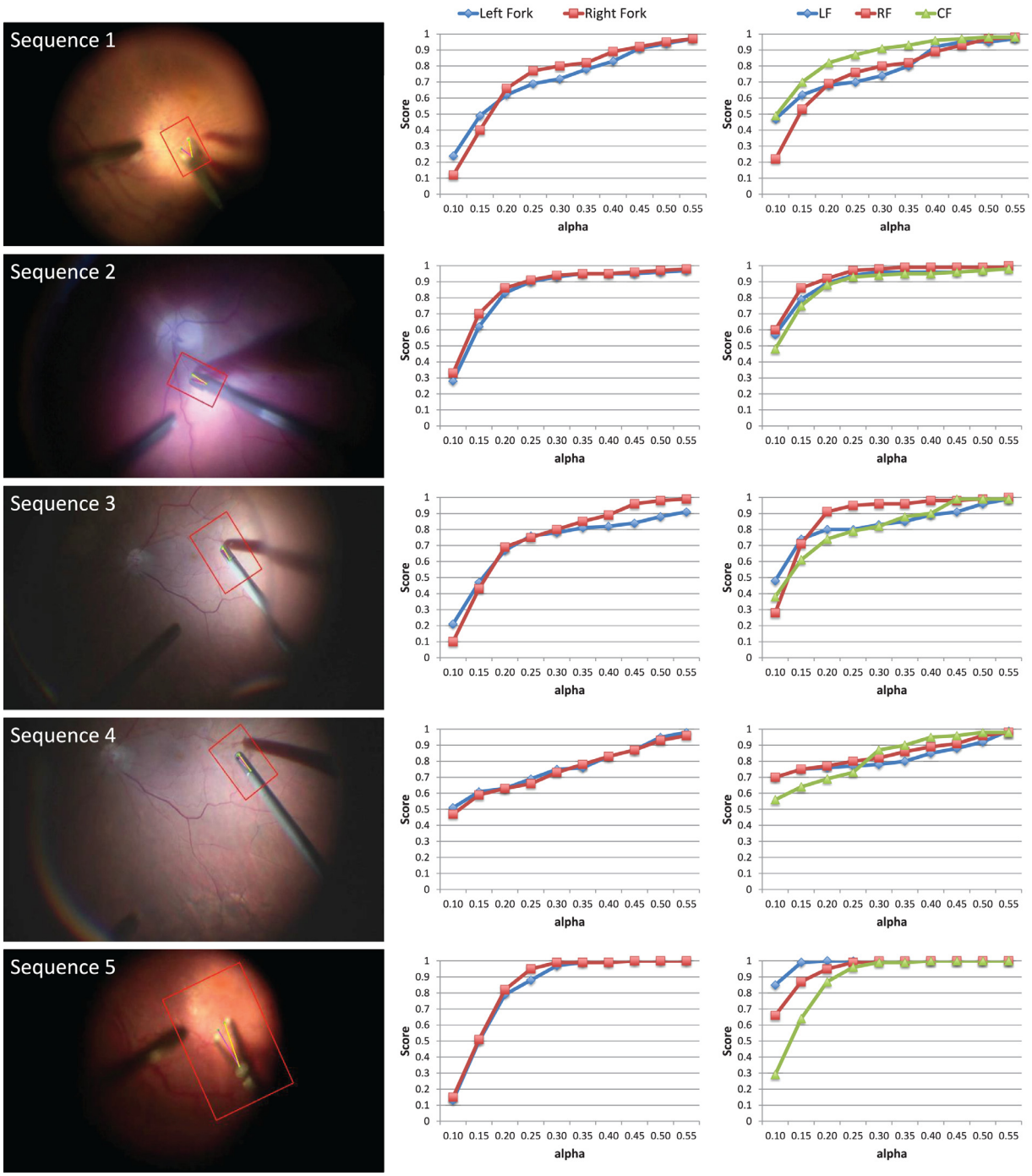


Fig. 8. Part I of the sequential evaluation of instrument dataset: Results for sequences 1 to 5. Left column: Example of the sequence. Middle column: strict PCP score for the left and the right fork. Right column: *KBB* score for evaluating the prediction of the keypoints.

movement of opening and closing is present. Each frame was annotated manually, following the definition of the ground truth given in Section 3.3. In comparison to the sequences in Rieke et al. (2015), the dataset was considerably extended and allows us to perform instrument dependent experiments. Furthermore, additional lightning variations and microscope zoom factors are present, increasing the complexity of learning. In total, four different types of instrument can be observed as depicted in Fig. 5. Therefore, depending on the type of tool present in the sequences, we divided the dataset in four smaller subsets, containing respectively 4, 4, 5 and 5 videos. An overview of characteristics of the sequences of the novel dataset can be found in Table 1.

Public dataset [1]**:** This is a fully annotated dataset of three different sequences of *in-vivo* vitreoretinal surgeries. It comprises of 1171 images with a resolution of 640 × 480 pixels with respectively 402, 222 and 547 frames for the first, second and third sequence. The main challenge of this dataset is variations of lighting as well as the presence of noise and shadows. Notably, the same instrument is utilized in all sequences. The key component of this dataset is the dominant blue and green colouring of the sequences, on which the dependence of an algorithm regarding its color reliance can be evaluated.

[1] https://sites.google.com/site/sznitr/code-and-datasets

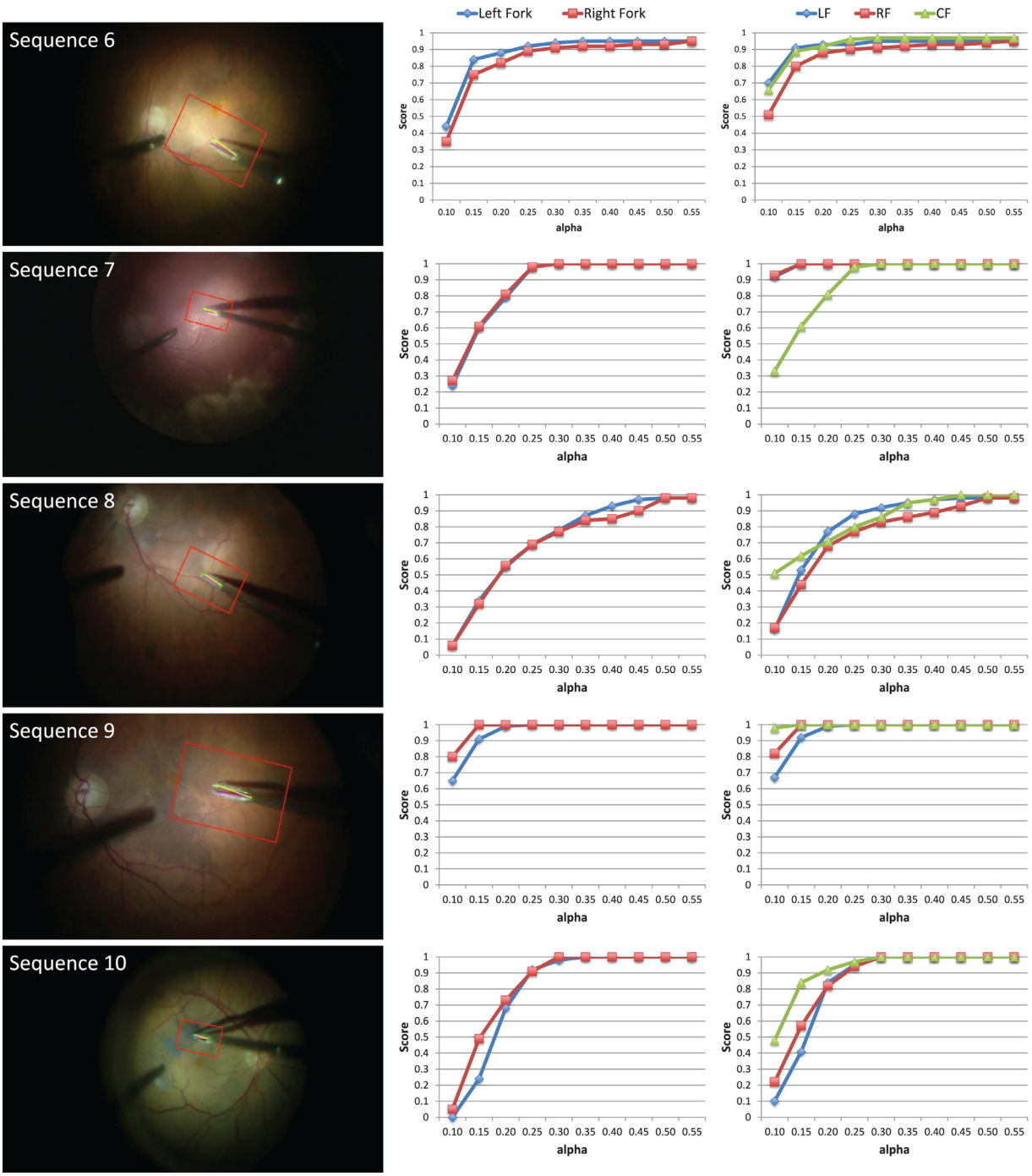


Fig. 9. Part II of the sequential evaluation of instrument dataset: Results for sequences 6 to 10. Left column: Example of the sequence. Middle column: strict PCP score for the left and the right fork. Right column: *KBB* score for evaluating the prediction of the keypoints.

Laparoscopic sequence[1]**:** This is an annotated and publicly available laparoscopic instrument sequence. It consists of 1000 frames and shows two surgical instruments. The location of the central joint is labelled for every visible instrument. We focus on the more challenging instrument (in previous works referred to as Tool 1 Li et al., 2014) and extend the provided labels by manually annotating the location of the tool tips. For pose estimation, the main difficulties are the partial occlusions when the instrument enters the tissue and the presence of smoke. Furthermore, the sequence is recorded with large variations regarding the distance between the instrument and the camera.

4.2. Description of the metrics

The performance of our method was evaluated by means of four different metrics which are presented in this section, including standard metrics and a newly proposed metric addressing the variation of the scales of the instruments and image resolutions in the Instrument Dataset.

Strict percentage of correct pose (strict PCP): This addresses the quality of the prediction for a part of an articulated object and is a standard metric in human pose estimation (Pickering et al., 2009). A prediction for a part connected by two joints $j_1, j_2 \in \mathbb{R}^2$

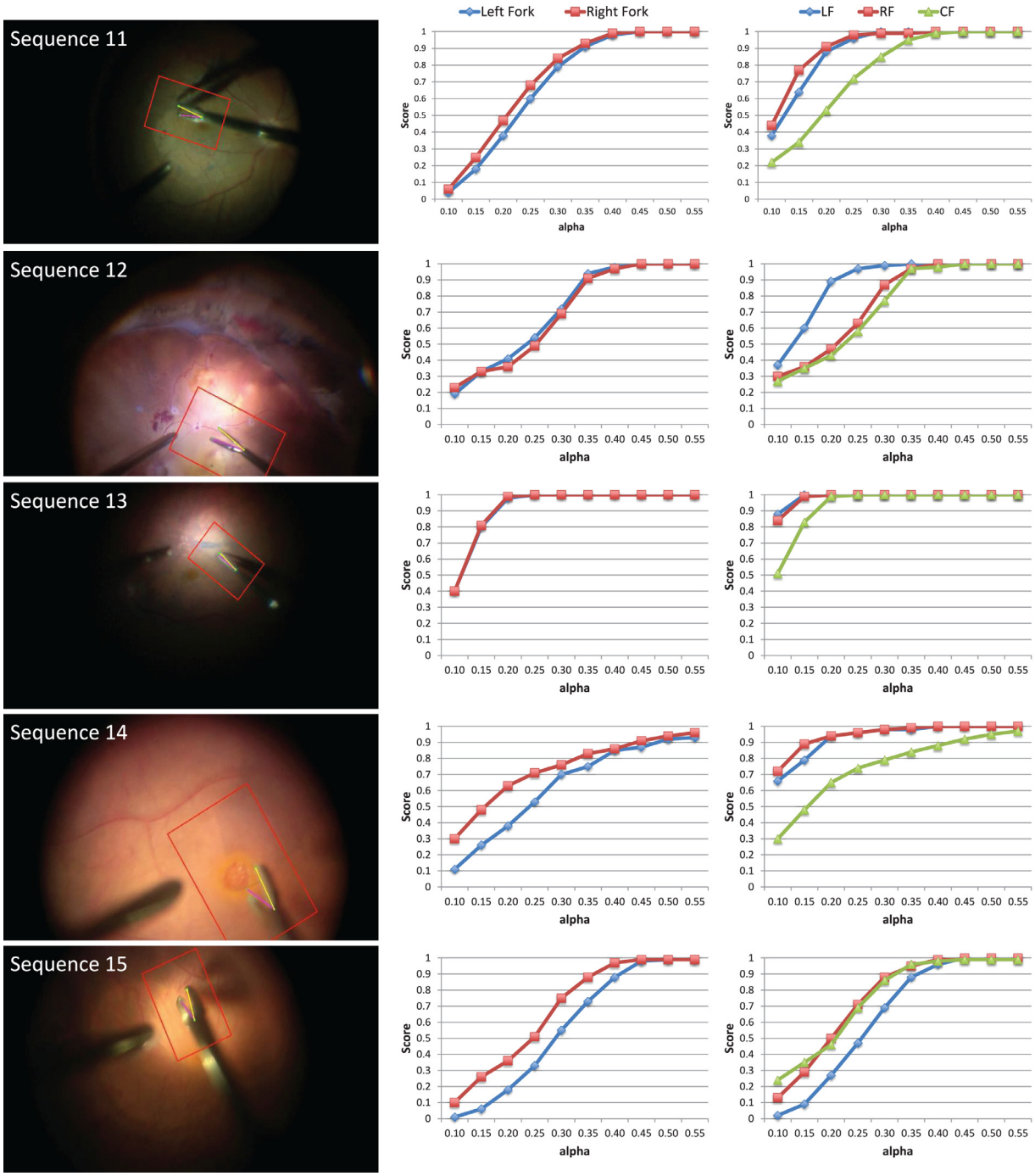


Fig. 10. Part III of the sequential evaluation of instrument dataset: Results for sequences 11 to 15. Left column: Example of the sequence. Middle column: strict PCP score for the left and the right fork. Right column: *KBB* score for evaluating the prediction of the keypoints.

is evaluated as correct only if both the euclidean distances of the predicted joints j_1, j_2 to its ground truths $\hat{j}_1$, $\hat{j}_2$ are lower than a threshold as a function of the ratio $\alpha \in \mathbb{R}$ times the ground truth length of the part, e.g. both of the following equations have to be fulfilled:

$$\|j_1 - \hat{j}_1\| < \alpha \cdot \|\hat{j}_1 - \hat{j}_2\|$$
$$\|j_2 - \hat{j}_2\| < \alpha \cdot \|\hat{j}_1 - \hat{j}_2\|.$$

For human pose estimation, the threshold value is usually set to $\alpha = 0.5$ (compare Pickering et al., 2009).

Keypoint Threshold (KT): This was employed by Sznitman et al. (2012) and addresses the quality of the keypoint predictions as a pixel-wise measure. Estimated joint locations $j \in \mathbb{R}^2$ are evaluated as correct if the euclidean distance to the ground truth annotation $\hat{j} \in \mathbb{R}^2$ is lower than a fixed pixel threshold $T \in \mathbb{R}$:

$$\|j - \hat{j}\| < T. \tag{7}$$

Therefore, it yields a separate evaluation for every keypoint $j \in J$.

Keypoint threshold bounding box (KBB): The *KT* metric indirectly assumes that the frames have the same resolution and show the same type of instrument. However, the selection of a reasonable threshold is difficult for different zoom factors and instru-

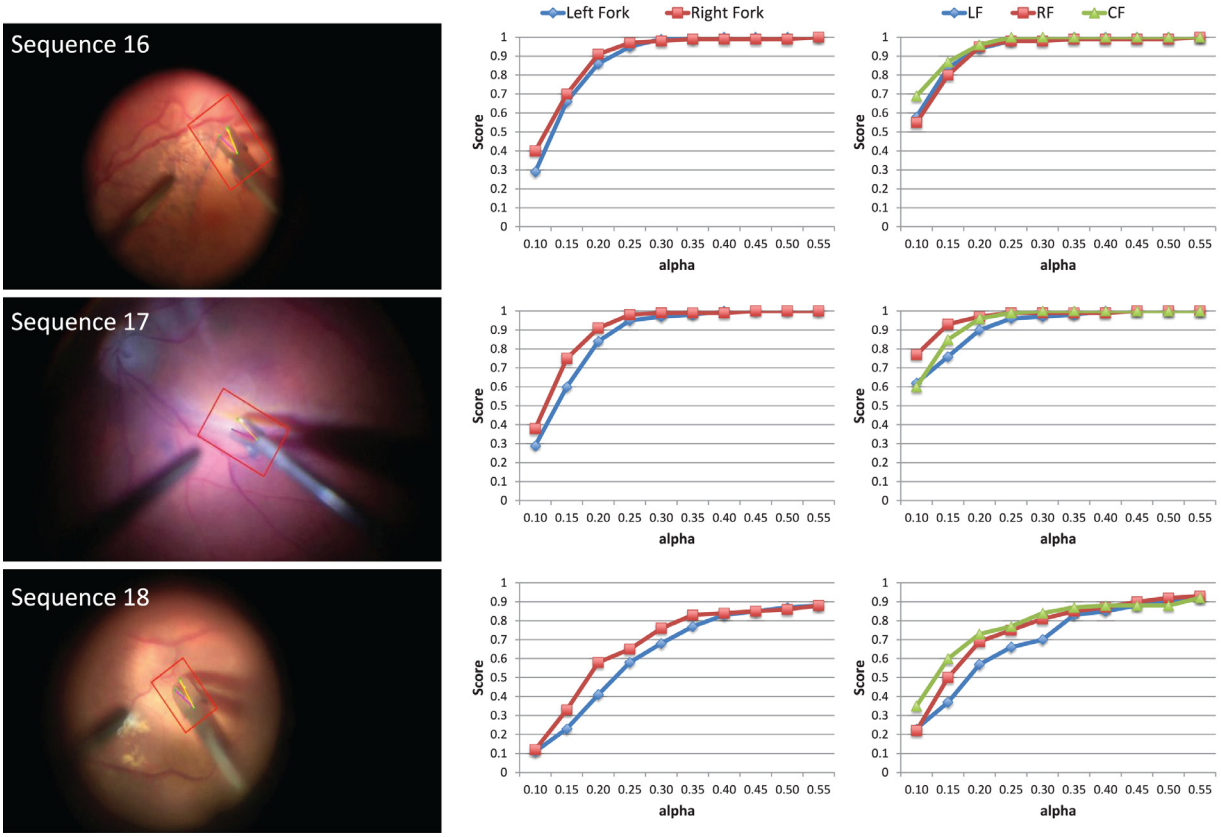


Fig. 11. Part IV of the sequential evaluation of instrument dataset: Results for sequences 16 to 18. Left column: Example of the sequence. Middle column: strict PCP score for the left and the right fork. Right column: *KBB* score for evaluating the prediction of the keypoints.

ments, leading to the problem that sequences are not directly comparable. Inspired by the metric introduced by Yang and Ramanan (2013) in the field of human pose estimation, we propose a novel metric for retinal microsurgery which addresses this problem. Instead of using a fixed pixel threshold, the accepted distance depends on the size of the instrument tip. In this way, a higher resolution of sequences and a change in the distance of the instrument from the retina does not automatically lead to a higher error for the keypoint evaluation. For this, we consider a tightly cropped, axis-aligned bounding box which contains all ground truth joints of the instrument in the respective frame. We define a joint j to be located correctly if

$$\|j - \hat{j}\| < \alpha \cdot \max(h, w), \tag{8}$$

where $\hat{j}$ is the ground truth annotation of the joint, w and h are the width and height of the bounding box around the instrument given by the ground truth, and $\alpha \in \mathbb{R}$. It should be noted that this metric is only computable if the ground truth of all joints is given. However, the evaluation of a keypoint is pose-independent and also applicable if only the joint point *CF* is estimated.

Success rate of the tracker: The tracker is evaluated by inducing random translation to the template to simulate the displacement of the bounding box from the previous frame to the current frame. Apart from the standard frame-to-frame tracking that is used to find the bounding box for the pose estimation, we also introduce this synthetic evaluation to numerically determine the range of translation error, which the tracker can handle. In this case, the maximum translation error is parameterized with respect to the percentage of the template's width. Here, a successfully tracked template is determined by asserting that all three joints must be within the bounding box, which is defined to be relatively tight around the tool tip. After applying the synthetic evaluation across multiple images, it follows that the success rate is defined as the percentage of successfully tracked templates over the total number of tests.

5. Experiments and results

In this section, we present the results of the experiments that are performed on the three different dataset presented in Section 4.1 and evaluated in terms of the metrics described in Section 4.2. First, the influence of the parameters for both the tracker and pose estimation is investigated (Section 5.1). We gradually evaluate the generalizability of our algorithm to unseen conditions in Section 5.2. The performance of the proposed method is compared to state-of-the-art methods on RM sequences in Section 5.3 and on a laparoscopic sequence in Section 5.4. The method is implemented in C++ and runs at 30 fps on an off-the-shelf computer.

5.1. Parameters experiments

For both the tracker and the pose estimation, several parameters have to be set during training. For this purpose, we evaluated the performance of the two algorithms independently as described in the following.

Parameter for tracker: Considering that the tracker is an iterative method, we evaluate its performance with respect to the number of iterations required to achieve convergence. After using the standard parameters of 100 trees with a maximum depth of 20, Fig. 6(a) illustrates the convergence rate of the tracker with respect to the success rate as the average translation error increases. Here, we show that the tracker performs equally well between 11

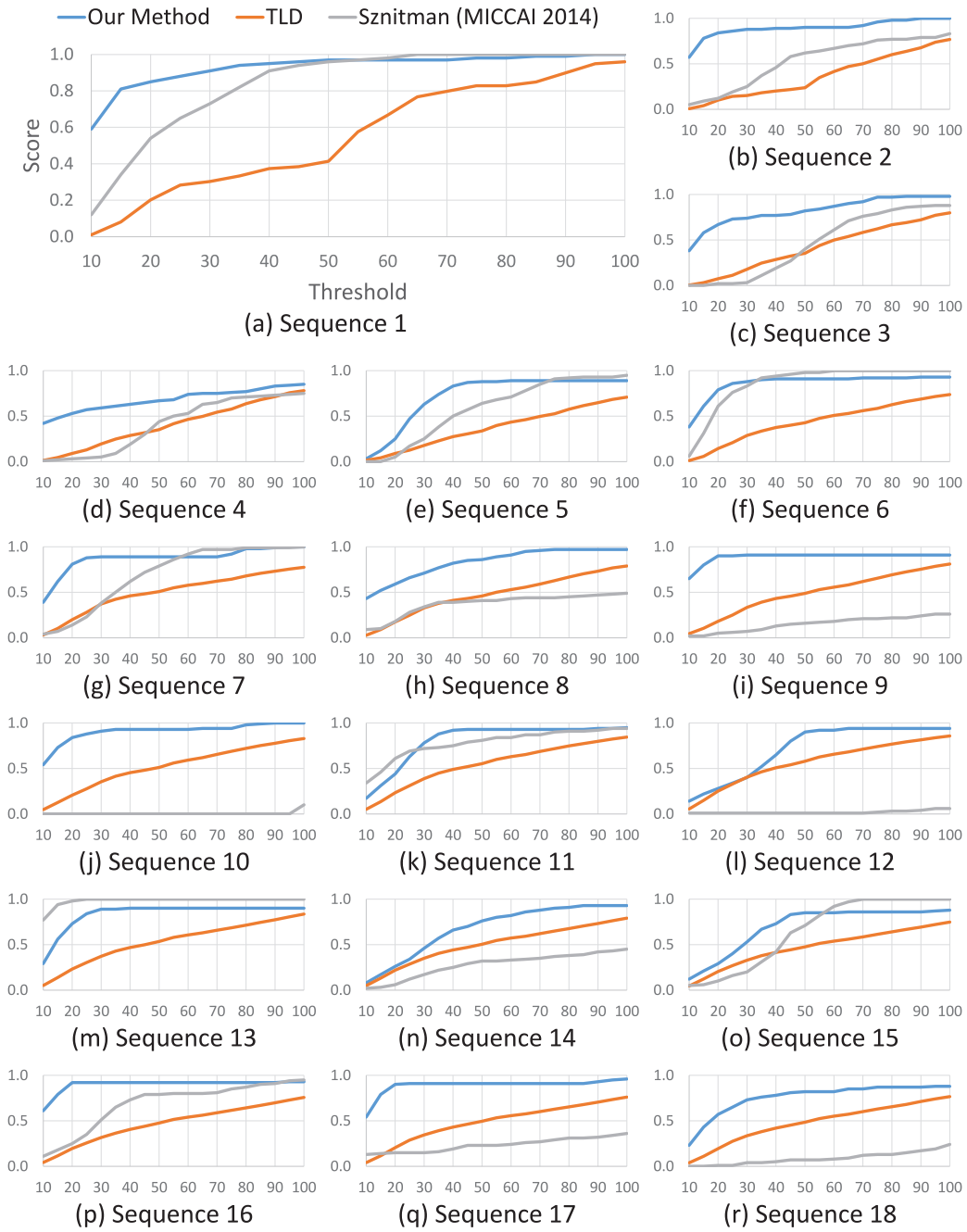


Fig. 12. Comparison sequentual evaluation of instrument dataset to TLD and Sznitman et al. (2014): The results of our method on the sequential evaluation of the instrument dataset is compared to the performance the TLD tracker from Kalal et al. (2012) and the tool tracker of Sznitman et al. (2014). The graphs show the scores for the estimation of the central joint (CF) by means of the pixel threshold metric *KT*.

and 12 iterations. With 11 iterations, the tracker runs at approximately 1.8 ms with one CPU core, see Fig. 6(b). In addition, Fig. 6(c) shows the performance of the tracker as the number of templates increases in learning the random forest. Notably, there is no significant decline in performance as the number of learned templates increases from 100 in the sequential evaluation to 400-500 in the instrument-dependent evaluation to 1800 in the generalized evaluation.

Parameters for pose estimation: We evaluate the performance of the pose estimation on one sequence of the Public Dataset and one sequence of the Instrument Dataset by varying the respective parameters. Based on the results depicted in the Fig. 7, we decided to use 15 trees with a maximum depth of 50 for the pose estimation in the following experiments. The depth of the trees are considerably high due to the high variation in terms of lighting conditions, appearance and motion of the surgical instrument. A patch size resolution of 50 pixels per dimension has proven to yield good results. For all the following experiments, a HoG features bin size of 9 is used and the resolution of the dense-window grid is set to 100×100 pixels.

5.2. Evaluations on the instrument dataset

With the introduction of this new dataset, we have the possibility of performing more detailed experiments regarding the ability of the algorithm to generalize for unseen conditions.

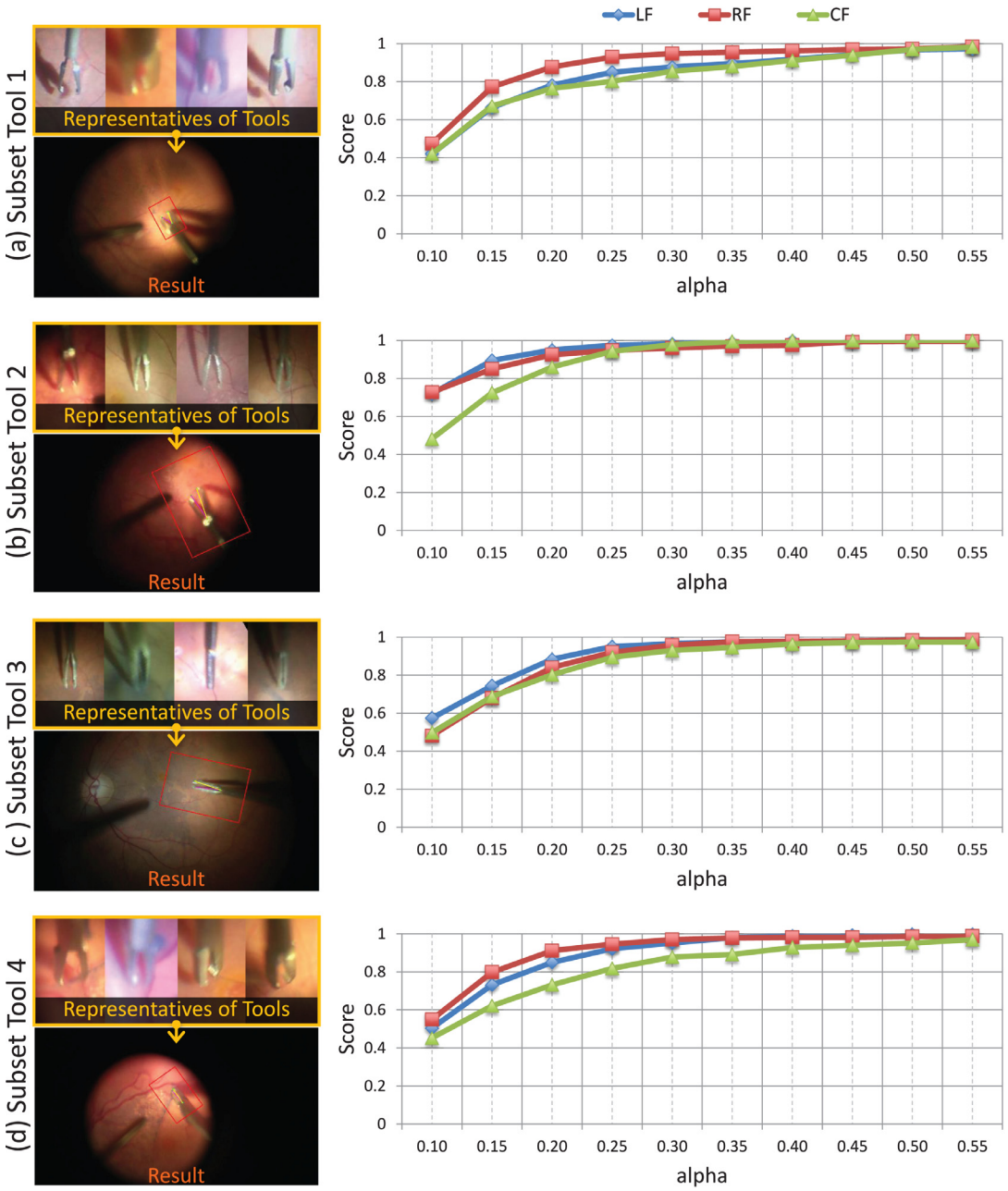


Fig. 13. Results for instrument dependent evaluation of instrument dataset: First stage of generalization. The experiment is performed separately for every subset of the Instrument Dataset by training on all the first halves of the respective sequences and evaluating on the remaining ones of the subset. The score in the right column shows the *KBB* score for all keypoints.

First, we perform a *sequence-wise* experiment on the dataset. The forests are trained on the first 100 frames of a sequence and evaluated on the remaining ones. As depicted in Figs. 8 to 11, the algorithm can reliably predict the joint positions for various blur levels and illumination changes reaching over 86% score for a strict PCP with $\alpha = 0.5$ in every sequence. The estimation of the joint positions seems to be more challenging in case of bulky instrument (compare Seq. 14, 15 and 18 – Tool 4). A reason for the comparatively worse result in Seq. 15 is also the higher amount of reflection and blur.

The performance of our method is exemplarily compared to the online learning algorithm TLD (Kalal et al., 2012) and to the offline method Fast Part-Based Classification (FPBC)[2] for RM sequences introduced by Sznitman et al. (2014) by comparing the estimation for the central joint (CF) by means of *KT*. TLD stands for tracking, learning and detection, whereas the tracker follows the object of interest in subsequent frames, the detector estimates the appearance changes and corrects the tracker and the learning step calculates the errors of the detector and updates it. The authors claim that TDL is successful for challenging videos and can handle frequent tracking failures. We initialized the bounding box around the central point using the ground truth annotation. As depicted in Fig. 12, our algorithm outperforms the baseline online tracker in every sequence.

The Fast Part-Based Classification (FPBC) algorithm represents an offline state-of-the-art tool tracking method for medical applications, which consists of the following steps: a multiclass classifier (Gradient Boosting) accelerated by an early-stopping scheme

[2] https://sites.google.com/site/sznitr/code-and-datasets

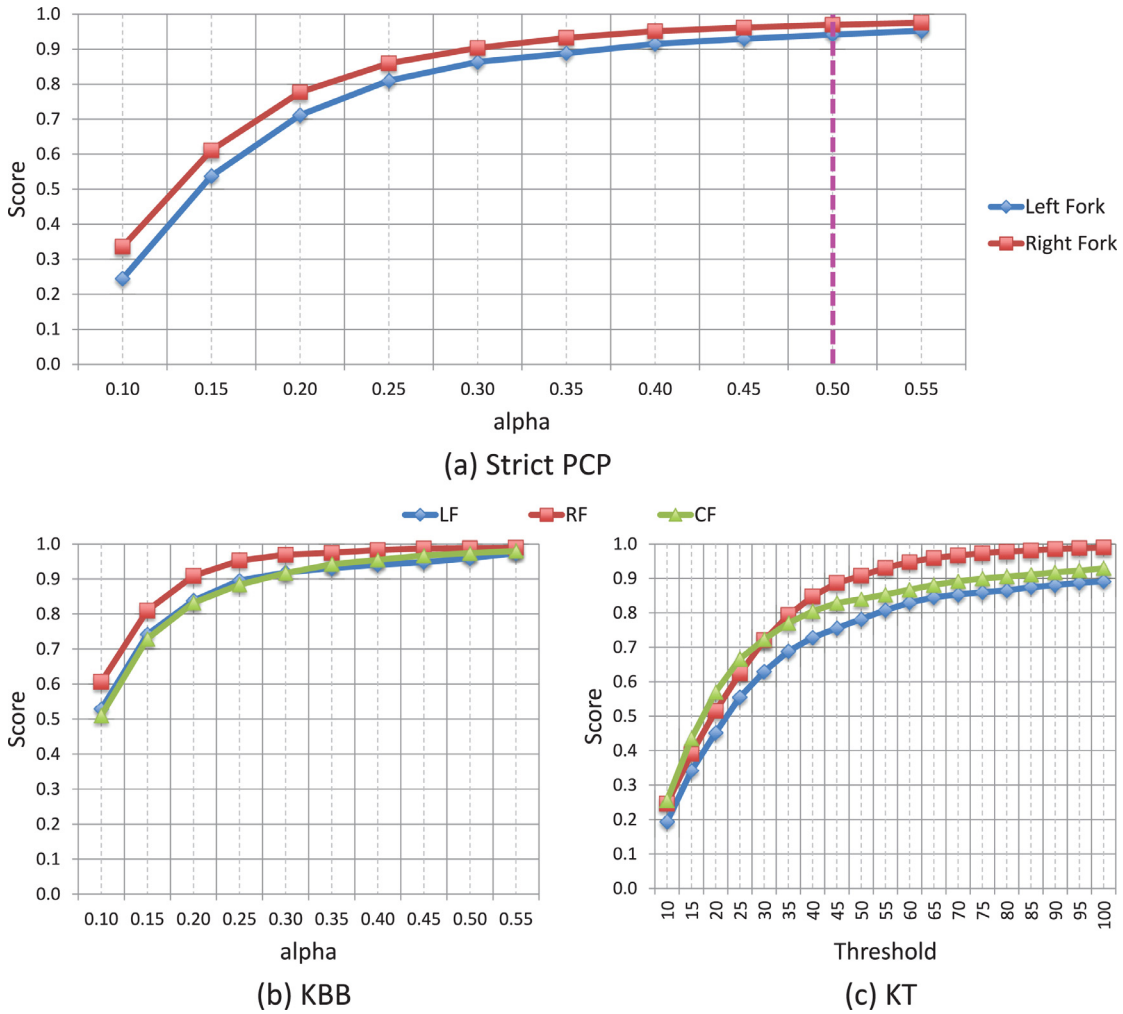


Fig. 14. Result for complete evaluation of the instrument dataset: The method was trainined on all the first 100 frames of the Instrument Dataset and evaluated on the second 100 frames. In (a), the strict PCP score for the instrument parts is visualized for different alpha values. (b) and (c) show the keypoint evaluation, whereas the former is the tool size dependent evaluation and the latter is the threshold metric.

(EDE) assigns each pixel to a class. Afterwards, a response map is generated and RANSAC is considered to obtain inliers. Finally, a weighted averaging estimates the pose of the instrument. For sake of completeness, it is important to mention that the following routine were implemented by ourselves: the patch extraction, where the original annotation were used as center of the $r \times r$ patch. For the background class, an algorithm was implemented to randomly select patches, which did not include the tool but the retina or the black background. For the Instrument Dataset, we downsampled the original images by a factor of 3 to increase evaluation speed (final image size 640 $\times$ 360) and considered four classes (background, insertion point, tool center and the tool shaft), whereas the tool center was defined as the middle point between insertion point and tool shaft. Patches were selected of size 48 $\times$ 48 pixel in order to include the instrument in all possible zoom factors of the different videos. We used the first 75 frames for training, the following 25 for the EDE early stopping criteria and finally the last 100 to test the procedure. The tree depth was set to 2, number of boosting iterations $T = 200$, RANSAC with 500 iterations, number of stopping criteria evaluation to $\delta = 10$ and the entropy threshold is set to $\gamma = 10^{-3}$. For details about the parameters, we refer the reader to the original paper by Sznitman et al. (2014). A graphical comparison can be seen in Fig. 12. It is noticeable that overall, our algorithm shows more stable performance results than FPBC. In the Seq. 10 and 12, the algorithm FPBC is confused by the presence of various vascular structures and high amount of black background, whereas the proposed method still produces reliable results.

In the next step, an *instrument dependent* experiment was performed on each subset of the Instrument Dataset. Within each subset, the shape of the instrument is similar. This allows us to investigate whether the method can generalize regarding changes in illumination and background. For this purpose, we include the first 100 frames of all the sequences of a subset in the training dataset and test on the remaining ones. Due to the differences in tool tip resolution, we now employ the newly introduced metric *KBB* for the performance evaluation. The results are summarized in Fig. 13. As already indicated by the sequential experiment, the localization is more challenging for the bulky tools (i.e., Tool 1 and Tool 4). However, the extension of the dataset to more sequences seems to increase the capture range and performance of the algorithm.

The next level is the generalization for various tool shapes. In the *complete* experiment, all the first halves of the 18 sequences are included in the training set. In this way, we can evaluate whether the algorithm can generalize not only for background, illumination changes and blurriness levels, but also for instrument shapes. Due to the more challenging scenario, we used 30 trees for the pose estimation. The results are visualized in Fig. 14. Although this experiment includes a high complexity for the learning algorithms, we reach a strict PCP score of 93.7% for the left instrument part and 95.54% for the right instrument part with respect

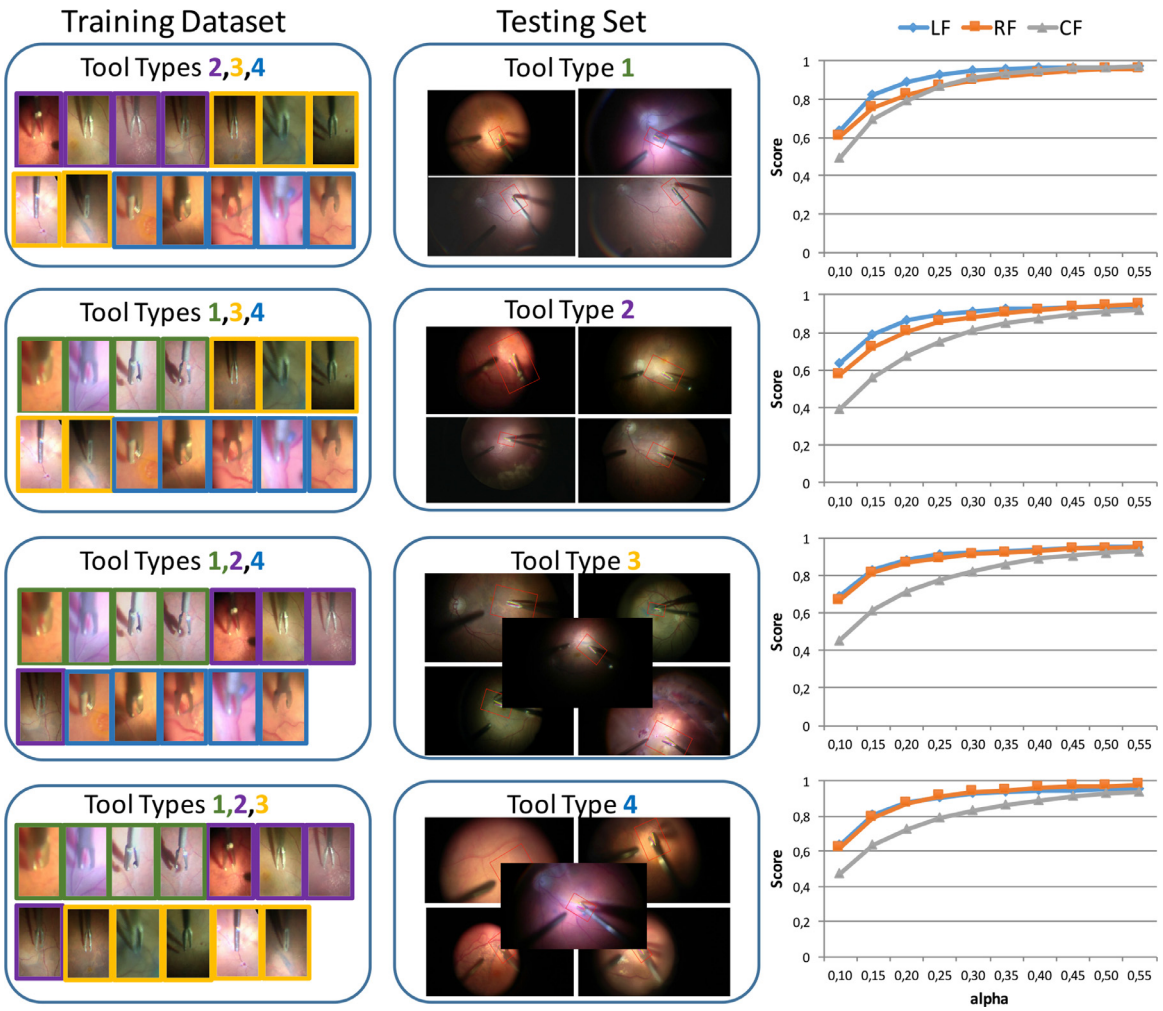


Fig. 15. Results for the cross validation of instrument dataset: The trees are trained on all sequences of three tool types and evaluated on all sequences on the remaining tool type. The score in the right column shows the *KBB* score for all keypoints.

to $\alpha = 0.5$. Regarding the keypoint scores, 83.7% for the LF, 90.4% for the RF and 80.5% for the CF of the predictions are evaluated as correct by means of the metric *KBB* with $\alpha = 0.2$. The metric *KT* indicates a worse performance because pixel thresholds are directly compared across sequences although the size of the instrument tip in pixel varies significantly.

The most challenging experiment is the *leave-one-out* validation on the subsets: the forests are trained on all sequences of three tool types and are tested on all sequences of the unseen tool type. The difficulty of this setting lies in the generalization to both an unknown geometry and unseen sequences. As depicted in Fig. 15, the proposed algorithm can build on the vast dataset and achieves at least 86.7% for the LF, 80.7% for the RF and 67.8% for the CF success rate by means of the *KBB* metric with $\alpha = 0.2$ in all four cross validations. Regarding the *PCP* score with $\alpha = 0.5$, the method predicts the instrument parts correctly in at least 88.2% for the left instrument part and 89.5% for the right instrument part.

5.3. Evaluation on the public dataset

On the Public Dataset, the performance of the proposed method is compared to state-of-the-art methods including the data-driven visual tracking (DDVT) by Sznitman et al. (2012), the visual tracking (MI) by Richa et al. (2011), a gradient-based image registration (SCV) by Pickering et al. (2009) and an online-learning approach (ITOL) by Li et al. (2014). In order to be consistent, the estimation of the position of the center joint (CF) is compared and the experiments are performed analogously to other works. The *sequential* evaluation was performed by training the random forests on the first half of a sequence and test on the remaining half (Fig. 16). In the *complete* experiment, the forest were trained on all the first halves of the three sequences and tested on the remaining halves (Fig. 17). Both experiments indicate that the proposed method outperforms the state-of-the-art methods.

5.4. Evaluation on the laparoscopy sequence

Analogously to works presented on this dataset (Sznitman et al., 2012; Li et al., 2014), we used the first 500 frames of the sequence as training dataset. For comparison, the performance was evaluated on the remaining frames by means of the pixel-wise measure *KT* for the center joint CF for thresholds between 15 and 40 pixels. Although two instruments are shown in the sequence, we perform the experiment only for Tool 1, which is more interesting for pose estimation due to grasping operations and various movements. Tool 2 remains relatively static and closed. As depicted in Fig. 18, the proposed method performs similar to the baseline algorithm DDVT (Sznitman et al., 2012) in terms of prediction of the keypoint CF. In contrast to the other methods, our algorithm did not need to be reinitialized and was able to track all three joints of the articulated instrument for the entire sequence.

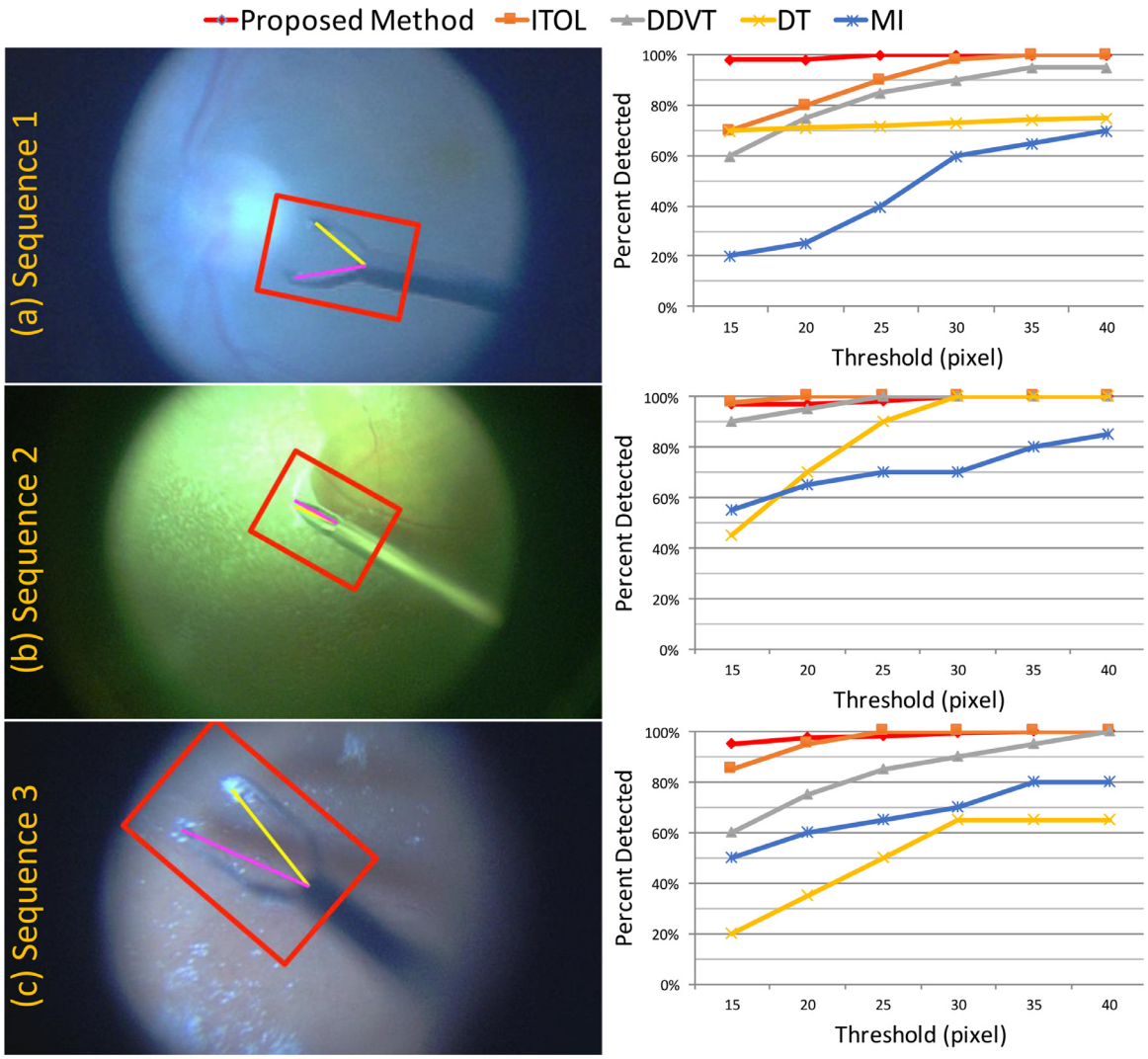
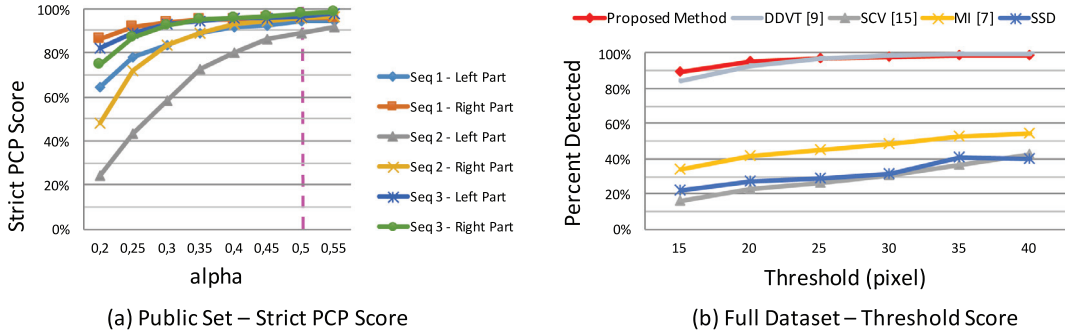


Fig. 16. Results for sequential evaluation of public dataset: For every sequence separately, the forests are trained on the first half and tested on the remaining half. The result for the central joint (CF) is compared analogously to the cited works by means of threshold distance in pixel (*KT*). The compared methods are ITOL and DT presented in the work by Li et al. (2014), DDVT by Sznitman et al. (2012) and MI by Richa et al. (2011).



(a) Public Set – Strict PCP Score

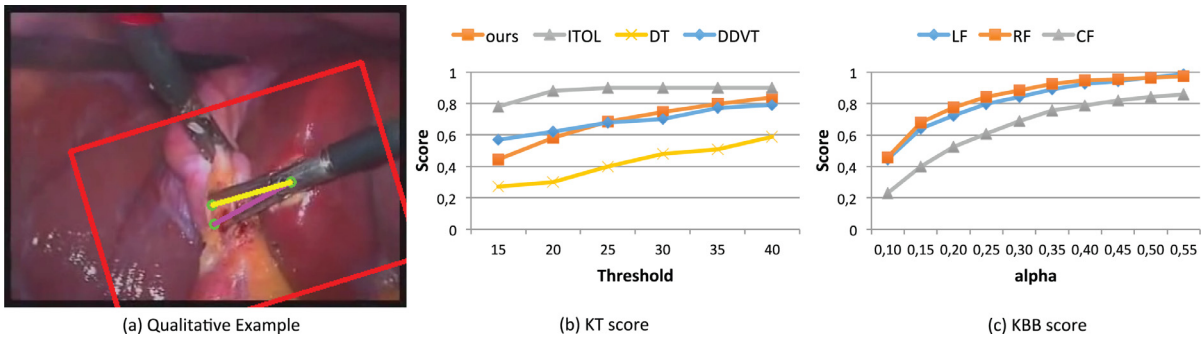
(b) Full Dataset – Threshold Score

Fig. 17. Results for public dataset: In (a), the strict PCP scores for learning and testing on the separate sequences is visualized. The vertical pink line represents the standard value for alpha in human pose estimation. (b) depicts the *KT* score for the estimation of the central joint (CF) when training the forest on all halves of the sequences and evaluating on the second halves. The compared methods are DDVT by Sznitman et al. (2012), SCV by Pickering et al. (2009), MI by Richa et al. (2011) and SSD. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

6. Discussion and concluding remarks

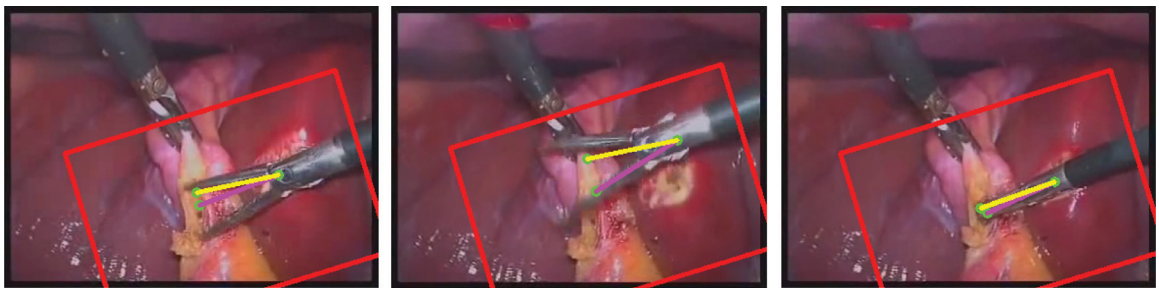
In this paper, we presented a robust framework for tracking a surgical instrument in *in-vivo* RM sequences. In contrast to other methods, which focus on estimating just the center joint of a forceps, we recover the tool's articulated pose in real-time. To withstand noisy and incomplete data, we have proposed to base both parts of the algorithm on random forest, which has shown to be a fast, flexible and robust machine learning tool for a variety of tasks.

The advantage of separating the problem into two tasks is twofold. On the one hand, the algorithm can make use of both color



(a) Qualitative Example (b) KT score (c) KBB score

Fig. 18. Results for laparoscopy sequence: In (a), a qualitative example of the estimation is visualized. Figure (b) depicts the KT metric for the central joint (CF) in comparison to the methods DDVT by Sznitman et al. (2012), DT and ITOL by Li et al. (2014). In (c), results for all joints by means of the metric *KBB* are shown.



(a) Wrong Pose (b) Scale Change (c) Shifted CF

Fig. 19. Failure cases for laparoscopy sequence: in some cases, the proposed method has problems with the localization of the keypoints. In (a) the position of the CF and RF are predicted correctly, but the left tool tip (LF) is distant to the ground truth, which results in an incorrect pose. In (b), the significant scale change of the size of the instrument leads to totally shifted localizations. In (c) the tool tips are inserted into tissue. Consequently, the geometric relation between the tool tips and the center joint is changed.

as well as gradient information. On the other hand, the computationally more expensive step of extracting gradient information is reduced to a smaller region of interest (i.e., for the sake of pose estimation only), thus making our algorithm particularly efficient. Please note that even if the prominent color of the background changes sensibly, the contrast between the metallic appearance of the instrument and the retina remains a valuable and easily accessible cue. With less than 2 ms of computation time using one CPU core, the color-based temporal tracker takes advantage of the contrast to efficiently localize the position of the instrument tip. However, color information tends to be less reliable for precise estimation in RM sequences, due to typically strong illumination and appearance changes. For this reason, we employed gradient information for pose estimation in the second step. Differently from the approach by Sznitman et al. (2014), we do not use gradient information from patches within the entire frame, but limit this computationally expensive step to the region of interest provided by the tracker. Another important difference is that the tracker relies on temporal information available from previous frames of the sequence, while the pose estimation stage only exploits the information available from the current frame.

The performance of the proposed methods was evaluated on three different datasets by means of four different metrics. The results show that our method can not only handle unseen changes within a sequence, but also generalize for various illumination and instrument appearance changes. In particular, the complete evaluation of the Instrument Dataset was one of the most challenging experiment including 18 sequences of four different instrument shapes. Our algorithm yielded a strict PCP score ($\alpha = 0.5$) of more than 95% for both the left and right parts of the forceps, and 84.1% for the *LF*, 90.8% for the *RF* and 83.2% for the *CF* in terms of *KBB* with $\alpha = 0.2$. In contrast to the *KT* metric, the newly introduced *KBB* metric takes into account the variations in instrument appearance size and image pixel resolution and thereby allows the performance evaluation across sequences. In the experiments on the laparoscopic instrument sequence, the pose estimation revealed difficulties regarding large scale changes of the instrument size (Fig. 19). This could be caused by the fact that the HoG features are extracted with a fixed patch size and can be tackled by extending the tracker so that it estimates the full rigid transformation update for the template. However, looking at Fig. 18, the performance of the proposed method is still comparable to state-of-the-art methods.

The proposed method is designed for one single instrument. However, the simultaneous tracking of several tools can easily be realized by initializing a separate thread of the algorithm for every instrument. An important observation is that our method is not confused by the presence of another tool in the image frame. In RM sequences, usually only one forceps is utilized. In contrast to the *Public* dataset, most parts of the sequences within the novel *Instrument* dataset include the intra-ocular light in the focused area. Being a metallic and rigid device, it is similar in appearance to the tracked forceps, and as such could be considered as a second tool. In some sequences (e.g. Seq. 5 and Seq. 15, see Figs. 9 and 10), the light source is very close to the tool tip. Even in this challenging situation, the proposed algorithm is not misled by this additional nuisance. Also in the laparoscopy dataset, the presence of a second forceps does not deteriorate the performance.

One limitation of our method is the lack of a recovery procedure in case of tracking failures. The pose estimation relies on the output of the tracker, which is the bounding box containing the tool. This region of interest does not necessarily have to be precise because the final prediction is produced by the pose estimation. However, if the instrument is not captured by the bounding box the pose estimation would also fail at inferring the instrument joints. Nevertheless, in all our experiments, this case did not occur

and therefore the tracker did not have to be re-initialized with the ground truth, if the tool was present in the frame and showed a continuous movement. A detector would further increase the robustness of our method and make it more suitable for the clinical practice.

An interesting future direction is represented by the use of the inferred pose as an additional input for the tracker, within a closed-loop framework where the pose estimation stage also provides feedback for the tracker. This brings the challenge of synergically combining the predictions from two forests to achieve a better performance.

Acknowledgments

This research is partially supported by the CARC grant (identification number IUK456/002) and Carl Zeiss Meditec AG, Munich.

References

Allan, M., Chang, P.-L., Ourselin, S., Hawkes, D.J., Sridhar, A., Kelly, J., Stoyanov, D., 2015. Image based surgical instrument pose estimation with multi-class labelling and optical flow. In: Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015. Springer, pp. 331–338.

Allan, M., Ourselin, S., Thompson, S., Hawkes, D.J., Kelly, J., Stoyanov, D., 2013. Toward detection and localization of instruments in minimally invasive surgery. In: IEEE Transactions on Biomedical Engineering, 60, pp. 1050–1058.

Baek, Y.M., Tanaka, S., Kanako, H., Sugita, N., Morita, A., Sora, S., Mochizuki, R., Mitsuishi, M., 2012. Full state visual forceps tracking under a microscope using projective contour models. In: Proceedings of IEEE ICRA, pp. 2919–2925.

Baker, S., Matthews, I., 2004. Lucas-kanade 20 years on: a unifying framework. IJCV 56 (3), 221–255.

Belagiannis, V., Amann, C., Navab, N., Ilic, S., 2014. Holistic human pose estimation with regression forests. In: Perales, F.J., Santos-Victor, J. (Eds.), AMDO 2014. LNCS, 8563. Springer, Heidelberg, pp. 20–30.

Blum, T., Sielhorst, T., Navab, N., 2007. Advanced augmented reality feedback for teaching 3d tool manipulation.

Breiman, L., 2001. Random forests. Mach. Learn. 45 (1), 5–32. doi:10.1023/A:1010933404324.

Cattin, P.C., Bay, H., Van Gool, L., Székely, G., 2006. Retina mosaicing using local features. In: Medical Image Computing and Computer-Assisted Intervention–MICCAI 2006. Springer, pp. 185–192.

Chen, C.-J., Huang, W.-W., Song, K.-T., 2013. Image tracking of laparoscopic instrument using spiking neural networks. In: ICCAS 2013, pp. 951–955.

Dalal, N., Triggs, B., 2005. Histograms of oriented gradients for human detection. In: Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on, 1. IEEE, pp. 886–893.

Ehlers, J., Kaiser, P.K., Srivastava, S.K., 2014. Intraoperative optical coherence tomography using the rescan 700: preliminary results from the discover study. Br. J Ophthalmol. 1329–1332.

Eitz, M., Hildebrand, K., Boubekeur, T., Alexa, M., 2011. Sketch-based image retrieval: Benchmark and bag-of-features descriptors. Vis. Comput. Graph., IEEE Trans. 17 (11), 1624–1636.

Felzenszwalb, P.F., Girshick, R.B., McAllester, D., Ramanan, D., 2010. Object detection with discriminatively trained part-based models. Pattern Anal. Mach. Intell., IEEE Trans. 32 (9), 1627–1645.

Gabriele, M.L., Wollstein, G., Ishikawa, H., Kagemann, L., Xu, J., Folio, L.S., Schuman, J.S., 2011. Optical coherence tomography: History, current status, and laboratory work. Invest. Ophthalmol. & Vis. Sci. 2425–2436.

Holzer, S., Pollefeys, M., Ilic, S., Tan, D.J., Navab, N., 2012. Online learning of linear predictors for real-time tracking. In: 12th European Conference on Computer Vision (ECCV).

Jurie, F., Dhome, M., 2002. Hyperplane approximation for template matching. PAMI 24 (7), 996–1000.

Kalal, Z., Mikolajczyk, K., Matas, J., 2012. Tracking-learning-detection. Pattern Anal. Mach. Intell., IEEE Trans. 34 (7), 1409–1422.

Li, Y., Chen, C., Huang, X., Huang, J., 2014. Instrument tracking via online learning in retinal microsurgery. In: Golland, P., et al. (Eds.), MICCAI 2014. LNCS, 8673. Springer, Heidelberg, pp. 464–471.

Nilsback, M.-E., Zisserman, A., 2008. Automated flower classification over a large number of classes. In: Computer Vision, Graphics & Image Processing, 2008. ICVGIP'08. Sixth Indian Conference on. IEEE, pp. 722–729.

Pavlidis, M., Georgalas, I., K÷rber, N., 2015. Determination of a new parameter, elevated epiretinal membrane, by en face oct as a prognostic factor for pars plana vitrectomy and safer epiretinal membrane peeling. J. Ophthalmol.

Pezzementi, Z., Voros, S., Hager, G.D., 2009. Articulated object tracking by rendering consistent appearance parts. ICRA 2009, pp. 3940–3947 (2009).

Pickering, M.R., Muhit, A.A., Scarvell, J.M., Smith, P.N., 2009. A new multi-modal similarity measure for fast gradient-based 2d-3d image registration. In: EMBC 2009, pp. 5821–5824.

Reiter, A., Allen, P.K., Zhao, T., 2012. Feature classification for tracking articulated surgical tools. In: Ayache, N., Delingette, H., Golland, P., Mori, K. (Eds.), MICCAI 2012, Part II. LNCS, 7511. Springer, Heidelberg, pp. 592–600.

Richa, R., Balicki, M., Meisner, E., Sznitman, R., Taylor, R., Hager, G., 2011. Visual tracking of surgical tools for proximity detection in retinal surgery. In: IPCAI, pp. 55–66.

Rieke, N., Duca, S., Navab, N., Eslami, A., 2016. Automatic ioct positioning during membrane peeling via real-time high resolution surgical forceps tracking. In: International Society of Imaging in the Eye Conference, Association for Research in Vision and Ophthalmology (ARVO 2016), To be published. Seattle, USA

Rieke, N., Tan, D.J., Alsheakhali, M., Tombari, F., Amat di San Filippo, C., Belagiannis, V., Eslami, A., Navab, N., 2015. Surgical tool tracking and pose estimation in retinal microsurgery. In: Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015. Springer, pp. 266–273.

Roodaki, H., Filippatos, K., Eslami, A., Navab, N., 2015. Introducing augmented reality to optical coherence tomography in ophthalmic microsurgery. In: 2015 IEEE International Symposium on Mixed and Augmented Reality, ISMAR 2015, Fukuoka, Japan, September 29 – Oct. 3, 2015, pp. 1–6.

Sznitman, R., Ali, K., Richa, R., Taylor, R.H., Hager, G.D., Fua, P., 2012. Data-driven visual tracking in retinal microsurgery. In: Ayache, N., Delingette, H., Golland, P., Mori, K. (Eds.), MICCAI 2012, Part II. LNCS, 7511. Springer, Heidelberg, pp. 568–575.

Sznitman, R., Basu, A., Richa, R., Handa, J., Gehlbach, P., Taylor, R.H., Jedynak, B., Hager, G.D., 2011. Unified detection and tracking in retinal microsurgery. In: Fichtinger, G., Martel, A., Peters, T. (Eds.), MICCAI 2011, Part I. LNCS, 6891. Springer, Heidelberg, pp. 1–8.

Sznitman, R., Becker, C., Fua, P., 2014. Fast part-based classification for instrument detection in minimally invasive surgery. In: Golland, P., et al. (Eds.), MICCAI 2014. LNCS, 8673. Springer, Heidelberg, pp. 692–699.

Tan, D.J., Ilic, S., 2014. Multi-forest tracker: A chameleon in tracking. In: CVPR 2014, pp. 1202–1209.

Yang, Y., Ramanan, D., 2013. Articulated human detection with flexible mixtures of parts. Pattern Anal. Mach. Intell., IEEE Trans. 35 (12), 2878–2890.

Yigitsoy, M., Belagiannis, V., Djurka, A., Katouzian, A., Ilic, S., Pernus, E., Eslami, A., Navab, N., 2015. Random ferns for multiple target tracking in microscopic retina image sequences. In: Biomedical Imaging (ISBI), 2015 IEEE 12th International Symposium on. IEEE, pp. 209–212.

Real-Time Online Adaption for Robust Instrument Tracking and Pose Estimation

Nicola Rieke[1], David Joseph Tan[1], Federico Tombari[1,3], Josué Page Vizcaíno[1], Chiara Amat di San Filippo[1], Abouzar Eslami[4], and Nassir Navab[1,2]

[1] Chair for Computer Aided Medical Procedures (CAMP), TU Munich, Germany.
[2] Johns Hopkins University, Baltimore, USA.
[3] DISI, University of Bologna, Italy
[4] Carl Zeiss MEDITEC München, Germany.

Contribution. The main contributions of this publication, including the main idea of the closed-loop approach with online refinement of the offline learned Random Forest, evaluation of the method and writing of the manuscript were done and coordinated by the author of this thesis. The validation of the method and the revision of the publication was done jointly together with the co-authors.

Real-Time Online Adaption for Robust Instrument Tracking and Pose Estimation

Nicola Rieke[1], David Joseph Tan[1], Federico Tombari[1,4], Josué Page Vizcaíno[1], Chiara Amat di San Filippo[3], Abouzar Eslami[3], and Nassir Navab[1,2]

[1] Computer Aided Medical Procedures, Technische Universität München, Germany
[2] Computer Aided Medical Procedures, Johns Hopkins University, Baltimore, USA
[3] Carl Zeiss MEDITEC München, Germany
[4] DISI, University of Bologna, Italy
Nicola.Rieke@tum.de

Abstract. We propose a novel method for instrument tracking in Retinal Microsurgery (RM) which is apt to withstand the challenges of RM visual sequences in terms of varying illumination conditions and blur. At the same time, the method is general enough to deal with different background and tool appearances. The proposed approach relies on two random forests to, respectively, track the surgery tool and estimate its 2D pose. Robustness to photometric distortions and blur is provided by a specific online refinement stage of the offline trained forest, which makes our method also capable of generalizing to unseen backgrounds and tools. In addition, a peculiar framework for merging together the predictions of tracking and pose is employed to improve the overall accuracy. Remarkable advantages in terms of accuracy over the state-of-the-art are shown on two benchmarks.

1 Introduction and Related Work

Retinal Microsurgery (RM) is a challenging task wherein a surgeon has to handle anatomical structures at micron-scale dimension while observing targets through a stereo-microscope. Novel imaging modalities such as interoperative Optical Coherence Tomography (iOCT) [1] aid the physician in this delicate task by providing anatomical sub-retinal information, but lead to an increased workload due to the required manual positioning to the region of interest (ROI). Recent research has aimed at introducing advanced computer vision and augmented reality techniques within RM to increase safety during surgical maneuvers and to simplify the surgical workflow. A key step for most of these methods is represented by an accurate and real-time localization of the instrument tips, which allows to automatically position the iOCT according to it. This further enables to calculate the distance of the instrument tip to the retina and to provide a real-time feedback to the physician. In addition, the trajectories performed by the instrument during surgery can be compared with other surgeries, thus paving the way to objective quality assessment for RM. Surgical tool tracking has been investigated in different medical specialties: nephrectomy [2], neurosurgery [3],

laparoscopy/endoscopy [4, 5]. However, RM presents specific challenges such as strong illumination changes, blur and variability of surgical instruments appearance, that make the aforementioned approaches not directly applicable in this scenario. Among the several works recently proposed in the field of tool tracking for RM, Pezzementi *et al.* [6] suggested to perform the tracking in two steps: first via appearance modeling, which computes a pixel-wise probability of class membership (foreground/background), then filtering, which estimates the current tool configuration. Richa *et al.* [7] employ mutual information for tool tracking. Snitzman *et al.* [8] introduced a joint algorithm which performs simultaneously tool detection and tracking. The tool configuration is parametrized and tracking is modeled as a Bayesian filtering problem. Succesively, in [9], they propose to use a gradient-based tracker to estimate the tool's ROI followed by foreground/background classification of the ROI's pixels via boosted cascade. In [10], a gradient boosted regression tree is used to create a multi-class classifier which is able to detect different parts of the instrument. Li *et al.* [11] present a multi-component tracking, i.e. a gradient-based tracker able to capture the movements and an online-detector to compensate tracking losses.

In this paper, we introduce a robust closed-loop framework to track and localize the instrument parts in *in-vivo* RM sequences in real-time, based on the dual-random forest approach for tracking and pose estimation proposed in [12]. A fast tracker directly employs the pixel intensities in a random forest to infer the tool tip bounding box in every frame. To cope with the strong illumination changes affecting the RM sequences, one of the main contributions of our paper is to adapt the offline model to online information while tracking, so to incorporate the appearance changes learned by the trees with real photometric distortions witnessed at test time. This offline learning - online adaption leads to a substantial capability regarding the generalization to unseen sequences. Secondly, within the estimated bounding box, another random forest predicts the locations of the tool joints based on gradient information. Differently from [12], we enforce spatial temporal constraints by means of a Kalman filter [13]. As a third contribution of this work, we propose to "close the loop" between the tracking and 2D pose estimation by obtaining a joint prediction concerning the template position acquired by merging the outcome of the two separate forests through the confidence of their estimation. Such cooperative prediction will in turn provide pose information for the tracker, improving its robustness and accuracy. The performance of the proposed approach is quantitatively evaluated on two different *in-vivo* RM datasets, and demonstrate remarkable advantages with respect to the state-of-the-art in terms of robustness and generalization.

2 Method

In this section, we discuss the proposed method, for which an overview is depicted in Fig. 1. First, a fast intensity-based **tracker** locates a template around the instrument tips using an offline trained model based on random forest (RF) and the location of the template in the previous frame. Within this ROI, a **pose**

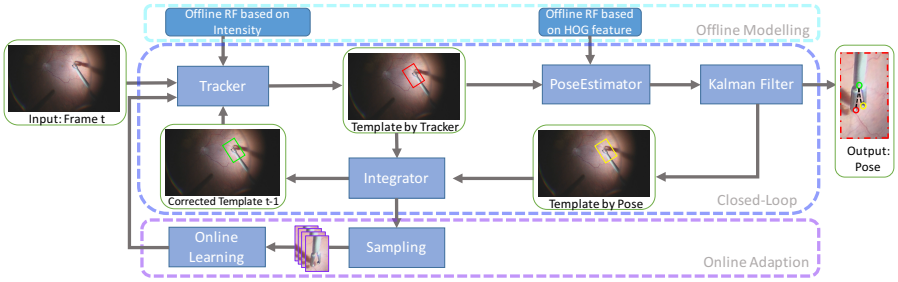


Fig. 1: **Framework:** The description of the Tracker, Sampling and Online Learning can be found in Sec. 2.1. The Pose Estimator and Kalman Filter is presented in Sec. 2.2. Details on the Integrator are given in Sec. 2.3.

estimator based on HOG recovers the three joints employing another offline learned RF and filters the result by temporal-spatial constraints. To close the loop, the output is propagated to an **integrator**, aimed at merging together the intensity-based and gradient-based predictions in a synergic way in order to provide the tracker with an accurate template location for the prediction in the next frame. Simultaneously, the refined result is propagated to a separate thread which adapts the model of the tracker to the current data characteristics via **online learning**.

A central element in this approach is the definition of the tracked template, which we define by the landmarks of the forceps. Let $(L, R, C)^\top \in \mathbb{R}^{2\times 3}$ be the left, right and central joint of the instrument, then the midpoint between the tips is given by $M = \frac{L+R}{2}$ and the 2D similarity transform from the patch coordinate system to the frame coordinate system can be defined as

$$\mathbf{H} = \begin{bmatrix} s\cdot\cos(\theta) & -s\cdot\sin(\theta) & C_x \\ s\cdot\sin(\theta) & s\cdot\cos(\theta) & C_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 30 \\ 0 & 0 & 1 \end{bmatrix}$$

with $s = \frac{b}{100}\cdot\max\{\|L-C\|_2,\ \|R-C\|_2\}$ and $\theta = \cos^{-1}\left(\frac{M_y - C_y}{\|M-C\|_2}\right)$ for a fixed patch size of 100×150 pixel and $b \in \mathbb{R}$ defining the relative size. In this way, the entire instrument tip is enclosed by the template and aligned with the tool's direction. In the following, details of the different components are presented.

2.1 Tracker – Offline Learning, Online Adaption

Derived from image registration, tracking aims to determine a transformation parameter that minimizes the similarity measure to a given template. In contrast to attaining a single template, the tool undergoes an articulated motion and a variation of lighting changes which is difficult to minimize as an energy function. Thus, the tracker learns a generalized model of the tool based on multiple templates, taken as the tool undergoes different movements in a variety

of environmental settings, and predicts the translation parameter from the intensity values at n random points $\{\mathbf{x}_p\}_{p=1}^{n}$ within the template, similar to [12]. In addition, we assume a piecewise constant velocity from consecutive frames. Therefore, given the image $\mathbf{I}_t$ at time t and the translation vector of the template from $t-2$ to $t-1$ as $\mathbf{v}_{t-1} = (v_x, v_y)^\top$, the input to the forest is a feature vector concatenating the intensity values on the current location of the template $\mathbf{I}_t(\mathbf{x}_p)$ with the velocity vector $\mathbf{v}_{t-1}$, assuming a constant time interval. In order to learn the relation between the feature vector and the transformation update, we use a random forest that follows a dimension-wise splitting of the feature vector such that the translation vector on the leaves point to a similar location.

The cost of generalization is the inadequacy to describe the conditions that are specific to a particular situation, such as the type of tool used in the surgery. As a consequence, the robustness of the tracker is affected, since it cannot confidently predict the location of the template for challenging frames with high variations from the generalized model. Hence, in addition to the offline learning for a generalized tracker, we propose to perform an online learning strategy that considers the current frames and learns the relation of the translation vector with respect to the feature vector. The objective is to stabilize the tracker by adapting its forest to the specific conditions at hand. In particular, we propose to incrementally add new trees to the forest by using the predicted template location on the current frames of the video sequence. To achieve this goal, we impose random synthetic transformations on the bounding boxes that enclose the templates to build the learning dataset with pairs of feature and translation vectors, such that the transformations emulate the motion of the template between two consecutive frames. Thereafter, the resulting trees are added to the existing forest and the prediction for the succeeding frames include both the generalized and environment-specific trees. Notably, our online learning approach does not learn from all the incoming frames, but rather introduces in Sec. 2.3 a confidence measure to evaluate and accumulate templates.

2.2 2D Pose Estimation with temporal-spatial Constraints

During pose estimation, we model a direct mapping between image features and the location of the three joints in the 2D space of the patch. Similar to [12], we employ HOG features around a pool of randomly selected pixel locations within the provided ROI as an input to the trees in order to infer the pixel offsets to the joint positions. Since the HOG feature vector is extracted as in [14], the splitting function of the trees considers only one dimension of the vector and is optimized by means of information gain. The final vote is aggregated by a dense-window algorithm. The predicted offsets to the joints in the reference frame of the patch are back-warped onto the frame coordinate system. Up to now, the forest considers every input as a still image. However, the surgical movement is usually continuous. Therefore, we enforce a temporal-spatial relationship for all joint locations via a Kalman filter [13] by employing the 2D location of the joints in the frame coordinate system and their frame-to-frame velocity.

2.3 Closed Loop via Integrator

Although the combination of the pose estimation with the Kalman filter would already define a valid instrument tracking for all three joints, it completely relies on the gradient information, which may be unreliable in case of blurred frames. In these scenarios, the intensity information is still a valid source for predicting the movement. On the other hand, gradient information tends to be more reliable for precise localization in focused images. Due to the definition of the template, the prediction of the joint positions can directly be connected to the expected prediction of the tracker via the similarity transform. Depending on the confidence for the current prediction of the separate random forests, we define the scale s_F and the translation t_F of the joint similarity transform as the weighted average

$$s_F = \frac{s_T \cdot \sigma_P + s_P \cdot \sigma_T}{\sigma_T + \sigma_P} \quad \text{and} \quad t_F = \frac{t_T \cdot \sigma_P + t_P \cdot \sigma_T}{\sigma_T + \sigma_P}$$

where σ_T and σ_P are the average standard deviation of the tracking prediction and pose prediction, respectively, and the t_F is set to be greater than or equal to the initial translation. In this way, the final template is biased towards the more reliable prediction. If σ_T is higher than a threshold τ_σ, the tracker transmits the previous location of the template, which is subsequently corrected by the similarity transform of the predicted pose. Furthermore, the prediction of the pose can also correct for the scale of the 2D similarity transform which is actually not captured by the tracker, leading to a scale adaptive tracking. This is an important improvement because an implicit assumption of the pose algorithm is that the size of the bounding box corresponds to the size of the instrument due to the HOG features. The refinement also guarantees that only reliable templates are used for the online learning thread.

3 Experiments and Results

We evaluated our approach on two different datasets ([9, 12]), which we refer to as *Szn*- and *Rie*-dataset, respectively. We considered both datasets because of their intrinsic difference: the first one presents a strong coloring of the sequences and a well-focused ocular of the microscope; the second presents different types of instruments, changing zoom factor, presence of light source and presence of detached epiretinal membrane. Further information on the dataset can be found in Table 1 and in [9, 12]. Analogously to baseline methods, we evaluate the performance of our method by means of a threshold measure [9] for the separate joint predictions and the strict PCP score [15] for evaluating the parts connected by the joints. The proposed method is implemented in C++ and runs at 40 fps on a Dell Alienware Laptop, Intel Core i7-4720HQ @ 2.6GHz and 16 GB RAM. In the offline learning for the tracker, we trained 100 trees per parameter, employed 20 random intensity values and velocity as feature vectors, and used 500 sample points. For the pose estimation, we used 15 trees and the HOG features are set to a bin size of 9 and pixel size resolution of 50×50.

	Set	*Szn* [9]	*Rie* [12]
# Frames	I	402	200
	II	222	200
	III	547	200
	IV	—	200
Resolution		640×480	1920×1080

Table 1: Summary of the datasets.

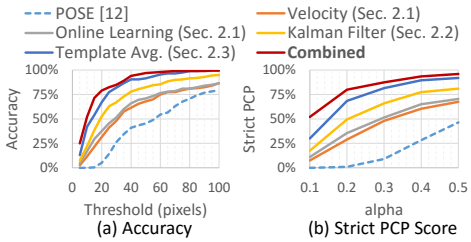


Fig. 2: Component evaluation.

3.1 Evaluation of Components

To analyze the influence of the different proposed components, we evaluate the algorithm with different settings on the *Rie*-dataset, whereby the sequences I, II and III are used for the offline learning and sequence IV is used as the test sequence. Fig. 2 shows the threshold measure for the left tip in (a) and the strict PCP for the left fork in (b). Individually, each component excels in performance and contribute to a robust performance when combined. Among them, the most prominent improvement is the weighted averaging of the templates from Sec. 2.3.

3.2 Comparison to State-of-the-Art

We compare the performance of our method against the state-of-the-art methods DDVT [9], MI [7], ITOL [11] and POSE [12]. Throughout the experiments on the *Szn*-dataset, the proposed method can compete with state-of-the-art methods, as depicted in Fig. 3. In the first experiment, in which the forest are learned on the first half of a sequence and evaluated on the second half, our method reaches

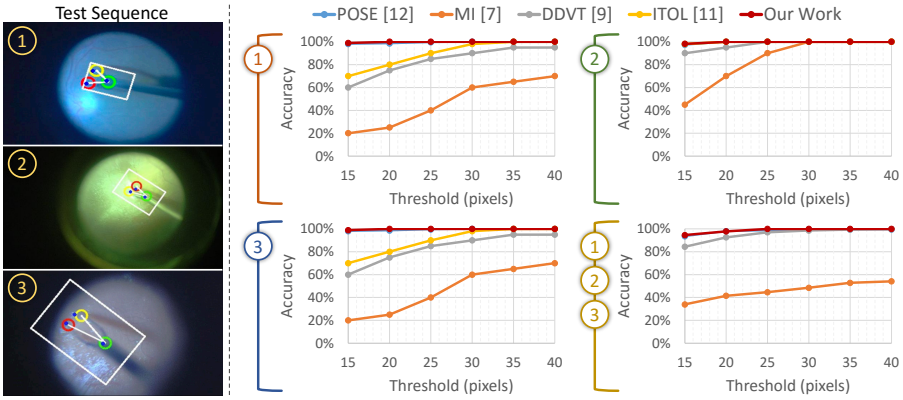


Fig. 3: ***Szn*-dataset:** Sequential and combined evaluation for sequence 1-3. For over 93%, the results are so close that the single graphs are not distinguishable.

Table 2: Strict PCP for Cross Validation of *Rie*-Dataset for **L**eft and **R**ight Fork.

Methods	Set I (L/R)	Set II (L/R)	Set III (L/R)	Set IV (L/R)
Our Work	**89.0/88.5**	**98.5/99.5**	**99.5/99.5**	**94.5/95.0**
POSE [12]	69.7/58.5	93.94/93.43	94.47/94.47	46.46/57.71

an accuracy of at least 94.3% by means of threshold distance for the central joint. In the second experiment, all the first halves of the sequences are included into the learning database and tested on the second halves.

In contrast to the *Szn*-dataset, the *Rie*-dataset is not as saturated in terms of accuracy and therefore the benefits of our methods are more evident. Fig. 4 illustrates the results for the cross-validation setting, i.e. the offline training is performed on three sequences and the method is tested on the remaining one. In this case, our method outperforms POSE for all test sequences. Notably, there is a significant improvement in accuracy for the *Rie*-Set IV which demonstrates the generalization capacity of our method for unseen illumination and instrument. Table 2 also reflects this improvement in the strict PCP scores which indicate that our method is nearly twice as accurate as the baseline method [12].

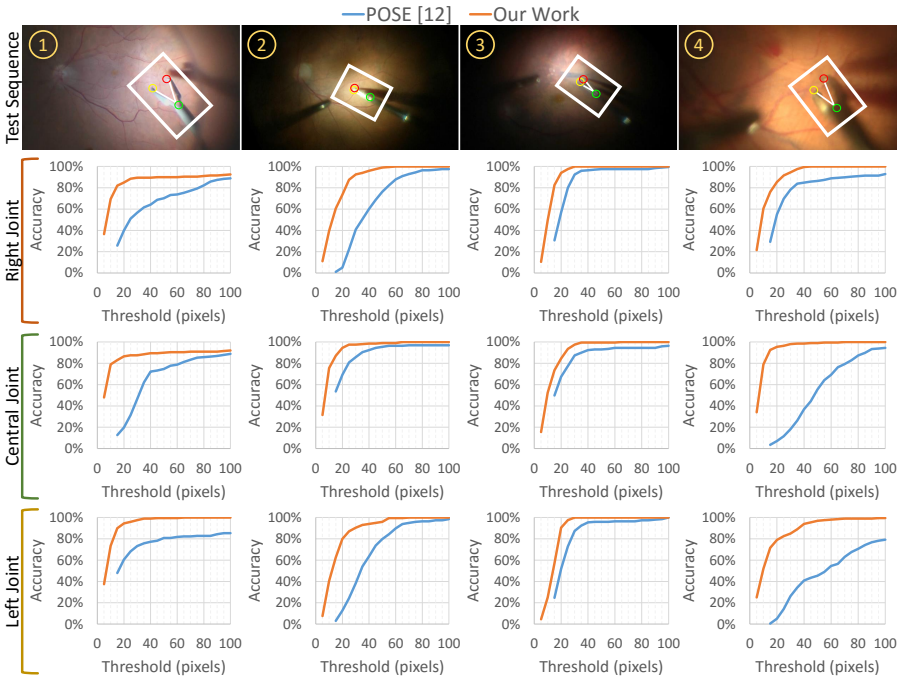


Fig. 4: ***Rie*-dataset:** Cross validation evaluation – the offline forests are learned on three sequences and tested on the unseen one.

4 Conclusion

In this work, we propose a closed-loop framework for tool tracking and pose estimation, which runs at 40 fps. A combination of separate predictors yields robustness which is able to withstand the challenges of RM sequences. The work further shows the method's capability to generalize to unseen instruments and illumination changes by allowing an online adaption. These key drivers allow our method to outperform state-of-the-art on two benchmark datasets.

References

1. Ehlers, J.P., Kaiser, P.K., Srivastava, S.K.: Intraoperative optical coherence tomography using the rescan 700: preliminary results from the discover study. British Journal of Ophthalmology (2014) 1329–1332
2. Reiter, A., Allen, P.K.: An online learning approach to in-vivo tracking using synergistic features. In: IROS. (2010) 3441–3446
3. Bouget, D., Benenson, R., Omran, M., Riffaud, L., Schiele, B., Jannin, P.: Detecting surgical tools by modelling local appearance and global shape. IEEE Transactions on Medical Imaging **34**(12) (Dec 2015) 2603–2617
4. Allan, M., Chang, P.L., Ourselin, S., Hawkes, D., Sridhar, A., Kelly, J., Stoyanov, D.: Image based surgical instrument pose estimation with multi-class labelling and optical flow. In: MICCAI. (2015) 331–338
5. Wolf, R., Duchateau, J., Cinquin, P., Voros, S.: 3d tracking of laparoscopic instruments using statistical and geometric modeling. In: MICCAI. (2011) 203–210
6. Pezzementi, Z., Voros, S., Hager, G.D.: Articulated object tracking by rendering consistent appearance parts. In: ICRA. (2009) 3940–3947
7. Richa, R., Balicki, M., Meisner, E., Sznitman, R., Taylor, R., Hager, G.: Visual tracking of surgical tools for proximity detection in retinal surgery. In: IPCAI. (2011) 55–66
8. Sznitman, R., Basu, A., Richa, R., Handa, J., Gehlbach, P., Taylor, R.H., Jedynak, B., Hager, G.D.: Unified detection and tracking in retinal microsurgery. In: MICCAI. (2011) 1–8
9. Sznitman, R., Ali, K., Richa, R., Taylor, R.H., Hager, G.D., Fua, P.: Data-driven visual tracking in retinal microsurgery. In: MICCAI. (2012) 568–575
10. Sznitman, R., Becker, C., Fua, P.: Fast part-based classification for instrument detection in minimally invasive surgery. In: MICCAI. (2014) 692–699
11. Li, Y., Chen, C., Huang, X., Huang, J.: Instrument tracking via online learning in retinal microsurgery. In: MICCAI. (2014) 464–471
12. Rieke, N., Tan, D.J., Alsheakhali, M., Tombari, F., Amat di San Filippo, C., Belagiannis, V., Eslami, A., Navab, N.: Surgical tool tracking and pose estimation in retinal microsurgery. MICCAI (2015) 266–273
13. Haykin, S.S.: Kalman Filtering and Neural Networks. J. Wiley & Sons, Inc. (2001)
14. Felzenszwalb, P.F., Girshick, R.B., McAllester, D., Ramanan, D.: Object detection with discriminatively trained part-based models. PAMI **32**(9) (2010) 1627–1645
15. Ferrari, V., Marin-Jimenez, M., Zisserman, A.: Progressive search space reduction for human pose estimation. In: CVPR. (2008) 1–8

Concurrent Segmentation and Localization for Tracking of Surgical Instruments

Iro Laina*[1], Nicola Rieke*[1], Christian Rupprecht [1,2], Josué Page Vizcaíno[1], Abouzar Eslami[3] , Federico Tombari[1], Nassir Navab[1,2]

[1] Chair for Computer Aided Medical Procedures (CAMP), TU Munich, Germany.
[2] Johns Hopkins University, Baltimore, USA.
[3] Carl Zeiss MEDITEC München, Germany.

Contribution. The main contributions of this publication, including the main idea of simultaneous segmentation and 2D instrument pose estimation, dataset preparation and validation were done and coordinated by the author of this thesis. Iro Laina, joint first author, was responsible for the implementation of the up-sampling layers and the regression tasks. Both joint first authors contributed equally to the development of the method and the writing of the manuscript. Co-authors contributed to the revision of the manuscript and to evaluation of alternative methods on the same dataset.

Concurrent Segmentation and Localization for Tracking of Surgical Instruments

Iro Laina[1]*, Nicola Rieke[1]*, Christian Rupprecht[1,2], Josué Page Vizcaíno[1], Abouzar Eslami[3], Federico Tombari[1], and Nassir Navab[1,2]

[1] Computer Aided Medical Procedures (CAMP), TU Munich, Germany
[2] Johns Hopkins University, Baltimore, USA
[3] Carl Zeiss MEDITEC München, Germany
Nicola.Rieke@tum.de

Abstract. Real-time instrument tracking is a crucial requirement for various computer-assisted interventions. To overcome problems such as specular reflection and motion blur, we propose a novel method that takes advantage of the interdependency between localization and segmentation of the surgical tool. In particular, we reformulate the 2D pose estimation as a heatmap regression and thereby enable a robust, concurrent regression of both tasks via deep learning. Throughout experimental results, we demonstrate that this modeling leads to a significantly better performance than directly regressing the tool position and that our method outperforms the state-of-the-art on a Retinal Microsurgery benchmark and the MICCAI EndoVis Challenge 2015.

1 Introduction and Related Work

In recent years there has been significant progress towards computer-based surgical assistance in Minimally Invasive Surgery (MIS) and Retinal Microsurgery (RM). One of the key components is tracking and segmentation of surgical instruments during the intervention, which enables for example proximity estimation to the retina in RM or detecting suitable regions for a graphical overlay of additional information without obstructing the surgeon's view. Marker-free approaches are particularly desirable for this task as they do not interfere with the surgical workflow or require modifications to the tracked instrument. Despite recent advances, the vision-based tracking of surgical tools in *in-vivo* scenarios remains challenging, as summarized by Bouget *et al.* [1], mainly due to nuisances such as strong illumination changes and blur. Prior work in the field relies on handcrafted features, such as Haar wavelets [2], HoG [3, 4] or color features [5], which come with their own advantages and disadvantages. While color features, for example, are computationally cheap, they are not robust towards strong illumination changes which are frequently present during the surgery. Gradients, on the other hand, are not reliable to withstand the typical motion blur of the tools.

* I. Laina and N. Rieke contributed equally to this work.

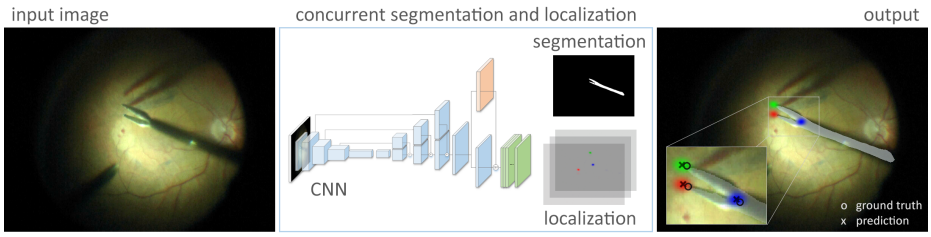


Fig. 1. Overview of the proposed method (CSL): Concurrent semantic segmentation and landmark localization with a CNN-based approach.

Rieke *et al.* [6] employed both feature types in two separate Random Forests and proposed to adaptively choose the more reliable one. Since their explicit feature representation incorporates implicit simplifications, this tends to limit the generalization power of the forests. Sarikaya *et al.* [7] present a deep learning approach for tool detection via region proposals, which provides a bounding box and but not a precise localization of the landmarks. Instead of tracking the tool directly, two-step methods based on tool segmentation have also been proposed. Color, HOG and SIFT features were employed by Allan *et al.* [8] for pixel-wise classification of the image. The position was subsequently determined based on largest connected components. Instead of reducing the region of interest, Reiter *et al.* [9] employ the segmentation as a post-processing step for improving the localization accuracy. Recent segmentation methods [10] can be employed for these two-step approaches. However, the observation that segmentation can be used both for pre- and post-processing suggests that tracking of an instrument landmark and its segmentation are not only dependent, but indeed interdependent.

Our contributions are as follows. Instead of carrying out the tasks as two subsequent pipeline stages, we propose to perform tool segmentation and pose estimation simultaneously, in a unified deep learning approach (Fig. 1). To this end, we reformulate the pose estimation task and model the problem as a heatmap regression where every pixel represents a confidence proportional to its proximity to the correct landmark location. This modeling allows for representing semantic segmentation and localization with equal dimensionality, which leverages on their spatial dependency and facilitates simultaneous learning. It also enables employing state-of-the-art deep learning techniques, such as Fully Convolutional Residual Networks [11, 12]. The resulting model is trained jointly and end-to-end for both tasks, relying only on contextual information, thus being capable of reaching both objectives efficiently without requiring any post-processing technique. We compare the proposed method to state-of-the-art algorithms on a benchmark dataset of *in-vivo* RM sequences and on the EndoVis Challenge[1], on which we also outperform other popular CNN architectures, such as U-net [13] and FCN [10]. To the best of our knowledge, this is the first approach that em-

[1] MICCAI 2015 Endoscopic Vision Challenge Instrument Segmentation and Tracking Sub-challenge `http://endovissub-instrument.grand-challenge.org`

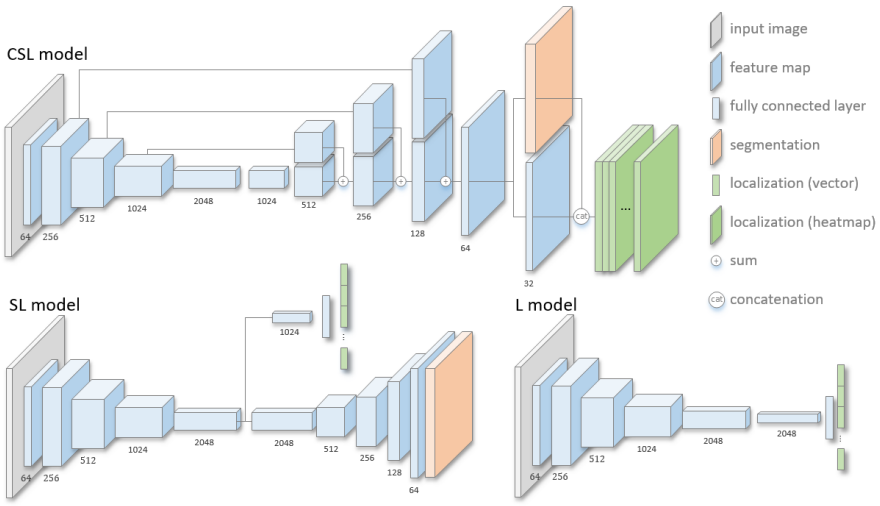


Fig. 2. Modeling Strategies: The proposed CSL architecture and two baselines.

ploys deep learning for surgical instrument tracking by predicting segmentation and localization simultaneously and is successful despite limited data.

2 Method

This section describes our CNN-based approach to model the mapping from an input image to the location of the tool landmarks and the corresponding dense semantic labeling. For this purpose, we motivate the use of a fully convolutional network, that models the problem of landmark localization as a regression of a set of heatmaps (one per landmark) in combination with semantic segmentation. This approach exploits global context to identify the position of the tool and has clear advantages comparing to patch-based techniques, which rely only on local information, thus being less robust towards false positives, e.g. reflections of the instrument. We compare the proposed architecture and discuss its advantage over two baselines. A common block for all discussed architectures is the encoder (Sec. 2.1), which progressively down-samples the input image through a series of convolutions and pooling operations. The differences lie in the subsequent *decoding* stages (Sec. 2.2) and the output formulation. An overview is depicted in Fig. 2. We denote a training sample as (X, S, y), where $y \in \mathbb{R}^{(n\times 2)}$ refers to the 2D coordinates of n tracked landmarks in the image $X \in \mathbb{R}^{w\times h\times 3}$, $S \in \mathbb{R}^{\frac{w}{2}\times\frac{h}{2}\times c}$ represents the semantic segmentation for c labels and w, h denote the image width and height respectively.

2.1 Encoder

For the encoding part of the three proposed models, we employ ResNet-50 [12], a state-of-the-art architecture that achieves top performance in several computer

vision tasks, such as classification and object detection. It is composed of successive *residual* blocks, each consisting of several convolutions and a shortcut (identity) connection summed to its output. In this way, it allows for a very deep architecture without hindering the learning process and at relatively low complexity. Although deeper versions of ResNet exist, we use the 50-layer variant, as computation time is still crucial for our problem. As input to the network, we consider images with $w = h = 480$ pixels. Thus, the feature maps at the last convolutional layer of ResNet have a resolution of 15×15 pixels. The last pooling layer and the loss layer are removed.

2.2 Decoder Tasks

We then define three different CNN variants, appended to the encoder, to find the best formulation for our task. In the following we outline the characteristics of each model and motivate the choice of the final proposed model.

Localization (L): First, we examine the naïve approach that regresses the real 2D locations of the landmarks directly. Here, the segmentation task is excluded. To further reduce the spatial dimensions of the last feature maps, we append another residual block with stride to the end of the encoder ($8 \times 8 \times 2048$). Similarly to the original architecture [12], this is followed by a 8×8 average pooling layer and a fully-connected layer which produces the output. This dimensionality reduction is needed so that the averaging is not applied over a large region, which would result in a greater loss of spatial information, thus affecting the precision with which the network is able to localize. In this case, the training sample is (X, y) and the predicted location is $\tilde{y} \in \mathbb{R}^{2\times n}$. The network is trained with a standard L_2 loss: $l_L(\tilde{y}, y) = ||\tilde{y} - y||_2^2$.

Segmentation and Localization (SL): In this model we regress the 2D locations and additionally predict the semantic segmentation map of an input within a single architecture. Both tasks share weights along the encoding part of the network and then split into two distinct parts to model their different dimensionality. For the regression of the landmark positions we follow the aforementioned model (L). For the semantic segmentation, we employ successive residual up-sampling layers as in [11], to predict the probability of each pixel belonging to a specified class, e.g. manipulator, shaft or background. Due to real-time constraints, we produce the network output with half of the input resolution and bilinearly up-sample the result. By sharing the encoder weights, the two tasks can influence each other while upholding their own objectives. Here, the training sample is (X, S, y), and the prediction consists of $\tilde{y} \in \mathbb{R}^{2\times n}$ and $\tilde{S} \in \mathbb{R}^{\frac{w}{2} \times \frac{h}{2} \times c}$. The network is trained by combining the losses for the separate tasks: $l_{SL}(\tilde{y}, y, \tilde{S}, S) = \lambda_L l_L(\tilde{y}, y) + l_S(\tilde{S}, S)$, where λ_L balances the influence of both loss terms. For the segmentation we employ a pixel-wise softmax-log loss:

$$l_S(\tilde{S}, S) = -\frac{1}{wh} \sum_{x=1}^{w} \sum_{y=1}^{h} \sum_{j=1}^{c} S(x, y, j) \log \left(\frac{e^{\tilde{S}(x,y,j)}}{\sum_{k=1}^{c} e^{\tilde{S}(x,y,k)}} \right) \tag{1}$$

Concurrent Segmentation and Localization (CSL): In both L and SL architectures, only a single 2D position is considered as the correct target for each landmark. However, manual annotations can differ in a range of several pixels, which in turn implies discrepancies or imprecise labeling. Predicting an absolute target location is arbitrary and ignores image context. Therefore, in the proposed model, we address this problem by regressing a *heatmap* for each tracked landmark instead of its exact coordinates. The heatmap represents the confidence of being close to the actual location of the tracked point and is created by applying a Gaussian kernel to its ground truth position. The heatmaps have the same size as the segmentation and can explicitly share weights over the entire network. We further enhance the architecture with long-range skip connections that *sum* lower-level feature maps from the encoding into the decoding stage, in addition to the residual connections of the up-sampling layers [11]. This allows higher resolution information from the initial layers to flow to the output layers without being compressed through the encoder, thus increasing the model's accuracy. Finally, we enforce a strong dependency of the two tasks by only separating them at the very end and concatenating the predicted segmentation scores (before softmax) to the last set of feature maps as an auxiliary means for guiding the location heatmaps. The overall loss is given by:

$$l_{CSL} = l_S(\tilde{S}, S) + \frac{\lambda_H}{n} \sum_{i=1}^{n} \sum_{x=1}^{w} \sum_{y=1}^{h} || \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{||y_i-(x,y)^T||_2^2}{2\sigma^2}} - \tilde{y}^*_{x,y,i}||_2^2 \qquad (2)$$

The standard deviation σ controls the spread of the Gaussian around the landmark location y_i. In testing, the point of maximum confidence in each predicted heatmap $\tilde{y}^*_i \in \mathbb{R}^{\frac{w}{2} \times \frac{h}{2} \times n}$ is used as the location of the instrument landmark. Notably, a misdetection is indicated by high variance in the predicted map.

3 Experiments and Results

In this section, we evaluate the performance of the proposed method in terms of localization of the instrument landmarks, as well as segmentation accuracy.

Datasets: The `Retinal Microsurgery` Dataset [3] consists of 18 *in-vivo* sequences, each with 200 frames of resolution 1920×1080 pixels. The set is further classified into four instrument-dependent subsets. The annotated tool joints are $n = 3$ and semantic classes $c = 2$ (tool and background). In the `EndoVis` Dataset, the training data contains four *ex-vivo* 45s sequences and the testing includes the rest 15s of the same sequences, plus two new 60s videos. All sequences have a resolution of 720×576 pixels and include one or two surgical instruments. There is $n = 1$ joint per tool and $c = 3$ semantic classes.

Implementation details: The encoder is initialized with ResNet-50 weights pretrained on ImageNet. All newly added layers are randomly initialized from a normal distribution with zero mean and 0.01 variance. All images are resized to 640×480 pixels and augmented during training with random rotations $[-5°, 5°]$, scaling $[1, 1.2]$, random crops of 480×480, gamma correction with $\gamma \in [0.9, 1.1]$,

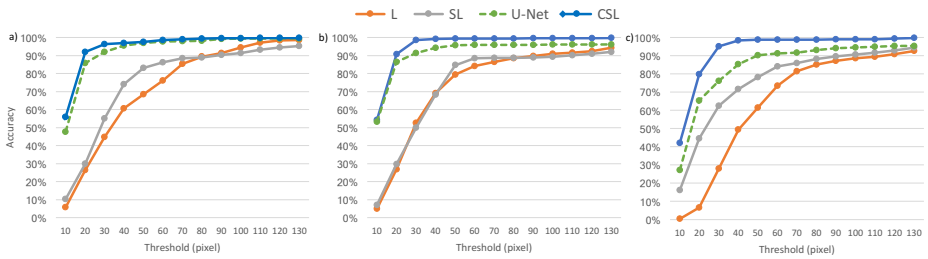


Fig. 3. Evaluation of Modeling Strategies: Accuracy of the models by means of Threshold Score for the left tip (a), right tip (b) and center joint (c) of the instrument.

a multiplicative color factor $c \in [0.8, 1.2]^3$ and specular reflections. For localization, we set $\sigma = 5$ for `RM` and $\sigma = 7$ for `EndoVis` in which the tools are larger. All CNNs are trained with stochastic gradient descent with learning rate 10^{-7}, momentum 0.9 and empirically chosen $\lambda_L, \lambda_H = 1$. The inference time is 56ms per frame on a NVIDIA GeForce GTX TITAN X using MatConvNet.

3.1 Evaluation of Modeling Strategies

First, we evaluate the models for tool landmark localization by training on 9 sequences of the RM dataset and testing on the remaining ones. In Fig. 3, the baseline of explicit 2D landmark localization (**L**) shows the lowest results, while its combination with a segmentation task (**SL**) increases the performance. The proposed **CSL** model achieves the highest accuracy of over 90% for both tool tips and 79% for the center joint (for thres. 20 pixels). Our model exploits contextual information for precise localization of the tool, by sharing feature maps with the semantic segmentation task. Another baseline is the U-Net architecture [13] trained with the same objectives. CSL is consistently more accurate in localization and achieves a DICE score of **75.4%**, while U-Net scores 72.5%, SL 73.7% and CSL without skip connections 74.4%.

3.2 Retinal Microsurgery

Analogously to [3], we train on all first halves of the 18 RM sequences and evaluate on the remaining frames, referred to as *Half Split*. As shown in Fig. 4, the proposed method clearly outperforms the state-the-art-methods, reaching an average accuracy of more than 84% considering the *KBB* score with $\alpha = 0.15$. Next, we evaluate the generalization ability of our method not only to unseen sequences and but also to unknown geometry. We employ a leave-one-out scheme on the subsets given by the 4 different instrument types, referred to as *Cross Validation*, and show that our method achieves state-of-the-art performance.

3.3 EndoVis Challenge

For this dataset, we performed our experiments in a leave-one-surgery-out fashion, as specified by the guidelines. We report our quantitative results in Ta-

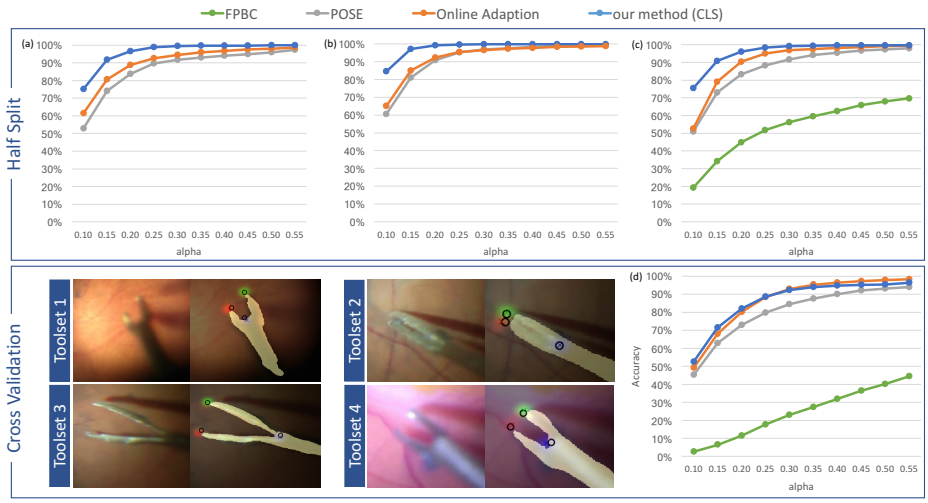


Fig. 4. RM dataset: Comparison to FPBC [14], POSE [3] and Online Adaption [6], measured by the metric *KBB*. The charts (a) to (c) show the accuracy for the left tip, right tip and center joint, respectively, for the *Half Split* experiment. In the *Cross Validation*, the training set is given by 3 instrument dependent subsets and the method is tested on the remaining set. (d) shows the average *KBB* score for the center point.

ble 1, both binary and multi-class, and compare to the previous state-of-the-art, which we significantly outperform. Notably, the proposed method can also provide multi-segmentation for the separate tools (Fig. 5) if trained with $c = 5$. A challenging aspect of this dataset is that two instruments can be present in the testing set, while only one is included in the training. To alleviate this problem, we additionally augment with horizontal flips, such that the instrument is at least seen from both sides. In Sets 5 and 6, the network was capable of successfully localizing and segmenting a previously unseen instrument and viewpoint[2].

	Binary				Shaft		Grasper		Joint
Sequence	B.Acc.	Rec.	Spec.	DICE	Rec.	Spec.	Rec.	Spec.	loc. error
1	91.9	85.0	98.7	88.5	79.2	99.1	76.2	98.7	39.0/30.8
2	94.8	90.0	99.7	93.0	90.9	99.8	82.0	99.8	9.7
3	94.7	90.1	99.3	91.6	89.1	99.5	86.8	99.7	10.9
4	91.1	83.1	99.0	85.8	82.9	99.2	65.4	99.6	13.0
5	91.5	84.2	98.8	87.3	82.8	99.1	75.9	99.2	38.4/60.0
6	91.7	84.9	99.0	88.9	78.0	99.3	78.1	98.4	36.4/63.9
CSL (mean)	**92.6**	86.2	**99.0**	88.9	83.8	99.3	77.4	99.2	24.8/51.6
FCN [10]	83.7	72.2	95.2	-	-	-	-	-	-
FCN+OF [10]	88.3	**87.8**	88.7	-	-	-	-	-	-

Balanced Accuracy (*B.Acc.*), Recall (*Rec.*), Specificity (*Spec.*) and DICE are in %. The average localization error (*loc. error*) is in pixel.

Table 1. *Cross-Validation* results for `EndoVis`.

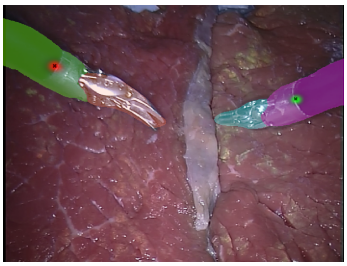


Fig. 5. Qualitative Result.

[2] The challenge administrators believe that the ground truth regarding tracking for sequence 5 and 6 is not as accurate as for the rest of the sequences.

4 Conclusion

In this paper, we propose to model the localization of instrument landmarks as a heatmap regression. This allows us to leverage deep-learned features via a CNN to concurrently regress the instrument segmentation and its articulated 2D pose in an end-to-end manner. We evaluate the performance on two different benchmarks and, throughout the experiments our approach outperforms state-of-the-art methods.

References

1. Bouget, D., Allan, M., Stoyanov, D., Jannin, P.: Vision-based and marker-less surgical tool detection and tracking: a review of the literature. Medical Image Analysis **35** (2017)
2. Sznitman, R., Richa, R., Taylor, R.H., Jedynak, B., Hager, G.D.: Unified detection and tracking of instruments during retinal microsurgery. IEEE trans. on Pattern Analysis and Machine Intelligence **35**(5) (2013)
3. Rieke, N., Tan, D.J., Amat di San Filippo, C., Tombari, F., Alsheakhali, M., Belagiannis, V., Eslami, A., Navab, N.: Real-time localization of articulated surgical instruments in retinal microsurgery. Medical Image Analysis **34** (2016)
4. Bouget, D., Benenson, R., Omran, M., Riffaud, L., Schiele, B., Jannin, P.: Detecting surgical tools by modelling local appearance and global shape. Trans. on Medical Imaging **34**(12) (2015)
5. Zhou, J., Payandeh, S.: Visual tracking of laparoscopic instruments. J. Autom. Cont. Eng. Vol. **2**(3) (2014) 234–241
6. Rieke, N., Tan, D.J., Tombari, F., Vizcaíno, J.P., di San Filippo, C.A., Eslami, A., Navab, N.: Real-time online adaption for robust instrument tracking and pose estimation. In: MICCAI, Springer (2016) 422–430
7. Sarikaya, D., Corso, J., Guru, K.: Detection and localization of robotic tools in robot-assisted surgery videos using deep neural networks for region proposal and detection. IEEE Transactions on Medical Imaging (2017)
8. Allan, M., Ourselin, S., Thompson, S., Hawkes, D.J., Kelly, J., Stoyanov, D.: Toward detection and localization of instruments in minimally invasive surgery. In: IEEE Transactions on Biomedical Engineering 60, pp. 1050 – 1058 (2013)
9. Reiter, A., Allen, P.K., Zhao, T.: Marker-less articulated surgical tool detection. In: Proc. Computer assisted radiology and surgery. Volume 7. (2012) 175–176
10. Garcia Peraza Herrera, L., Li, W., Gruijthuijsen, C., Devreker, A., Attilakos, G., Deprest, J., Vander Poorten, E., Stoyanov, D., Vercauteren, T., Ourselin, S.: Real-time segmentation of non-rigid surgical tools based on deep learning and tracking. In: CARE workshop at MICCAI, Springer (2016)
11. Laina, I., Rupprecht, C., Belagiannis, V., Tombari, F., Navab, N.: Deeper depth prediction with fully convolutional residual networks. In: Int. Conf. on 3D Vision (3DV), IEEE (2016) 239–248
12. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2016) 770–778
13. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: MICCAI, Springer (2015) 234–241
14. Sznitman, R., Becker, C., Fua, P.: Fast part-based classification for instrument detection in minimally invasive surgery. In: MICCAI, Springer (2014)

B

Abstracts of Publications not Discussed in this Dissertation

CFCM: Segmentation via Coarse to Fine Context Memory [129]

Fausto Milletari, Nicola Rieke, Maximilian Baust, Marco Esposito and Nassir Navab

Abstract. Recent neural-network-based architectures for image segmentation make extensive usage of feature forwarding mechanisms to integrate information from multiple scales. Although yielding good results, even deeper architectures and alternative methods for feature fusion at different resolutions have been scarcely investigated for medical applications. In this work we propose to implement segmentation via an encoder- decoder architecture which differs from any other previously published method since (i) it employs a very deep architecture based on residual learning and (ii) combines features via a convolutional Long Short Term Memory (LSTM), instead of concatenation or summation. The intuition is that the memory mechanism implemented by LSTMs can better integrate features from different scales through a coarse-to-fine strategy; hence the name Coarse-to-Fine Context Memory (CFCM). We demonstrate the remarkable advantages of this approach on two datasets: the Montgomery county lung segmentation dataset, and the EndoVis 2015 challenge dataset for surgical instrument segmentation.

International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI 2018), 2018, Granada, Spain.

Fast 5DOF Needle Tracking in iOCT [1]

Jakob Weiss, Nicola Rieke, Ali M. Nasseri, Mathias Maier, Chris Lohmann, Abouzar Eslami and Nassir Navab

Purpose. Intraoperative Optical Coherence Tomography (iOCT) is an increasingly available imaging technique for ophthalmic microsurgery that provides high-resolution cross-sectional information of the surgical scene. We propose to build on its desirable qualities and present a method for tracking the orientation and location of a surgical needle. Thereby, we enable direct analysis of instrument-tissue interaction directly in OCT space without complex multimodal calibration that would be required with traditional instrument tracking methods.

Method. The intersection of the needle with the iOCT scan is detected by a peculiar multi-step ellipse fitting that takes advantage of the directionality of the modality. The geometric modelling allows us to use the ellipse parameters and provide them into a latency aware estimator to infer the 5DOF pose during needle movement.

Results. Experiments on phantom data and ex-vivo porcine eyes indicate that the algorithm retains angular precision especially during lateral needle movement and provides a more robust and consistent estimation than baseline methods.

Conclusion. Using solely cross-sectional iOCT information, we are able to successfully and robustly estimate a 5DOF pose of the instrument in less than 5.4 ms on a CPU.

International Conference on Information Processing in Computer-Assisted Interventions (IPCAI 2018), 2018, Berlin, Germany.

Injection Assistance via Surgical Needle Guidance using Microscope-Integrated OCT (MI-OCT) [2]

Jakob Weiss, Nicola Rieke, Ali M. Nasseri, Mathias Maier, Chris Lohmann, Nassir Navab and Abouzar Eslami

Purpose. Injection in ophthalmic interventions requires precise targeting of anatomic layers both in anterior and posterior segment surgery. In the current workflow surgeons mainly rely on the binocular en-face view of the microscope which provides only a limited perception of depth and distance to target. Our work aims at assisting the surgeon by providing the projected intersection point of instrument and anatomy and thereby enabling intuitive and more precise targeting.

Method. The proposed method uses cross-sectional information provided by MI-OCT. The surgical needle and the position of the anatomical target is localized in five continuously acquired parallel MI-OCT B-Scans. To that end, the maximum intensity along each A-Scan is extracted for each OCT image. Given this point set, the target layer and the instrument cross-section is found by a geometry fitting algorithm. In the second step, we find the 3D needle pose based on these segmented cross-sections by using the specific geometry of the surgical needle and a temporal filter. Finally, the estimated injection position is determined by the geometric intersection of the approximated target layer with the needle. By projecting the computed intersection point to the en-face view, we can overlay the estimated position in real-time via a heads-up display on the microscopic view. As a result, the surgeons can infer the distance to the target layer by the distance between needle tip and projected injection point in an intuitive way.

Results. We evaluated our method on both anterior and posterior phantoms. In both scenarios, we acquired three sequences of free needle movement. For acquisition, we used a Zeiss Lumera 700 with Resight 700 in 5-line HD OCT mode and the needle is constantly touching

the target surface to have ground truth. In the en-face camera view, we manually annotate the ground truth touching/intersection point in each image as the needle tip in 811 images. We report a median error of 0.230mm (mean: 0.299+-0.062mm) for anterior and median error of 0.268mm (mean: 0.358+- 0.090mm) for posterior guidance.

Conclusion. We present an approach to provide continuous injection guidance based on conventional MI-OCT B- Scans. The system provides accurate prediction and visualization of the injection point, thus supporting the surgeon in difficult injection tasks.

Proceedings of the Association for Research in Vision and Ophthalmology Annual Meeting (ARVO 2018), 2018, Honolulu, USA.

Automatic Initialization and Failure Detection for Surgical Tool Tracking in Retinal Microsurgery [4]

Josué Page Vizcaíno, Nicola Rieke, David Joseph Tan, Federico Tombari, Abouzar Eslami, Nassir Navab

Instrument tracking is a key step for various computer-aided interventions in retinal microsurgery. One of the bottlenecks of state-of-the-art template based algorithms is the (re-)initialization during the surgery. We propose an algorithm for robustly detecting the bounding box around the tool tip together with a failure detection of the tracking algorithm. Hereby, the user input dependent algorithm is transformed into a completely automatic framework without the need of an assistant. The performance was compared to two state-of-the-art methods.

Proceedings of Workshop Bildverarbeitung für die Medizin (BVM 2017), 2017, Heidelberg, Germany.

Automatic iOCT Positioning During Membrane Peeling via Real-time High Resolution Surgical Forceps Tracking [8]

Nicola Rieke, Stefan Duca, Abouzar Eslami and Nassir Navab

Purpose. During membrane peeling, the intraoperative OCT (iOCT) has to be positioned manually, which may be time consuming and distracting from the actual procedure. By automatically repositioning according to the position of forceps tips, this step can be avoided. Compared to rigid instruments such as diamond dusted tools, the robust tracking of the forceps is considerably more challenging due to partial occlusions and pose variations, i.e. opening and closing movement. We introduce a novel method for automatic repositioning of the iOCT

by analyzing the surgeon's maneuver based on a real-time, precise tracking of the tool tips in the co-registered microscopic view. The developed algorithm is robust to different shapes of the available forceps in the market as well as the field of view and microscope zoom.

Method. The algorithm is a two-step process: first, the tracker infers the location of the tips and the joint point of the forceps in real-time from local image features in the microscopic view. The resulting angle spanned by the two forceps tips in stored in a buffer of several frames. In the second step, a linear regression is performed on this time-series for detecting a grasping movement and the current forceps tip locations are used to reposition the OCT scan location in a microscope-integrated OCT system. The SD-OCT scanner is calibrated with the microscopic view which is fed to our algorithm through a high definition camera. Machine learning is employed to make the algorithm robust to lighting changes induced by the intraocular light.

Results. We created a dataset of 2400 annotated frames from 12 different recorded in-vivo surgery sequences showing at least one grasping movement. For evaluation, the dataset was equally divided in a training and a testing set. The median Euclidean localization error of the tool tips is 14.45 pixel for the left tip, 124.2 pixel for the right tip and 11.84 pixel for the center joint considering a frame of 1950×1010 pixel. The grasping movement was detected correctly in $92,85\%$ of the cases. *Conclusion*. We presented a novel method for repositioning the iOCT during peeling based on a robust instrument tracking which can handle the large variations in illumination, instrument appearance and shape.

Proceedings of International Society of Imaging in the Eye Conference, Association for Research in Vision and Ophthalmology (ARVO 2016), 2016, Seattle, USA.

Image Descriptors in Angiography [9]

Katharina Hofschen, Timo Geissler, Nicola Rieke, Christian Schulte zu Berge, Nassir Navab and Stefanie Demirci

Abstract. Despite recent advances in the field of image-guided interventions (IGI), the bottleneck for Angiography/X-ray guided procedures in particular is accurate and robust 2D-3D image alignment. The conventional, straight-forward parameter optimisation approach is known to be ill-posed and less efficient. Retrieval-based approaches may be of superior choice here. However, this requires salient and robust image features, which can handle the difficulties of Angiographic images such as high level of noise and contrast variance. In this paper, we investigate state-of-the-art features of the field of Computer Vision regarding the applicability and reliability in the challenging scenario of Angiography.

Proceedings on Workshop Bildverarbeitung für die Medizin (BVM 2016), 2016, Berlin, Germany.

Ultrasound Interactive Segmentation with Tensor-Graph Methods [11]

Nicola Rieke, Christoph Hennersperger, Diana Mateus and Nassir Navab

Abstract. We address the problem of segmenting aortic aneurysms in ultrasound images. As solution we propose a novel frame- work based on graph-based interactive segmentation methods, such as graph-cuts and random walks. Our main contribution is extending these approaches to handle structure tensor ultrasound images. Our hypothesis is that the structure tensor is better suited to represent the contextual information in ultrasound images than the pure b-mode intensity values. We demonstrate that this extension significantly improves the performance of both methods in clinical data.

IEEE International Symposium on Biomedical Imaging (ISBI 2014), 2014, Beijing, China.

List of Figures

List of Tables

Bibliography

[1] J. Weiss, N. Rieke, M. A. Nasseri, M. Maier, A. Eslami, and N. Navab. “Fast 5DOF needle tracking in iOCT”. In: *International Journal of Computer Assisted Radiology and Surgery* 13.6 (2018), pp. 787–796 (cit. on pp. 1, 7, 103).

[2] J. Weiss, N. Rieke, M. A. Nasseri, M. Maier, C. P. Lohmann, N. Navab, and A. Eslami. “Injection Assistance via Surgical Needle Guidance using Microscope-Integrated OCT (MI-OCT)”. In: vol. 59. 9. The Association for Research in Vision and Ophthalmology, 2018, pp. 287–287 (cit. on pp. 1, 7, 104).

[3] I. Laina*, N. Rieke*, C. Rupprecht, J. Page Vizcaíno, A. Eslami, F. Tombari, and N. Navab. “Concurrent Segmentation and Localization for Tracking of Surgical Instruments”. In: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2017: 20th International Conference, Quebec City, QC, Canada, September 11-13, 2017, Proceedings, Part II*. Ed. by M. Descoteaux, L. Maier-Hein, A. Franz, P. Jannin, D. L. Collins, and S. Duchesne. The first two authors contributed equally to this paper. Springer International Publishing. 2017, pp. 664–672 (cit. on pp. 1, 7, 37, 42).

[4] J. P. Vizcaíno, N. Rieke, D. J. Tan, F. Tombari, A. Eslami, and N. Navab. “Automatic Initialization and Failure Detection for Surgical Tool Tracking in Retinal Microsurgery”. In: *Bildverarbeitung für die Medizin 2017: Algorithmen-Systeme-Anwendungen. Proceedings des Workshops vom 12. bis 14. März 2017 in Heidelberg*. Ed. by K. H. Maier-Hein geb. Fritzsche, T. M. Deserno geb. Lehmann, H. Handels, and T. Tolxdorff. Springer, 2017, pp. 346–351 (cit. on pp. 1, 7, 48, 105).

[5] N. Rieke, D. J. Tan, F. Tombari, J. P. Vizcaíno, C. Amat di San Filippo, A. Eslami, and N. Navab. “Abstract: Real-Time Online Adaption for Robust Instrument Tracking and Pose Estimation”. In: *Bildverarbeitung für die Medizin 2017*. Ed. by K. H. Maier-Hein geb. Fritzsche, T. M. Deserno geb. Lehmann, H. Handels, and T. Tolxdorff. Berlin, Heidelberg: Springer Berlin Heidelberg, 2017, pp. 337–338 (cit. on pp. 1, 7).

[6] N. Rieke, D. J. Tan, F. Tombari, J. Page Vizcaíno, C. Amat di San Filippo, A. Eslami, and N. Navab. “Real-Time Online Adaption for Robust Instrument Tracking and Pose Estimation”. In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Ed. by S. Ourselin, L. Joskowicz, M. R. Sabuncu, G. Unal, and W. Wells. Springer International Publishing. 2016, pp. 422–430 (cit. on pp. 2, 7, 37, 40).

[7] N. Rieke, D. J. Tan, C. Amat di San Filippo, F. Tombari, M. Alsheakhali, V. Belagiannis, A. Eslami, and N. Navab. “Real-time localization of articulated surgical instruments in retinal microsurgery”. In: *Medical Image Analysis* 34 (2016). Special Issue on the 2015 Conference on Medical Image Computing and Computer Assisted Intervention, pp. 82 –100 (cit. on pp. 2, 7, 28, 37, 39).

[8] N. Rieke, S. Duca, N. Navab, and A. Eslami. “Automatic iOCT Positioning During Membrane Peeling via Real-time High Resolution Surgical Forceps Tracking”. In: *International Society of Imaging in the Eye Conference, Association for Research in Vision and Ophthalmology (ARVO 2016)*. Seattle, USA, 2016 (cit. on pp. 2, 7, 18, 47, 105).

[9] K. Hofschen, T. Geissler, N. Rieke, C. S. z. Berge, N. Navab, and S. Demirci. “Image Descriptors in Angiography”. In: *Bildverarbeitung für die Medizin 2016*. Ed. by T. Tolxdorff, T. M. Deserno, H. Handels, and H.-P. Meinzer. Berlin, Heidelberg: Springer Berlin Heidelberg, 2016, pp. 283–288 (cit. on pp. 2, 106).

[10] N. Rieke, D. J. Tan, M. Alsheakhali, F. Tombari, C. Amat di San Filippo, V. Belagiannis, A. Eslami, and N. Navab. “Surgical Tool Tracking and Pose Estimation in Retinal Microsurgery”. In: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*. Ed. by N. Navab, J. Hornegger, W. M. Wells, and A. Frangi. Cham: Springer International Publishing, 2015, pp. 266–273 (cit. on pp. 2, 7, 37–41).

[11] N. Rieke, C. Hennersperger, D. Mateus, and N. Navab. “Ultrasound interactive segmentation with tensor-graph methods”. In: *2014 IEEE 11th International Symposium on Biomedical Imaging (ISBI)*. IEEE. 2014, pp. 690–693 (cit. on pp. 2, 107).

[12] K. Cleary and T. M. Peters. “Image-guided interventions: technology review and clinical applications”. In: *Annual review of biomedical engineering* 12 (2010), pp. 119–142 (cit. on p. 5).

[13] L. Maier-Hein, S. Vedula, S. Speidel, N. Navab, R. Kikinis, A. Park, M. Eisenmann, H. Feussner, G. Forestier, S. Giannarou, et al. “Surgical data science: enabling next-generation surgery”. In: *arXiv preprint arXiv:1701.06482* (2017) (cit. on pp. 5, 49, 50).

[14] J. I. Barraquer. “The history of the microscope in ocular surgery”. In: *Microsurgery* 1.4 (1980), pp. 288–299 (cit. on p. 5).

[15] M. G. Yasargil. *Microsurgery: applied to neurosurgery*. Thieme, 2011 (cit. on p. 5).

[16] K. Uluç, G. C. Kujoth, and M. K. Başkaya. “Operating microscopes: past, present, and future”. In: *Neurosurgical focus* 27.3 (2009), E4 (cit. on p. 5).

[17] A. G. Harrell and B. T. Heniford. “Minimally invasive abdominal surgery: lux et veritas past, present, and future”. In: *The American journal of surgery* 190.2 (2005), pp. 239–243 (cit. on p. 5).

[18] J. Ponsky. “Endoluminal surgery: past, present and future”. In: *Surgical Endoscopy And Other Interventional Techniques* 20.2 (2006), S500–S502 (cit. on p. 5).

[19] Z.-H. Cho. *Foundations of medical imaging*. Wiley-Interscience, 1993 (cit. on p. 5).

[20] B. J. Pichler, M. S. Judenhofer, and C. Pfannenberg. “Multimodal Imaging Approaches: PET/CT and PET/MRI”. In: *Molecular Imaging I*. Springer, 2008, pp. 109–132 (cit. on p. 5).

[21] G. I. Barbash and S. A. Glied. “New technology and health care costs—the case of robot-assisted surgery”. In: *New England Journal of Medicine* 363.8 (2010), pp. 701–704 (cit. on p. 5).

[22] A. R. Lanfranco, A. E. Castellanos, J. P. Desai, and W. C. Meyers. “Robotic surgery: a current perspective”. In: *Annals of surgery* 239.1 (2004), p. 14 (cit. on p. 5).

[23] G. Giguère and B. C. Love. “Limits in decision making arise from limits in memory retrieval”. In: *Proceedings of the National Academy of Sciences* 110.19 (2013), pp. 7613–7618 (cit. on p. 5).

[24] A. Bartoli, T. Collins, N. Bourdel, and M. Canis. “Computer assisted Minimally Invasive Surgery: Is medical Computer Vision the answer to improving laparosurgery?” In: *Medical hypotheses* 79.6 (2012), pp. 858–863 (cit. on pp. 5, 15, 16).

[25] L. Maier-Hein, S. S. Vedula, S. Speidel, et al. “Surgical data science for next-generation interventions”. In: *Nature Biomedical Engineering* 1.9 (2017), p. 691 (cit. on pp. 5, 6).

[26] K. Cleary, H. Y. Chung, and S. K. Mun. “OR2020 workshop overview: operating room of the future”. In: *International Congress Series*. Vol. 1268. Elsevier. 2004, pp. 847–852 (cit. on p. 6).

[27] C. Loukas. “Video content analysis of surgical procedures”. In: *Surgical endoscopy* (2017), pp. 1–16 (cit. on p. 10).

[28] G. J. Johnson, D. C. Minassian, R. A. Weale, S. K. West, et al. *The epidemiology of eye disease.* Ed. 2. Arnold, 2003 (cit. on p. 10).

[29] V Narendran, A. Kothari, S. Charles, V. Saravanan, and I. Kreissig. *Principles and Practice of Vitreoretinal Surgery*. JP Medical Ltd, 2014 (cit. on p. 11).

[30] G. A. Peyman, S. A. Meffert, F. Chou, and M. D. Conway. *Vitreoretinal surgical techniques*. CRC Press, 2000 (cit. on p. 11).

[31] K. Z. Aung, G. Makeyeva, M. K. Adams, E. W.-T. Chong, L. Busija, G. G. Giles, D. R. English, J. Hopper, P. N. Baird, R. H. Guymer, et al. "The prevalence and risk factors of epiretinal membranes: the Melbourne Collaborative Cohort Study". In: *Retina* 33.5 (2013), pp. 1026–1034 (cit. on p. 11).

[32] C. H. Ng, N. Cheung, J. J. Wang, A. F. Islam, R. Kawasaki, S. M. Meuer, M. F. Cotch, B. E. Klein, R. Klein, and T. Y. Wong. "Prevalence and risk factors for epiretinal membranes in a multi-ethnic United States population". In: *Ophthalmology* 118.4 (2011), pp. 694–699 (cit. on p. 11).

[33] M. Miyazaki, H. Nakamura, M. Kubo, Y. Kiyohara, M. Iida, T. Ishibashi, and Y. Nose. "Prevalence and risk factors for epiretinal membranes in a Japanese population: the Hisayama study". In: *Graefe's archive for clinical and experimental ophthalmology* 241.8 (2003), pp. 642–646 (cit. on p. 11).

[34] G. L. Spaeth, H. Danesh-Meyer, I. Goldberg, and A. Kampik. *Ophthalmic Surgery: Principles and Practice E-Book*. Elsevier Health Sciences, 2011 (cit. on p. 12).

[35] D. W. Park, P. U. Dugel, J. Garda, J. O. Sipperley, A. Thach, S. R. Sneed, and J. Blaisdell. "Macular pucker removal with and without internal limiting membrane peeling: pilot study". In: *Ophthalmology* 110.1 (2003), pp. 62–64 (cit. on p. 12).

[36] U. C. Christensen. "Value of internal limiting membrane peeling in surgery for idiopathic macular hole and the correlation between function and retinal morphology". In: *Acta ophthalmologica* 87.thesis2 (2009), pp. 1–23 (cit. on p. 12).

[37] R. Gelman, W. Stevenson, C. Prospero Ponce, D. Agarwal, and J. B. Christoforidis. "Retinal damage induced by internal limiting membrane removal". In: *Journal of ophthalmology* 2015 (2015) (cit. on p. 12).

[38] W. Drexler and J. G. Fujimoto. "State-of-the-art retinal optical coherence tomography". In: *Progress in retinal and eye research* 27.1 (2008), pp. 45–88 (cit. on p. 12).

[39] J. G. Fujimoto and W. Drexler. "Introduction to OCT". In: *Optical Coherence Tomography: Technology and Applications* (2015), pp. 3–64 (cit. on p. 12).

[40] J. P. Ehlers, P. K. Kaiser, and S. K. Srivastava. "Intraoperative optical coherence tomography using the RESCAN 700: preliminary results from the DISCOVER study". In: *British journal of Ophthalmology* (2014), bjophthalmol–2014 (cit. on p. 12).

[41] J. Ehlers. "Intraoperative optical coherence tomography: past, present, and future". In: *Eye* 30.2 (2016), pp. 193–201 (cit. on p. 12).

[42] J. P. Ehlers, M. Khan, D. Petkovsek, L. Stiegel, P. K. Kaiser, R. P. Singh, J. L. Reese, and S. K. Srivastava. "Outcomes of Intraoperative OCT–Assisted Epiretinal Membrane Surgery from the PIONEER Study". In: *Ophthalmology Retina* (2017) (cit. on p. 12).

[43] J. P. Ehlers, Y. S. Modi, P. E. Pecen, J. Goshe, W. J. Dupps, A. Rachitskaya, S. Sharma, A. Yuan, R. Singh, P. K. Kaiser, et al. "The DISCOVER Study 3-Year Results: Feasibility and Usefulness of Microscope-Integrated Intraoperative OCT during Ophthalmic Surgery". In: *Ophthalmology* () (cit. on p. 12).

[44] J. P. Ehlers, S. K. Srivastava, D. Feiler, A. I. Noonan, A. M. Rollins, and Y. K. Tao. "Integrative advances for OCT-guided ophthalmic surgery and intraoperative OCT: microscope integration, surgical instrumentation, and heads-up display surgeon feedback". In: *PloS one* 9.8 (2014), e105224 (cit. on p. 13).

[45] J. P. Ehlers, A. Uchida, and S. K. Srivastava. “Intraoperative optical coherence tomography-compatible surgical instruments for real-time image-guided ophthalmic surgery”. In: *British Journal of Ophthalmology* 101.10 (2017), pp. 1306–1308 (cit. on pp. 13, 18).

[46] P. K. Gupta, P. S. Jensen, and E. de Juan. “Surgical Forces and Tactile Perception During Retinal Microsurgery”. In: *Medical Image Computing and Computer-Assisted Intervention – MICCAI’99*. Ed. by C. Taylor and A. Colchester. Berlin, Heidelberg: Springer Berlin Heidelberg, 1999, pp. 1218–1225 (cit. on p. 13).

[47] R. Sznitman, S. Billings, D. Rother, D. Mirota, Y. Yang, J. Handa, P. Gehlbach, J. U. Kang, G. D. Hager, and R. Taylor. “Active multispectral illumination and image fusion for retinal microsurgery”. In: *Information Processing in Computer-Assisted Interventions*. Springer, 2010, pp. 12–22 (cit. on p. 13).

[48] M. J. Mack. “Minimally invasive and robotic surgery”. In: *Jama* 285.5 (2001), pp. 568–572 (cit. on p. 13).

[49] A. Polychronidis, P. Laftsidis, A. Bounovas, and C. Simopoulos. “Twenty years of laparoscopic cholecystectomy: Philippe Mouret—March 17, 1987”. In: *JSLS: Journal of the Society of Laparoendoscopic Surgeons* 12.1 (2008), p. 109 (cit. on p. 13).

[50] B Makhoul, A De La Taille, D Vordos, L Salomon, P Sebe, J. Audet, L Ruiz, A Hoznek, P Antiphon, A Cicco, et al. “Laparoscopic radical nephrectomy for T1 renal cancer: the gold standard? A comparison of laparoscopic vs open nephrectomy”. In: *BJU international* 93.1 (2004), pp. 67–70 (cit. on p. 13).

[51] T. Baron. “Natural orifice transluminal endoscopic surgery”. In: *British journal of surgery* 94.1 (2007), pp. 1–2 (cit. on p. 14).

[52] J. R. Romanelli and D. B. Earle. “Single-port laparoscopic surgery: an overview”. In: *Surgical endoscopy* 23.7 (2009), pp. 1419–1427 (cit. on p. 14).

[53] S. Apostolopoulos, S. De Zanet, C. Ciller, S. Wolf, and R. Sznitman. “Pathological OCT Retinal Layer Segmentation Using Branch Residual U-Shape Networks”. In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Ed. by M. Descoteaux, L. Maier-Hein, A. Franz, P. Jannin, D. L. Collins, and S. Duchesne. Cham: Springer International Publishing, 2017, pp. 294–301 (cit. on p. 15).

[54] A. G. Roy, S. Conjeti, S. P. K. Karri, D. Sheet, A. Katouzian, C. Wachinger, and N. Navab. “ReLayNet: retinal layer and fluid segmentation of macular optical coherence tomography using fully convolutional networks”. In: *Biomedical optics express* 8.8 (2017), pp. 3627–3642 (cit. on p. 15).

[55] M. Pavlidis, I. Georgalas, and N. Körber. “Determination of a new parameter, elevated epiretinal membrane, by en face oct as a prognostic factor for pars plana vitrectomy and safer epiretinal membrane peeling”. In: *Journal of ophthalmology* 2015 (2015) (cit. on p. 15).

[56] I. N. Fleming, S. Voros, B. Vagvolgyi, Z. Pezzementi, J. Handa, R. Taylor, and G. D. Hager. “Intraoperative visualization of anatomical targets in retinal surgery”. In: *Applications of Computer Vision, 2008. WACV 2008. IEEE Workshop on*. IEEE. 2008, pp. 1–6 (cit. on p. 16).

[57] S. Nicolau, L. Soler, D. Mutter, and J. Marescaux. “Augmented reality in laparoscopic surgical oncology”. In: *Surgical oncology* 20.3 (2011), pp. 189–201 (cit. on p. 16).

[58] M. Hu, G. Penney, D. Rueckert, P. Edwards, F. Bello, M. Figl, R. Casula, Y. Cen, J. Liu, Z. Miao, et al. “A robust mosaicing method with super-resolution for optical medical images”. In: *International Workshop on Medical Imaging and Virtual Reality*. Springer. 2010, pp. 373–382 (cit. on p. 16).

[59] S. De Zanet, T. Rudolph, R. Richa, C. Tappeiner, and R. Sznitman. “Retinal slit lamp video mosaicking”. In: *International journal of computer assisted radiology and surgery* 11.6 (2016), pp. 1035–1041 (cit. on p. 16).

[60] R. Sznitman, D. Rother, J. Handa, P. Gehlbach, G. D. Hager, and R. Taylor. “Adaptive Multispectral Illumination for Retinal Microsurgery”. In: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2010*. Ed. by T. Jiang, N. Navab, J. P. W. Pluim, and M. A. Viergever. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 465–472 (cit. on p. 16).

[61] D. Stoyanov and G. Z. Yang. “Removing specular reflection components for robotic assisted laparoscopic surgery”. In: *IEEE International Conference on Image Processing, 2005. ICIP 2005.* Vol. 3. IEEE. 2005, pp. III–632 (cit. on p. 16).

[62] S. Bernhardt, S. A. Nicolau, L. Soler, and C. Doignon. “The status of augmented reality in laparoscopic surgery as of 2016”. In: *Medical image analysis* 37 (2017), pp. 66–90 (cit. on pp. 16, 18).

[63] R. E. MacLaren, J. Bennett, and S. D. Schwartz. “Gene therapy and stem cell transplantation in retinal disease: the new frontier”. In: *Ophthalmology* 123.10 (2016), S98–S106 (cit. on p. 17).

[64] C. Lea, A. Reiter, R. Vidal, and G. D. Hager. “Segmental spatiotemporal cnns for fine-grained action segmentation”. In: *European Conference on Computer Vision*. Springer. 2016, pp. 36–52 (cit. on p. 17).

[65] F. Lalys, D. Bouget, L. Riffaud, and P. Jannin. “Automatic knowledge-based recognition of low-level tasks in ophthalmological procedures”. In: *International journal of computer assisted radiology and surgery* 8.1 (2013), pp. 39–49 (cit. on p. 17).

[66] O. Dergachyova, D. Bouget, A. Huaulmé, X. Morandi, and P. Jannin. “Automatic data-driven real-time segmentation and recognition of surgical workflow”. In: *International journal of computer assisted radiology and surgery* 11.6 (2016), pp. 1081–1089 (cit. on p. 17).

[67] A. P. Twinanda, S. Shehata, D. Mutter, J. Marescaux, M. de Mathelin, and N. Padoy. “Endonet: A deep architecture for recognition tasks on laparoscopic videos”. In: *IEEE transactions on medical imaging* 36.1 (2017), pp. 86–97 (cit. on p. 17).

[68] N. Padoy, T. Blum, S.-A. Ahmadi, H. Feussner, M.-O. Berger, and N. Navab. “Statistical modeling and recognition of surgical workflow”. In: *Medical image analysis* 16.3 (2012), pp. 632–641 (cit. on p. 17).

[69] O. Weede, F. Dittrich, H. Wörn, B. Jensen, A. Knoll, D. Wilhelm, M. Kranzfelder, A. Schneider, and H. Feussner. “Workflow analysis and surgical phase recognition in minimally invasive surgery”. In: *IEEE International Conference on Robotics and Biomimetics (ROBIO), 2012*. IEEE. 2012, pp. 1080–1074 (cit. on p. 17).

[70] R. Stauder, A. Okur, L. Peter, A. Schneider, M. Kranzfelder, H. Feussner, and N. Navab. “Random forests for phase detection in surgical workflow analysis”. In: *International Conference on Information Processing in Computer-Assisted Interventions*. Springer. 2014, pp. 148–157 (cit. on p. 17).

[71] C. E. Reiley, H. C. Lin, D. D. Yuh, and G. D. Hager. “Review of methods for objective surgical skill evaluation”. In: *Surgical endoscopy* 25.2 (2011), pp. 356–366 (cit. on pp. 17, 18).

[72] S. Speidel, M. Delles, C. Gutt, and R. Dillmann. “Tracking of Instruments in Minimally Invasive Surgery for Surgical Skill Analysis”. In: *Medical Imaging and Augmented Reality*. Ed. by G.-Z. Yang, T. Jiang, D. Shen, L. Gu, and J. Yang. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 148–155 (cit. on p. 17).

[73] H. Roodaki, K. Filippatos, A. Eslami, and N. Navab. “Introducing Augmented Reality to Optical Coherence Tomography in Ophthalmic Microsurgery”. In: *2015 IEEE International Symposium on Mixed and Augmented Reality, ISMAR 2015, Fukuoka, Japan, September 29 - Oct. 3, 2015*. 2015, pp. 1–6 (cit. on p. 18).

[74] *Endoscopic Vision Challenge 2015 - Instrument Segmentation and Tracking Sub-Challenge*. `https://endovissub-instrument.grand-challenge.org`. Accessed: 2018-01-30 (cit. on pp. 21, 31, 50).

[75] M. Chmarra, C. Grimbergen, and J Dankelman. “Systems for tracking minimally invasive surgical instruments”. In: *Minimally Invasive Therapy & Allied Technologies* 16.6 (2007), pp. 328–340 (cit. on p. 21).

[76] D. A. Forsyth and J. Ponce. *Computer vision: a modern approach*. Prentice Hall Professional Technical Reference, 2002 (cit. on p. 22).

[77] A. Yilmaz, O. Javed, and M. Shah. “Object tracking: A survey”. In: *Acm computing surveys (CSUR)* 38.4 (2006), p. 13 (cit. on pp. 22, 23, 31).

[78] D. G. Lowe. “Distinctive image features from scale-invariant keypoints”. In: *International journal of computer vision* 60.2 (2004), pp. 91–110 (cit. on p. 22).

[79] N. Dalal and B. Triggs. “Histograms of oriented gradients for human detection”. In: *IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2005. CVPR 2005.* Vol. 1. IEEE. 2005, pp. 886–893 (cit. on p. 22).

[80] M. Allan, P.-L. Chang, S. Ourselin, D. J. Hawkes, A. Sridhar, J. Kelly, and D. Stoyanov. “Image Based Surgical Instrument Pose Estimation with Multi-class Labelling and Optical Flow”. In: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part I*. Ed. by N. Navab, J. Hornegger, W. M. Wells, and A. Frangi. Springer International Publishing, 2015, pp. 331–338 (cit. on pp. 23, 34, 36).

[81] L. Breiman. “Random Forests”. In: *Machine Learning* 45.1 (2001), pp. 5–32 (cit. on p. 26).

[82] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. “Gradient-based learning applied to document recognition”. In: *Proceedings of the IEEE* 86.11 (1998), pp. 2278–2324 (cit. on p. 27).

[83] J. Long, E. Shelhamer, and T. Darrell. “Fully convolutional networks for semantic segmentation”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2015, pp. 3431–3440 (cit. on p. 27).

[84] K. Simonyan and A. Zisserman. “Very deep convolutional networks for large-scale image recognition”. In: *arXiv preprint arXiv:1409.1556* (2014) (cit. on p. 27).

[85] K. He, X. Zhang, S. Ren, and J. Sun. “Deep residual learning for image recognition”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2016, pp. 770–778 (cit. on pp. 27, 42).

[86] O. Ronneberger, P. Fischer, and T. Brox. “U-Net: Convolutional Networks for Biomedical Image Segmentation”. In: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*. Ed. by N. Navab, J. Hornegger, W. M. Wells, and A. F. Frangi. Cham: Springer International Publishing, 2015, pp. 234–241 (cit. on p. 27).

[87] I. Goodfellow, Y. Bengio, A. Courville, and Y. Bengio. *Deep learning*. Vol. 1. MIT press Cambridge, 2016 (cit. on p. 28).

[88] R. Sznitman, K. Ali, R. Richa, R. H. Taylor, G. D. Hager, and P. Fua. “Data-Driven Visual Tracking in Retinal Microsurgery”. In: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2012*. Ed. by N. Ayache, H. Delingette, P. Golland, and K. Mori. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 568–575 (cit. on pp. 28, 33, 36, 38, 47, 50).

[89] M. R. Pickering, A. A. Muhit, J. M. Scarvell, and P. N. Smith. “A new multi-modal similarity measure for fast gradient-based 2d-3d image registration”. In: *EMBC 2009,pp. 5821-5824 (2009)*. 2009 (cit. on p. 29).

[90] D. Bouget, M. Allan, D. Stoyanov, and P. Jannin. “Vision-based and marker-less surgical tool detection and tracking: a review of the literature”. In: *Medical Image Analysis* 35 (2017) (cit. on p. 33).

[91] A. Reiter and P. K. Allen. “An online learning approach to in-vivo tracking using synergistic features”. In: *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE. 2010, pp. 3441–3446 (cit. on pp. 33, 36).

[92] R. Richa, M. Balicki, E. Meisner, R. Sznitman, R. Taylor, and G. Hager. “Visual Tracking of Surgical Tools for Proximity Detection in Retinal Surgery”. In: *Information Processing in Computer-Assisted Interventions*. Ed. by R. H. Taylor and G.-Z. Yang. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 55–66 (cit. on pp. 33, 36).

[93] Y. Li, C. Chen, X. Huang, and J. Huang. “Instrument Tracking via Online Learning in Retinal Microsurgery”. In: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2014*. Ed. by P. Golland, N. Hata, C. Barillot, J. Hornegger, and R. Howe. Cham: Springer International Publishing, 2014, pp. 464–471 (cit. on pp. 34, 36).

[94] D. Sarikaya, J. J. Corso, and K. A. Guru. “Detection and localization of robotic tools in robot-assisted surgery videos using deep neural networks for region proposal and detection”. In: *IEEE transactions on medical imaging* 36.7 (2017), pp. 1542–1549 (cit. on pp. 34, 36).

[95] Z. Pezzementi, S. Voros, and G. D. Hager. “Articulated object tracking by rendering consistent appearance parts”. In: *International Conference on Robotics and Automation (ICRA)*. IEEE. 2009, pp. 3940–3947 (cit. on pp. 34, 36).

[96] Y. M. Baek, S. Tanaka, H Kanako, N. Sugita, A. Morita, S. Sora, R. Mochizuki, and M. Mitsuishi. “Full state visual forceps tracking under a microscope using projective contour models”. In: *IEEE International Conference on Robotics and Automation (ICRA), 2012*. IEEE. 2012, pp. 2919–2925 (cit. on pp. 34, 36, 38).

[97] A. Reiter, P. K. Allen, and T. Zhao. “Marker-less articulated surgical tool detection”. In: *Proc. Computer assisted radiology and surgery*. Vol. 7. 2012, pp. 175–176 (cit. on pp. 34, 36, 38).

[98] M. Allan, S. Ourselin, S. Thompson, D. J. Hawkes, J. Kelly, and D. Stoyanov. “Toward detection and localization of instruments in minimally invasive surgery”. In: *IEEE Transactions on Biomedical Engineering*. Vol. 60. 4. IEEE, 2013, pp. 1050–1058 (cit. on pp. 34, 36, 47).

[99] J. Zhou and S. Payandeh. “Visual tracking of laparoscopic instruments”. In: *J. Autom. Cont. Eng. Vol.* 2.3 (2014), pp. 234–241 (cit. on pp. 34, 36, 38).

[100] D. Bouget, R. Benenson, M. Omran, L. Riffaud, B. Schiele, and P. Jannin. “Detecting Surgical Tools by Modelling Local Appearance and Global Shape”. In: *IEEE Transactions on Medical Imaging* 34.12 (2015), pp. 2603–2617 (cit. on pp. 34, 36).

[101] L. Garcia Peraza Herrera, W. Li, C. Gruijthuijsen, A. Devreker, G. Attilakos, J. Deprest, E. Vander Poorten, D. Stoyanov, T. Vercauteren, and S. Ourselin. “Real-Time Segmentation of Non-Rigid Surgical Tools based on Deep Learning and Tracking”. In: *CARE workshop at International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer. 2016 (cit. on pp. 34, 36, 49).

[102] L. C. Garcia-Peraza-Herrera, W. Li, L. Fidon, C. Gruijthuijsen, A. Devreker, G. Attilakos, J. Deprest, E. V. Poorten, D. Stoyanov, T. Vercauteren, et al. “Toolnet: Holistically-nested real-time segmentation of robotic surgical tools”. In: *arXiv preprint arXiv:1706.08126* (2017) (cit. on pp. 35, 36).

[103] D. Pakhomov, V. Premachandran, M. Allan, M. Azizian, and N. Navab. “Deep Residual Learning for Instrument Segmentation in Robotic Surgery”. In: *arXiv preprint arXiv:1703.08580* (2017) (cit. on pp. 35, 36).

[104] R. Sznitman, A. Basu, R. Richa, J. Handa, P. Gehlbach, R. H. Taylor, B. Jedynak, and G. D. Hager. “Unified Detection and Tracking in Retinal Microsurgery”. In: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2011*. Ed. by G. Fichtinger, A. Martel, and T. Peters. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 1–8 (cit. on pp. 35, 36).

[105] A. Reiter, P. K. Allen, and T. Zhao. “Feature Classification for Tracking Articulated Surgical Tools”. In: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2012*. Ed. by N. Ayache, H. Delingette, P. Golland, and K. Mori. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 592–600 (cit. on pp. 35, 36).

[106] R. Sznitman, C. Becker, and P. Fua. “Fast Part-Based Classification for Instrument Detection in Minimally Invasive Surgery”. In: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2014*. Ed. by P. Golland, N. Hata, C. Barillot, J. Hornegger, and R. Howe. Cham: Springer International Publishing, 2014, pp. 692–699 (cit. on pp. 35, 36, 38, 40, 47).

[107] M. Alsheakhali, A. Eslami, H. Roodaki, and N. Navab. “CRF-based model for instrument detection and pose estimation in retinal microsurgery”. In: *Computational and mathematical methods in medicine* 2016 (2016) (cit. on pp. 35, 36).

[108] T. Kurmann, P. M. Neila, X. Du, P. Fua, D. Stoyanov, S. Wolf, and R. Sznitman. “Simultaneous Recognition and Pose Estimation of Instruments in Minimally Invasive Surgery”. In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer. 2017, pp. 505–513 (cit. on pp. 35, 36).

[109] X. Du, T. Kurmann, P.-L. Chang, M. Allan, S. Ourselin, R. Sznitman, J. D. Kelly, and D. Stoyanov. “Articulated Multi-Instrument 2D Pose Estimation Using Fully Convolutional Networks”. In: *IEEE Transactions on Medical Imaging* (2018) (cit. on pp. 35, 36).

[110] D. J. Tan and S. Ilic. “Multi-forest tracker: A chameleon in tracking”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2014, pp. 1202–1209 (cit. on p. 38).

[111] V. Belagiannis, C. Amann, N. Navab, and S. Ilic. “Holistic human pose estimation with regression forests”. In: *In: Perales, F.J., Santos-Victor, J. (eds.) AMDO 2014. LNCS, vol. 8563, Springer, Heidelberg (2014)*. 2014, pp. 20–30 (cit. on p. 39).

[112] Z. Kalal, K. Mikolajczyk, and J. Matas. “Tracking-learning-detection”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 34.7 (2012), pp. 1409–1422 (cit. on p. 40).

[113] S. S. Haykin. *Kalman Filtering and Neural Networks*. J. Wiley & Sons, Inc., 2001 (cit. on p. 41).

[114] I. Laina, C. Rupprecht, V. Belagiannis, F. Tombari, and N. Navab. “Deeper depth prediction with fully convolutional residual networks”. In: *Fourth International Conference on 3D vision (3DV)*. IEEE. 2016, pp. 239–248 (cit. on p. 43).

[115] Z. Li and D. Hoiem. “Learning without forgetting”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2017) (cit. on p. 49).

[116] S.-A. Rebuffi, A. Kolesnikov, G. Sperl, and C. H. Lampert. “icarl: Incremental classifier and representation learning”. In: *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*. 2017, pp. 2001–2010 (cit. on p. 49).

[117] R. Aljundi, P. Chakravarty, and T. Tuytelaars. “Expert gate: Lifelong learning with a network of experts”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2017, pp. 3366–3375 (cit. on p. 49).

[118] K. Shmelkov, C. Schmid, and K. Alahari. “Incremental Learning of Object Detectors Without Catastrophic Forgetting”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2017, pp. 3400–3409 (cit. on p. 49).

[119] W. Knight. *The Dark Secret at the Heart of AI*. Ed. by M. T. Review. [Online; posted 11-April-2017]. 2017 (cit. on p. 49).

[120] M. D. Zeiler and R. Fergus. “Visualizing and understanding convolutional networks”. In: *European conference on computer vision*. Springer. 2014, pp. 818–833 (cit. on p. 49).

[121] A. Mahendran and A. Vedaldi. “Understanding deep image representations by inverting them”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2015, pp. 5188–5196 (cit. on p. 49).

[122] J. Yosinski, J. Clune, A. Nguyen, T. Fuchs, and H. Lipson. “Understanding neural networks through deep visualization”. In: *arXiv preprint arXiv:1506.06579* (2015) (cit. on p. 49).

[123] K. Simonyan, A. Vedaldi, and A. Zisserman. “Deep inside convolutional networks: Visualising image classification models and saliency maps”. In: *arXiv preprint arXiv:1312.6034* (2013) (cit. on p. 49).

[124] J. Su, D. V. Vargas, and S. Kouichi. “One pixel attack for fooling deep neural networks”. In: *arXiv preprint arXiv:1710.08864* (2017) (cit. on p. 50).

[125] N. Carlini and D. Wagner. “Towards evaluating the robustness of neural networks”. In: *IEEE Symposium on Security and Privacy (SP)*. IEEE. 2017, pp. 39–57 (cit. on p. 50).

[126] N. Papernot, P. McDaniel, S. Jha, M. Fredrikson, Z. B. Celik, and A. Swami. “The limitations of deep learning in adversarial settings”. In: *IEEE European Symposium on Security and Privacy (EuroS&P)*. IEEE. 2016, pp. 372–387 (cit. on p. 50).

[127] M. Brouillette. *Deep Learning Is a Black Box, but Health Care Won’t Mind*. Ed. by M. T. Review. [Online; posted 17-April-2017]. 2017 (cit. on p. 50).

[128] *Endoscopic Vision Challenge 2017 - Robotic Instrument Segmentation Sub-Challenge*. `https://endovissub2017-roboticinstrumentsegmentation.grand-challenge.org`. Accessed: 2018-01-30 (cit. on p. 50).

[129] F. Milletari, N. Rieke, M. Baust, M. Esposito, and N. Navab. “CFCM: Segmentation via Coarse to Fine Context Memory”. In: *Medical Image Computing and Computer Assisted Intervention – MICCAI 2018*. Ed. by A. F. Frangi, J. A. Schnabel, C. Davatzikos, C. Alberola-López, and G. Fichtinger. Cham: Springer International Publishing, 2018, pp. 667–674 (cit. on p. 103).