Visual Instrument Guidance in Minimally Invasive Robot Surgery

C. Staub, G. Panin and A. Knoll
Robotics and Embedded Systems
Technical University Munich
Munich, Germany
{*staub* | *panin* | *knoll*}*@in.tum.de*

Robert Bauernschmitt
Department of Cardiovascular Surgery
German Heart Center Munich
Munich, Germany
bauernschmitt@dhm.mhn.de

***Abstract*—Surgical tool tracking is an important key functionality for many high-level tasks such as the visual guidance of surgical instruments or automated camera control. Readings from robot encoders and the kinematic chain are usually error prone in this kind of complex setup, but still allow for a coarse pose estimation of the instruments in image space. This information can be utilized to (re-)initialize image-based tracking in case of tracking loss and supervise the tracking process. Accounting for the difficult environmental conditions in surgery, the choice of an appropriate tracking modality is important. We have chosen the Contracting Curve Density algorithm (CCD) that maximizes the separation of local color statistics along the contour of a model in contrast to the background. As an application example, the visual guidance of laparoscopic instruments under trocar kinematic is presented.**

Keywords-robotic surgery; minimally invasive surgery; instrument tracking; visual guidance.

I. INTRODUCTION

This paper is an extended version of a conference paper referenced in [1]. It details the formerly announced tracking of surgical instruments and proposes an approach that combines both pose prediction using kinematically derived data and image-based tracking. The extracted position of the instrument is then utilized for visual guidance, a prerequisite for many automation scenarios.

Endoscopic surgery is a challenging technique and has had significant impact on both patients and surgeons. Minimally invasive surgery (MIS) techniques avoid large cuts and patients profit from less pain and collateral trauma. Therefore, the time of hospitalization and the infection rate can be reduced. Unfortunately, surgeons have to cope with increasingly complex working conditions. Long instruments which are unfamiliar and sometimes awkward to operate for the surgeon, are used through small incisions or ports in the body of the patient to perform the intervention. In contrast to the conventional open surgery, visual access to the internal operating scenery is not feasible. This constrains the visual perception to an endoscopic view without an intuitive depth perception or hand-eye coordination. The introduction of telemanipulators, such as the daVinci^{TM}machine [2], has overcome these limitations and is a remarkable example of the ongoing research. The instruments can now be controlled remotely by a surgeon sitting at a master console, which can be placed somewhere in the operation theater. A stereoscopic endoscope provides a 3D view on the situs and improves the perceptual limitations of flattened images. The master console is equipped with sophisticated input devices and provides an intuitive handling of the surgical instruments (Cartesian control without any chopstick effect). The robots at the slave system offer as much freedom of movement as the surgeon's own hand would do in conventional open surgery. Also immersiveness is often improved by means of haptic feedback.

Recently, automation of error-prone and recurrent (sub-) tasks that yield to the quick fatigue of surgeons and noticeable account for a higher overall surgery time have drawn the attention of researchers. Given that knot-tying occurs frequently during surgery, automating this challenging subtask is tackled by several groups (e.g., [3]–[5]). Furthermore, techniques for assisting the surgeon with visually guided instruments (see [6]–[9]) and autonomously navigated endoscopic cameras have been developed (e.g., [10], [11]). Visual servoed instruments are a promising approach in robot-assisted surgery to introduce autonomy and to overcome intrinsic system limitations, often caused by calibration problems. Since visual servoing uses feedback from one ore more cameras to guide a robotic appendage, robust tracking of surgical tools is of particular interest for this kind of application. Also for the reason of documenting and benchmarking surgical interventions, and to anticipate potential mistakes in the surgical workflow, modeling and analyzing surgical procedures has become an active field of research, whereat instrument tracking plays an important role (e.g., [12], [13]).

Despite the manifold of challenges in minimally invasive surgery and the above mentioned achievements in partially autonomous navigation and manipulation, the visual identification, segmentation, and tracking of operated surgical tools during surgery is a crucial requirement for developing techniques that assist the surgeon. As most of the methods require position information of the surgical instrument, a robust and precise automatic detection is

the first step towards higher level functionality. In this paper, we present an approach that allows for a markerless tracking of surgical instruments and its application to visual instrument guidance.

In literature, many of the proposed instrument tracking approaches rely on image processing techniques that use either pure color information or additional geometrical knowledge. Wei et al. [10] analyzed the typical color distribution in laparoscopic images to identify an adequate color for optical markers that are attached to the distal end of the instrument. The marker is segmented in HSV color space and background noise is filtered at a rate of 17Hz. Uckert et al. [14] includes additional shape information about the shaft to fit a bounding box to the color-classified pixels. In order to cope with the typical camera distortion of endoscopes, two different shapes are used: a trapezoidal for near-field cases and a rectangular for far-field cases. In [15], it was taken advantage of the metallic appearance of the shaft to track gray regions by joint hue saturation color features. A seeded region growing method was implemented, operating at 13fps. The fulcrum is estimated with a series of images in order to project an approximated instrument direction and shape into the image. Voros et al. [16] also reduce the search space by considering the insertion point of the instrument. At the beginning of the procedure, the fulcrum has to be visible in the image and is marked with a "vocal mouse". They state that any kind of surgical instrument can be detected since no color information is used, but only the gradients of the instrument edges, constrained by the incision point. To enhance the computation speed, the image resolution is reduced to 200×100 pixels. The precision of the predicted tip position ranges around 11 pixels. The *Center for Computer Integrated Surgical Systems and Technology* (CISST, Johns Hopkins University, Baltimore) tracks the articulated DaVinci^{TM}instruments. Burschka et al. [17] used template images of the instrument to detect the position of the forceps in stereo images, enriched with additional information and orientation information derived from the trajectory provided by the robot. The method works in real-time, but they report that the kinematic data suffers from significant rotational and translational errors. More recently, the CISST reported a general purpose articulated object tracker [18] and demonstrated its application to surgical scenarios. The geometry and kinematics of the objects have to be known a priori. The appearance of different body parts is modeled by a class-conditional probability and compared with the image after rendering the target object geometry. So far, images are hand segmented to train the appearance model and computation time is around $5sec$ per frame at a resolution of 640×480.

The remainder of this paper is as follows. In Section II, the underlying hard- and software that is used for all experiments is introduced. Section III outlines our tracking approach that combines servo readings from the robot as well as image analysis to robustly track surgical instruments in image space. In Section IV, the tracking output is applied to the visual guidance of the instruments. Finally, in Section V, experimental results are presented. Conclusions are drawn and future work is outlined in Section VI.

Figure 1. **Hardware Setup.** Ceiling mounted robots with surgical instruments and master console

II. Robotic System

Several works that are engaged with computer vision aspects in robot-assisted surgery are drawn on a simplified environment. Either the system lacks an endoscopic camera (that usually suffers from strong distortions) or the evaluation was performed within an unrealistic environment. In the majority of cases dimensions of the workspace or distances between camera and instrument are incorrect due to a missing multi-arm setup or port-kinematics. Therefore, the findings of this research project have been assessed within a realistic scenario of robotic surgery [4].

A. Research platform for MIS

As illustrated in Figure 1, the slave manipulator of the system consists of four ceiling-mounted robots which are attached to an aluminum gantry. The robots have six degrees of freedom (DoF) and are equipped with either a 3D endoscopic stereo camera or with minimally invasive surgical instruments, which are originally deployed by the DaVinci^{TM}system. The surgical instruments have 3DoF. A micro-gripper at the distal end of the shaft can be rotated and adaption to pitch and jaw angles is possible. Through the aid of a magnetic clutch the instruments can be interchanged quickly for better handling. The mechanism will also disengage the instruments if forces beyond a certain level are exerted and prevents damage in case of a severe collision. Forces exerting on the instruments are measured by

strain gauge sensors and fed back to the operator by means of haptic devices. The master-side manipulator is equipped with a 3D display, some foot switches for user interaction (such as starting and stopping the system or executing the piercing process) and with the main in-/output devices, two PHANToM^{TM}haptic displays. The devices are used for 6DoF control of the slave manipulator, but also provide 3DoF force feedback derived from the measurements at the instruments. The control software of the system realizes trocar kinematics, whereby all instruments will move about a fixed fulcrum after insertion into the body.

B. Distributed Software Environment

The multi-tier software architecture of our system is distributed over 3 standard PC's: a *simulation and control* PC, a *vision* PC (equipped with a NVIDIA^{TM}Quadro FX 580 graphics card) and one computer is connected to a *CAN* network (cp. Fig. 2). The commands for the servo-motors that control the joints of the instrument as well as the data that is provided by the amplifiers of the strain gauge sensors are communicated between the simulation PC and the PC that is connected to the CAN network. The GUI of the simulation environment comprises an interface to a 3D model of the scene, which can be manipulated in real time. Parameters of each model can be adjusted and joint angles of the robots can be altered this way. New trajectories can be generated by means of a key framing module, incorporating a collision detection. On one hand, joint data is directly sent to the robot hardware, on the other hand the poses of the instruments and the robots are synchronized with the "Vision PC" for further application in image analysis. For this reason, enough computing power can be provided for image analysis, i.e., instrument tracking, visual servoing or augmented reality. Most of the image processing tasks run in individual threads that have access to an image database, which holds up-to-date images provided by the stereoscopic endoscope.

C. System Calibration

Unfortunately, many possible error sources contribute to a comparably high aberration between the real-world hardware and the underlying CAD models of the simulation environment. Camera calibration, exact mounting of the surgical instruments (concerning the magnet coupling) and even the instruments itself introduce quiet large errors. For instance, the flexibility and play of the carbon fiber shaft of the instruments and the gripper at the distal end may vary approximately $\pm 1.5cm$ (cp. Fig. 3(c)). Furthermore, the ceiling mounting of the robots afflicts several intrinsic aberrations, such as variations in the dimensions of the elements and errors of mounting angles. Since all errors sum up, the exact Cartesian position of the distal end of the instrument deviates from the emulation. In order to minimize all intrinsic errors and to establish the

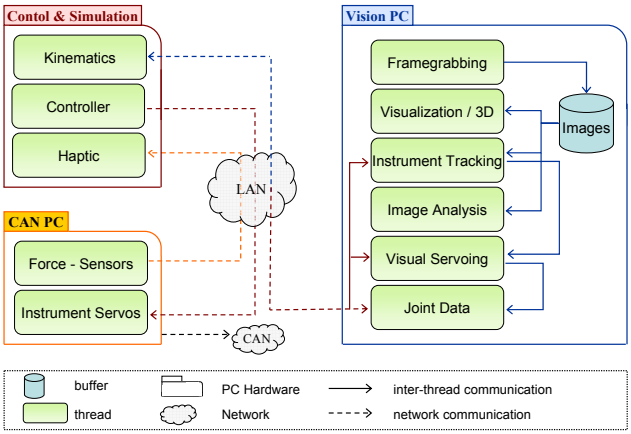


Figure 2. **Software Architecture.** The software of the system is distributed across 3 PC's that communicate via network connections

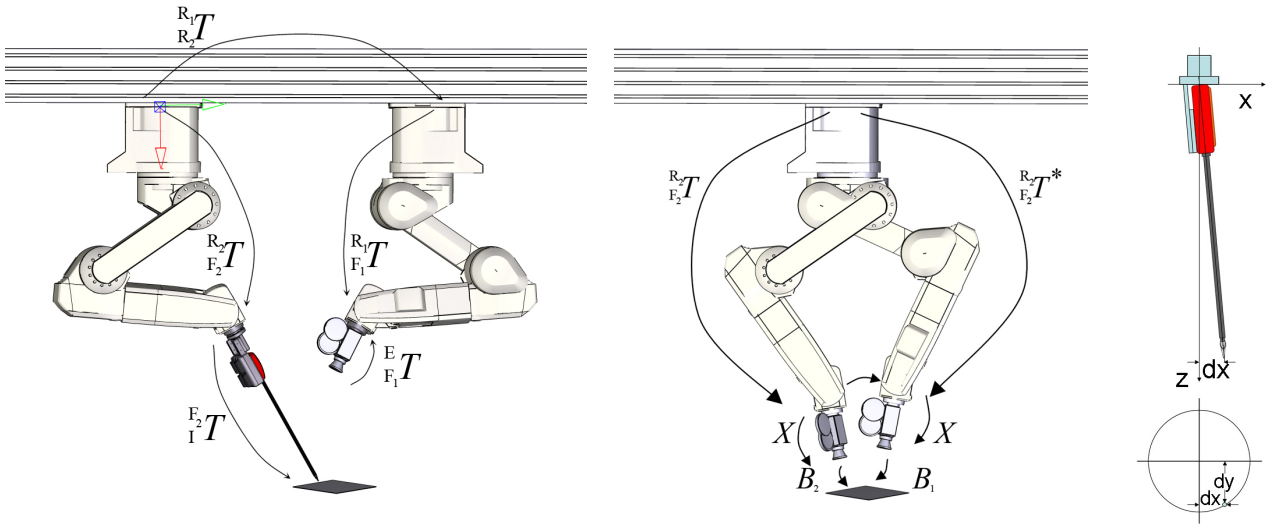
transformations between the individual system components (such as the instrument, different robot bases, etc.) a precise calibration has to be performed. However, an important issue for the acceptance of robotic systems in the operating theater are pre-calibrated components to avoid complicated or long-lasting procedures during an intervention or ahead.

As mentioned above, the robots are mounted on a gantry, assembled of profiled girders. Particularly the coplanarity of the robot's base relative to its attachment cannot be guaranteed and is hardly to be measured. In order to overcome intrinsic variations of the single aluminum elements and errors of mounting angles, a calibration between each of the robot basements is performed.
To align the basements of two robots R_1 and R_2 we employ the following error model:

$$ {}^{0}_{R_1}T \cdot {}^{R_1}_{C}T = {}^{0}_{R_2}T \cdot {}^{R_2}_{C}T \quad (1) $$

In this equation ${}^{0}_{R_1}T$ is the position of the base of the robot R_1, expressed in global coordinates. In order to measure the relative displacement between the robots a calibration frame C in global coordinates is defined and the position and orientation of this frame is measured in local coordinates of each robot. The frame can be replicated by mounting a precisely manufactured calibration trihedron of known size on the flange of both robots. A number of points $M = (p_1, \ldots, p_i)$ are labeled on a checkerboard calibration plate that is positioned in-between the robots, reachable for all four arms (figure 3(a)). The trihedrons of the robots R_1 and R_2 are then driven to all points and the corresponding relative transform (e.g., ${}^{R_2}_{C}T$) can be determined.

Mounting displacements of the robots are not the only source of errors in the system. If we go further down, we find that also the attachment of the instruments bears



(a) System Calibration Overview (b) Hand-Eye Calibration (c) Excentricity

Figure 3. **System Calibration.** Figure 3(a) depicts a schematic overview of the kinematic chain. The hand-eye calibration is exemplified in illustration 3(b). Figure 3(c) shows excentricity and play of the instrument shaft.

certain variances. The magnetic clutch as well as the mechanical fit and the flexibility of the carbon fiber shafts results in a quite high aberration. In case of the endoscopic camera a hand-eye calibration solves this problem [19]. One way to calculate the displacement of an attached endoscope with respect to the flange of the robot, is to solve ${}^{R_2}_{F_2}T \cdot X = X \cdot {}^{R_2}_{F_2}T^*$ (compare Fig. 3(b)). Regarding the surgical tools, this method would introduce two issues in the context of medical procedures: on one hand it is difficult to create a calibration pattern which can be precisely reached by the forceps or attached to the shaft. On the other hand, it is challenging to perform the calibration in the sterile environment of an operating room. The proposed method allows a pre-calibration of every instrument, which can be applied to the system previous to the intervention. To compensate the excentricity, an approximation which simplifies the calculation and applies only to small angles is used. An aberration dx and dy from the center will lead to a positional error or approximately $\sqrt{dx^2 + dy^2}$. The parameters shown in Fig. 3(c) can be found by positioning the instrument over a planar surface with the z-axis of the robot's tool system normal to the surface. By rotating the end effector about $360°$ a circular path is described and the relevant parameters can be determined. In order to compensate for this excentricity, the found correctional transformation has to be applied to the end effector prior to the calculation of the inverse kinematics of the robot.

State of the art endoscopes offer physicians a wide-angled field of view which is imperative for minimally invasive interventions. In order to determine the projective parameters of the camera system a calibration procedure is to be performed a priori.

III. Instrument Tracking

The tracking of surgical tools is particularly challenging due to the changing appearance of the background (e.g., background movement through organs, non-uniform and time-varying lightning conditions, smoke caused by electro-dissection and specularities), but also due to the partial occlusion of the instrument and body fluids that may change the appearance of the instrument itself. In many cases of surgical tool tracking the tracking is constricted to a sequential "frame-by-frame detection" (also referred to as *detection*), rather than including a motion model. Accordingly, no optimization of the configuration space or pose prediction is performed over time.

A. Instrument Tracking Supported by Kinematic Prediction

In a Bayesian *prediction-correction* context, the state of the object is updated by integrating posterior statistics and therewith knowledge about time-depending characteristics of the movement. This "intelligence" within our tracking pipeline is provided by a Kalman filter [20] that is running on the output of a contour tracker, known as contracting curve density algorithm (CCD), based on the separation of local color statistics (see [21], [22]). The separation is performed between the object and the background regions, across the projected shape contour of a CAD model under a predicted pose hypotheses. An overview of the process flow is given in Figure 5.
Tracking always involves a detection step to initialize the system in the very first frame or after encountering a

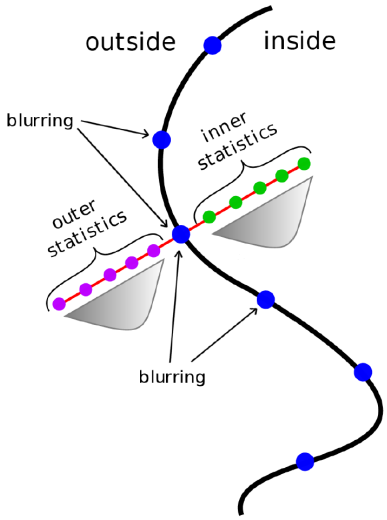


Figure 4. The CCD algorithm tries to maximize the separation of color statistics between two image regions. The algorithm first samples pixels along the normals for collecting local color statistics.

track loss. Instead of simply relying upon visual data, we take an estimated object pose, derived from the kinematic measurement of robot sensor readings. The precision of this approximation is limited due to the absolute accuracy of our system (and the performed calibration).

The idea of integrating joint angle measurements for tracking purposes was e.g., also applied by Ruf et al. [23] to track a polyhedral tool and simultaneously adapt inaccuracies in the static calibration of the robot. To restrict the initial search from the first frame to a specific region is computational more efficient than a complete image analysis and can also be considered from a biological point of view: Biologically inspired algorithms seek to direct the attention rapidly towards a region of interest, using an attention-based type of filter, and only process a smaller amount of the visual input data [24]. *Bottom-up* approaches compute visual salient features, such as regions of high contrast, local scene complexity or high scene dynamics. The second type of visual attention is often referred to as *top-down* attention, as the attention is controlled from higher areas of cognition. Kinematic measurements, which are fed to the visual information processing by another software component (thus, a higher area of cognition), can guide the attention directly to a region of interest

B. Model Building

Our system is equipped with the EndoWrist™needle driver tools that are originally deployed with the DaVinci™system. The instruments are composed of a long shaft, a wrist joint and two brackets. It is represented as a polygonal mesh model (cp. Fig. 6) with 6DoF (3 rotations, 3 translations) by a 4×4 transformation matrix in our simulation environment. In order to represent the instrument in 2D image space, we build a rectangular model with rounded edges at the distal end and neglect the brackets. As already mentioned, CCD maximizes local color statistics (object vs. background) along the model contour. More detail is given in Section III-C. Only three of the object edges can be used for normal contour point sampling and collecting statistics for the CCD algorithm. The fourth edge has to be neglected, since it is not an exterior edge of the shaft and therewith no color separation between model and background is possible. The inclusion of the edge would yield to irrepressible shifting of the model alongside of the shaft.

The instrument's 6 pose parameters are reduced to a planar roto-translational pose s with scale h and rotation θ by projecting the 3D model into the image plane.

$$s = (t_x, t_y, h, \theta) \tag{2}$$

An important aspect of the proposed approach is the use of pose estimates, derived from the kinematic chain, that are fed to the tracking pipeline prior to the visual tracking. Instead of referring the tracking parameters s to global image coordinates, we align the tracking frame with the estimated orientation and position of the instrument T_{ref} in every cycle (cp. Fig. 6). Therewith, the tracking is performed in a *local* coordinate system that is axis aligned with the frame of the instrument model. Since the uncertainty of the state

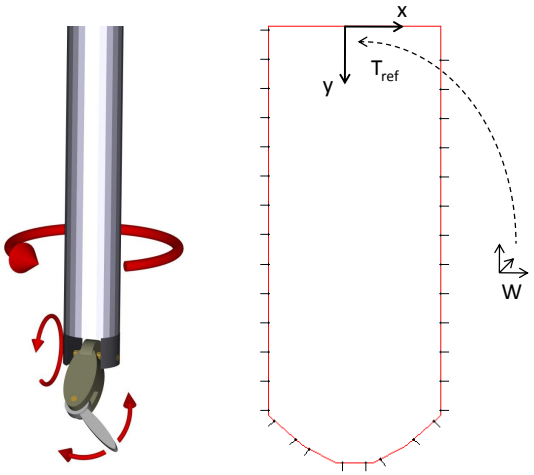


Figure 6. Instrument CAD model and tracking contour model of the shaft with sampling normals.

hypothesis is represented by a squared covariance matrix (with the dimension of the state), we can now alter the matrix and set a higher confidence in the direction of the shaft (the y-axis). This anticipates an uncontrolled sliding of the model alongside of the instrument shaft. The entries in the covariance matrix are found empirically for all DoFs.

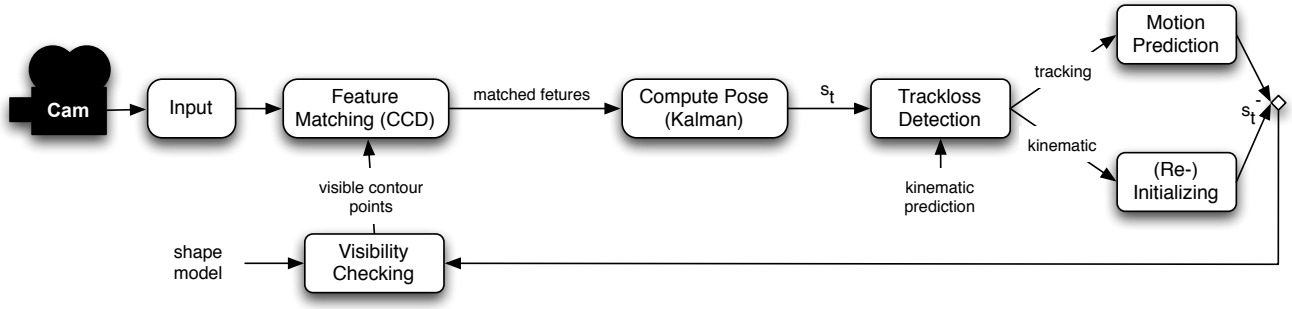


Figure 5. **Tracking Pipeline.** The camera pose can be obtained after calibrating the extrinsic parameters and the overall system. The kinematic measurement of the instrument in 6 degrees of freedom is transfered to a 2D model with 4 DoF (t_x, t_y, h, θ_x). It is used to (re-)initialize the Contracting Curve Density algorithm and to supervise the tracking quality.

C. Tracking with CCD

As already mentioned above, tracking in the context of MIS procedures is exacerbate by changing environment conditions. Simple color segmentation approaches often fail due to varying lightening conditions of different light sources or need a sophisticated fine tuning of parameters. Algorithms that are based upon edge detection suffer from the large amount of feature edges from the background. Figure 7 shows a typically intra-operative scene with an artificial heart and tissue in the background. Neither the Sobel- nor the Canny operator can distinguish the instrument shaft reliable from the background.
The amenity of the CCD modality is that the model's appearance is adjusted over time, since local color statistics are computed in every tracking cycle and maximized according to the shape of the model. Therefore, the method can be applied to marker-based as well as markerless tracking. In fact, the color or texture of the tracked object does not matter, as long as a separation in terms of color between object and background can be achieved. Also a change of the appearance over time (e.g., an account of body liquids) does not disturb the tracking if not the entire object is affected at once.

After setting the initial pose, a Kalman filter generates a prior state hypothesis s_t^- by applying a Brownian motion model to the previous state (s_{t-1}).

$$s_t^- = s_{t-1} + w_t \tag{3}$$

with w being a white Gaussian noise sequence.
The CCD modality requires a sampling of good features for tracking from the object model under the given pose s_t^- and camera view. As a first step, the visible internal and external edges from the polygonal mesh model have to be identified under the current pose hypothesis. Alongside of this contour a set K of uniformly distributed sampling points $\{h_1, \ldots, h_k\}$ is taken to collect color statistics around each sample position on each side of the contour. The basic idea of CCD is to maximize the separation of local color statistics between the two sides of the object boundaries (object vs. background) [21]. The colored shaft of the instrument supports this idea by varying from red tissue and organs. Contemporaneously, the algorithm can account for small change of the shaft appearance over time (e.g., from body liquids), since the statistics are updated in every iteration.
We first sample points along the respective normals, separately collect the statistics, and afterwards blur each statistic with the neighboring ones (cp. Fig. 4). From each contour position h_i, foreground and background color pixels are collected along the normals n_i up to a distance L (that is manually defined and fix), and local statistics up to the 2^{nd} order are estimated

$$\begin{aligned} v_i^{0,B/F} &= \sum_{d=1}^{D} w_{id} \\ v_i^{1,B/F} &= \sum_{d=1}^{D} w_{id} I(h_i \pm L\bar{d}n_i) \\ v_i^{2,B/F} &= \sum_{d=1}^{D} w_{id} I(h_i \pm L\bar{d}n_i) I(h_i \pm L\bar{d}n_i)^T \end{aligned} \tag{4}$$

(5)

with $\bar{d} \equiv d/D$ the normalized contour distance, where the $\pm$ sign is referred to the respective side, and image values I are 3-channel RGB. The local weights w_{id} decay exponentially with the normalized distance, thus giving a higher confidence to observed colors near the contour.
Single line statistics are afterwards *blurred* along the contour, providing statistics distributed on local areas

$$\widetilde{v}_i^{o,B/F} = \sum_j exp(-\lambda\,|i-j|) v_j^{o,B/F}; o = 0, 1, 2 \tag{6}$$

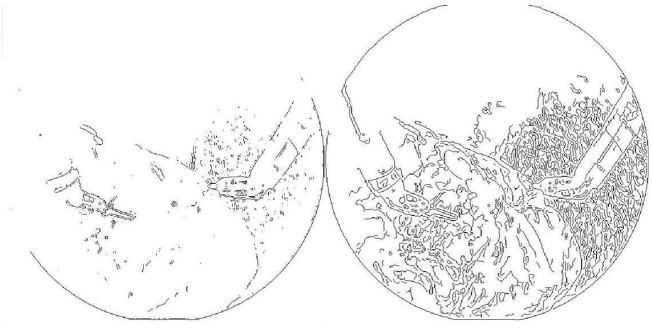


Figure 7. **Edge detection.** The images show edge detection results of the Sobel (left) and the Canny (right) filter. In both cases the tool shaft can hardly be distinguished from background noise. The organ surface comprises many small vessels and structures that raise edges in the vicinity of the tool tip.

and finally normalized

$$\overline{I}_i^{B/F} = \frac{\widetilde{v}_i^{1,B/F}}{\widetilde{v}_i^{0,B/F}} \qquad (7)$$
$$\overline{R}_i^{B/F} = \frac{\widetilde{v}_i^{2,B/F}}{\widetilde{v}_i^{0,B/F}}$$

in order to provide the two-sided, local RGB means $\overline{I}$ and (3×3) covariance matrices $\overline{R}$.

The second step involves computing the residuals and Jacobian matrices for the Gauss-Newton pose update. For this purpose, observed pixel colors $I(h_i + L\overline{d}n_i)$ with $\overline{d} = -1, \ldots, 1$ are classified according to the collected statistics (8), under a fuzzy membership rule $a(x)$ to the foreground region

$$a(\overline{d}) = \frac{1}{2}\left[erf\left(\frac{\overline{d}}{\sqrt{2}\sigma}\right) + 1\right] \qquad (8)$$

which becomes a sharp $\{0; 1\}$ assignment for $\sigma \to 0$; pixel classification is then accomplished by mixing the two statistics accordingly

$$\hat{I}_{id} = a(\overline{d})\overline{I}_i^F + (1 - a(\overline{d}))\overline{I}_i^B \qquad (9)$$
$$\hat{R}_{id} = a(\overline{d})\overline{R}_i^F + (1 - a(\overline{d}))\overline{R}_i^B$$

and color residuals are given by

$$E_{id} = I(h_i + L\overline{d}n_i) - \hat{I}_{id} \qquad (10)$$

with covariances $\hat{R}_{id}$.

Finally the $(3 \times n)$ derivatives of E_{id} can be computed by differentiating (8) and (10) with respect to the pose parameters

$$J_{id} = \frac{\partial \overline{I}_{id}}{\partial s} = \frac{1}{L}\left(\overline{I}_i^F - \overline{I}_i^B\right)\frac{\partial a}{\partial \overline{d}}\left(n_i^T \frac{\partial h_i}{\partial s}\right) \qquad (11)$$

which are stacked together in a global Jacobian matrix $\mathbf{J}_{ccd}$. The state is then updated using a Gauss Newton step:

$$s = s + \Delta s \qquad (12)$$
$$\Delta s = \mathbf{J}_{ccd}^{+}\mathbf{E}_{ccd}$$

The optimization is done until the termination criteria is satisfied $(\Delta s \approx 0)$.

The tracking pose is observed and compared to the kinematic prediction in order to detect tracking loss. This can either be a total loss of tracking, or the sliding of the model alongside of the instrument shaft. For this purpose, we restrict the output of the visual system to lie within a certain range, derived from the current prediction (position and angular values). Since we perform the tracking in a local coordinate frame, we can also easily set pose limits from this values. Furthermore, the estimate of the pose covariance matrix gives a hint for the quality of the tracking. By choosing an empirical maximum threshold for the determinant of the posterior covariance, we can imply a tracking loss.

IV. Application to Visual Guidance

The tracking approach introduced above is utilized to visually guide the surgical instruments and the endoscopic camera.

Although the robotic system is calibrated carefully, the above mentioned inherent imprecisions cannot be determined with a satisfying accuracy to position instruments with a very high precision (which means $1mm$ or below). In particular, the transformation ${}^{F_2}_{I}T$ that follows from the aberration of the carbon fiber shaft of the instrument (cp. Fig. 3(c)) cannot be minimized to a satisfying amount.

Visual servoing is a popular approach to guide a robotic appendage (i.e., a surgical instrument in our case) using visual feedback from a camera system. In general, visual servoing can roughly be divided into two categories: position-based visual servoing control (PBVS), in which a Cartesian coordinate is estimated from image measurements and image-based visual servoing (IBVS) approaches, which seek to extract features directly from an image series. In general, the accuracy of image-based methods for static positioning tasks is less sensitive to calibration than PBVS [25]. Image-based servoing does not depend as much on calibration as the error is reduced directly in image pixels. However, a practical difficulty during the alignment of surgical instruments with a desired position in space lies in the fact that the instrument is not necessarily in the field of view of the camera and therewith no image-features can be extracted. Hence, we first drive the instrument to a Cartesian coordinate (reconstructed using stereopsis of the 3D endoscope) which is in the field of view of the camera.

Since the 3D reconstruction suffers from a certain error (caused by the mentioned intrinsic errors) we continue with image-based servoing to overcome the remaining distance. In fact, an eligible point close to the final position is chosen.

Given a target position that the robot is to reach, visual servoing aims to minimize an error $e(t)$, typically defined by

$$e(t) = s(m(t), a) - s^* \tag{13}$$

where s^* represents the target pose, $s(m(t), a)$ the measured pose, $m(t)$ the measured image feature points and a any additional knowledge needed, such as information from the camera calibration. The function $s(m(t), a)$ characterizes the end point of the tool tip of an instrument carried by the robot. In PBVS the position of the tracked features is extracted from the camera image coordinates and projected to the world frame by the mapping a, determined during camera calibration. The target position can be extracted from image features in a similar way. While PBVS minimizes the error $e(t)$ in world coordinates and the camera is treated as a 3D positioning sensor, IBVS directly tries to find a mapping from the error function to a commanded robot motion.
As mentioned above, PBVS is used to drive the instrument to a reconstructed point, which is located within the view of the camera. As soon as this point is reached, the remaining distance to the target goal is minimized in image coordinates. In many IBVS scenarios the camera is attached to the robot which is to be commanded (eye-in-hand configuration) and therewith the velocity of the camera ξ is calculated. In our setup, the instrument and the endoscope are carried by two different robots and the calculated velocity ξ has to be transformed to the robot that carries the instrument.
A single image feature, for instance the tip of an instrument or a carried needle, is tracked in both left and right camera coordinates. The feature vector $s = (x_L, x_R)^T = (u_L, v_L, u_R, v_R)^T$ comprises these coordinates:

$$s(t) = \begin{bmatrix} u(t) \\ v(t) \end{bmatrix} \tag{14}$$

Its derivative $\dot{s}(t)$ is referred to as image feature velocity. It is linearly related to the camera velocity $\xi = [\ \upsilon \quad \omega\ ]^T$, which is composed of linear velocity υ and angular velocity ω. The relationship between the time variation of the feature vector s and the velocity in Cartesian coordinates ξ is then established by

$$\dot{s} = L_s \xi \tag{15}$$

where L is the *interaction matrix* or *image Jacobian* [26]. The interaction matrix L_x related to an image point $x = (u, v)^T$ reads as follows:

$$L_x = \begin{bmatrix} -\frac{1}{z} & 0 & \frac{u}{z} & uv & -(1+u^2) & v \\ 0 & -\frac{1}{z} & \frac{v}{z} & 1+v^2 & -uv & -u \end{bmatrix} \tag{16}$$

Variable z represents the depth of a point relative to the camera frame. There exist different ways to approximate the value of z, for example via triangulation in a stereo setup or via pose estimation. Most of the existing methods assume an calibrated camera, even if the impact of the calibration is not very high. Few systems even assume a constant depth of the tracked feature and therewith a constant image Jacobian. In our approach, variable z is estimated via the kinematic chain of the system. The interaction matrix can then be updated online and the approach is easily transferable to miscellaneous camera configurations. For instance, we equipped another robot arm with a second monocular Fujinon™endoscope that provides a different view on an object. Using equations (13) and (15) we obtain $\dot{e} = L_e \xi$ and our final control law

$$\xi = \lambda L_e^+ e \tag{17}$$

where λ is a positive gain factor and L_e^+ the Moore-Penrose pseudo-inverse of L_e.
As mentioned above, a single visual feature s is tracked in the left and right images equation (15) is rewritten as

$$\begin{bmatrix} \dot{x}_L \\ \dot{x}_R \end{bmatrix} = \begin{bmatrix} L_L \\ L_R \, {}^R_L V \end{bmatrix} \xi_L \tag{18}$$

The spatial motion transform ${}^R_L V$ to transform velocities expressed in the right camera frame R to the left camera frame L is given by

$${}^R_L V = \begin{bmatrix} {}^R_L R & S(t)\, {}^R_L R \\ 0 & {}^R_L R \end{bmatrix} \tag{19}$$

where $S(t)$ is the skew symmetric matrix associated with the linear transformation vector t and where (R, t) is the transformation from the left to the right camera frame.

To consider the characteristics of the trocar kinematic during minimally invasive surgery, the instrument movement at the fulcrum has to be zero in all directions that are perpendicular to the instrument shaft. Since the location of the incision point is well-known from the simulation software, the 6DoF motion of the robot can be constrained to 4DoF at the trocar. The velocities ${}^T_T \xi = ({}^T_T \upsilon, {}^T_T \omega)^T$ at the trocar point T and the velocities ${}^I_I \xi = ({}^I_I \upsilon, {}^I_I \omega)^T$ of the instruments tip I are related as follows:

$$\begin{aligned} {}^T_I V \cdot {}^I_I \xi &= {}^T_T \xi \\ \Leftrightarrow \begin{bmatrix} {}^T_I R & S({}^T_I t)\, {}^T_I R \\ 0 & {}^T_I R \end{bmatrix} \begin{bmatrix} {}^I_I \upsilon \\ {}^I_I \omega \end{bmatrix} &= \begin{bmatrix} {}^T_T \upsilon \\ {}^T_T \omega \end{bmatrix} \end{aligned} \tag{20}$$

Assuming a straight shaft, ${}^T_I R$ is the identity matrix and $t = (0, 0, d)^T$ with d being the insertion depth of the instrument. Since only the z-direction (the direction of the shaft) is free to move, the linear velocity at the insertion point is denoted by ${}^I_I \upsilon = (0, 0, {}^I_I \upsilon_z)$. Solving (20) yields to

$${}^I_I \omega_x = -\frac{{}^I_I \upsilon_y}{d} \quad \text{and} \quad {}^I_I \omega_y = \frac{{}^I_I \upsilon_x}{d} \tag{21}$$

So far, we covered the control of surgical instruments. Furthermore, automated camera control (e.g., the endoscope automatically follows an instrument) is also of high interest to assist surgeons. The control law is similar to the instrument control, but in contrast, we prohibit movements in the directions of the shaft. For safety reasons of the patient it is not suitable that the endoscope induces depth motion. Taking Eqn. (21) into account and setting the camera velocities ${}^C_C v_z = {}^C_C \omega_z = 0$ we obtain the new interaction matrix L_{cam}

$$\begin{aligned} \dot{s} &= \begin{bmatrix} L_v \\ L_\omega \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} \qquad (22) \\ &= \begin{bmatrix} -\frac{1}{z} & 0 \\ 0 & \frac{1}{z} \end{bmatrix} \begin{bmatrix} v_x \\ v_y \end{bmatrix} \\ &+ \begin{bmatrix} xy & -(1+x^2) \\ 1+y^2 & -xy \end{bmatrix} \begin{bmatrix} \omega_x \\ \omega_y \end{bmatrix} \\ &= \underbrace{\begin{bmatrix} -\frac{1}{z} - \frac{1}{d}(1+x^2) & -\frac{1}{d}xy \\ -\frac{1}{d}xy & -\frac{1}{z} - \frac{1}{d}(1+y^2) \end{bmatrix}}_{L_{cam}} \begin{bmatrix} v_x \\ v_y \end{bmatrix} \end{aligned}$$

V. Experimental Results

In order to evaluate the performance of the tracking system, more or less crucial instrument poses during system operation have been taken. After presenting the results of the tracking, the compliance of the trocar during visual guidance of an instrument is shown.

The evaluation has been performed on a Intel Xeon QuadCore™2.4Ghz system. Images were taken and processed in real-time with full PAL resolution (768×576) from the framegrabber. As a first step, the precision of the instrument projection into image space, derived from the kinematic data, was verified (cf. Fig. 12, first image). The

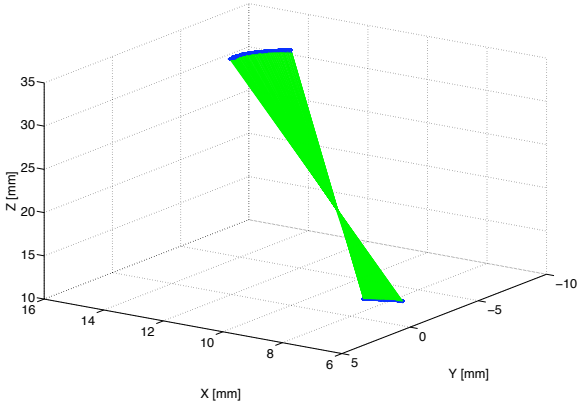


Figure 11. Evaluation of the trocar constraint by means of a magnetic tracking system

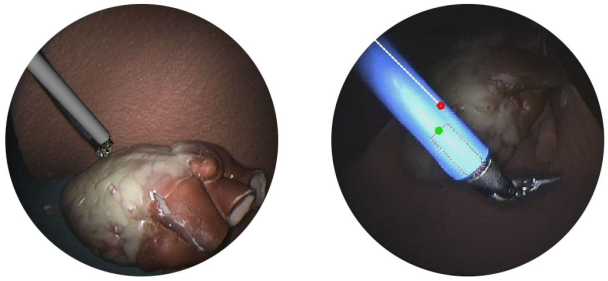


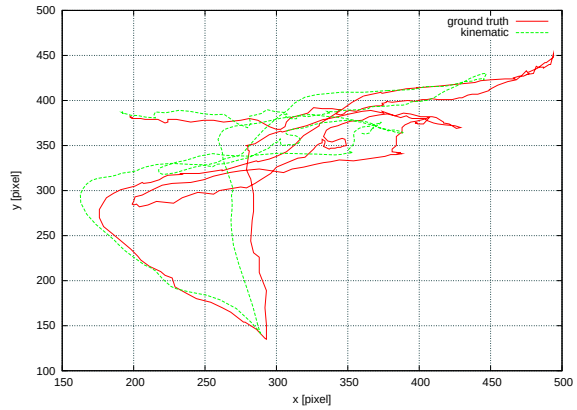
Figure 12. Left image: Initial estimation of the instrument pose, derived from sensor readings. Right image: Mismatch of the model scaling factor and the instrument due to strong specular reflections.

data is received by the tracking framework via network and applied to the CAD model of the instrument. To project the instrument pose into image space, a virtual camera is set up in a similar fashion, with position and orientation equal to the real endoscope. The projection of the shaft does not have to overlay the instrument that is to be tracked perfectly, but a good match supports a fast initialization of the tracking. A good agreement of projection and instrument helps to keep the normals of the sampled contour points smaller, making the tracking more robust and faster. The search length was determined experimentally.

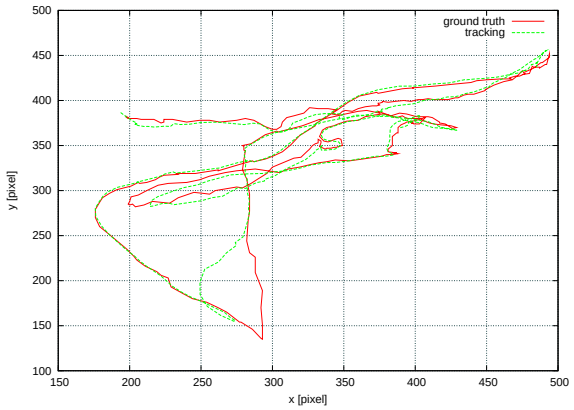
For evaluation purposes, a series of images was annotated by hand. Figure 8(a) shows the projection of the kinematic prediction into image space in comparison with the ground truth. The average distance error calculated over all frames is around 33 pixels. The offset between the estimation and the actual position of the instrument is well observable. The average distance between the tracked point and the ground truth could be reduced to 5.7 pixels. For the depicted image series, tracking was lost one time for a period of approximately 10 frames (lower left side in Fig. 8(b)). The plots of the tracking $x-$ and $y-$ errors (Fig. 8(c) and Fig. 8(d)) point out a fast reinitialization of the tracking around frame number 170. Excluding the 10 frames of the tracking loss, the accuracy can further be reduced to $4.6px$.

As already stated, the presented approach is not limited to a specific appearance of the instrument. In fact, we used three different instruments with blue, red and gray colored shafts. In our artificial environment, the background is very dark, since no brightened ribcage or abdominal wall limits the sight. Therewith, tracking the gray shaft is most challenging. While the detection of the shaft itself works flawless, more sliding of the model is observable, compared to the blue or red shaft. Since the color of the distal end of the instrument changes from gray to silver, no hard contour is given anymore. In this case, CCD loses tracking during fast movements.

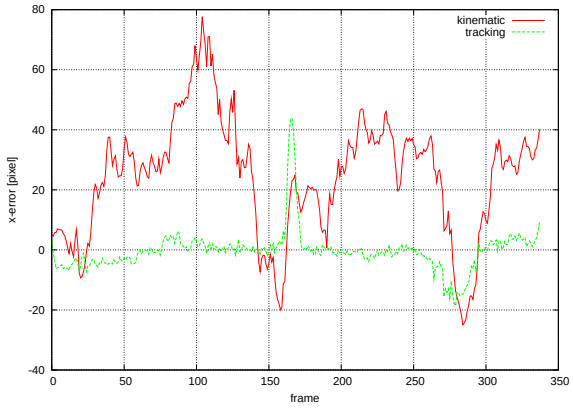
The main flaw of the proposed approach is the detection of the accurate scaling factor. Since the CCD algorithm seeks to maximize color statistics alongside of the contour



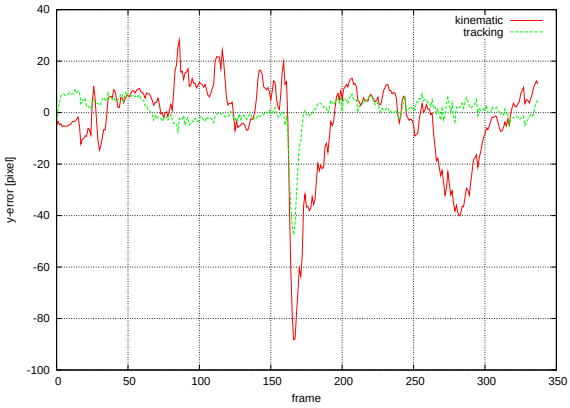
(a) Kinematic Prediction vs. ground truth in image space



(b) Tracking vs. ground truth in image space



(c) Errors compared to ground truth (x-axis)



(d) Errors compared to ground truth (y-axis)

Figure 8. **Tracking errors:** The ground truth data was annotated by hand. Figures 8(a) and Fig. 8(b) show the error of the kinematic prediction vs. the ground truth and the tracking in image space respectively. Figure 8(c) and Fig. 8(d) depict the errors in $x-$ and $y-$ direction in pixels, compared to the ground truth.

edges between model and background, strong specularities at the shaft can distort the measurement and are spuriously recognized as part of the instrument (cp. Fig. 12). Those kind of reflections especially appear at low distances (less than $3cm$) between the instrument and the light source, dependent on the present luminosity. Then, the center of the shaft can still be located accurately, but the distance to the tip is wrong.

Regarding the visual guidance, we evaluated the compliance with the trocar point, in addition to the experiments that have been performed in the original work. Therefore, we utilized a magnetic tracking system (Polhemus Liberty^{TM}). Since the magnetic markers were attached at the distal end of the instrument, the influence of the robot motors can be neglected. Figure 11 shows the movement of the instrument and the fulcrum.

VI. Conclusion and Future Works

This paper has explored the tracking of surgical instruments in minimally invasive surgery and its application to the visual guidance of the instruments and the endoscopic camera. Encoder readings from the robots were used to predict an approximated pose of the instrument in image space. The approximation is then used to (re-)initialize the image-based tracking, to set pose limits and to supervise a tracking loss. As modality, the Contracting Curve Density algorithm was used, which maximizes local color statistics collected at the model contour in order to separate it from the background.

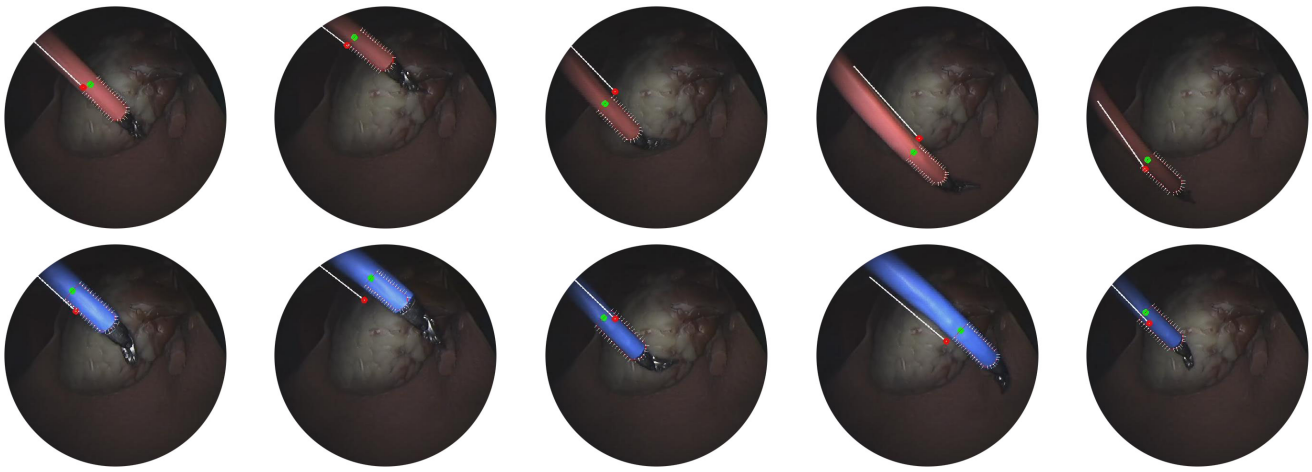


Figure 9. Top row: Tracking a reddish instrument. The tracking is very stable, even is the shaft color is similar to the background. The white line (ending in a red dot) indicates the projection of the kinematic prediction, the green dot is dedicated to the actually tracked position. The images are overlie with the instrument model and the contour normals used for sampling the local color statistics. Bottom row: Instrument tracking with a blue shaft. Since CCD does not employ a color or texture map of the instrument it can be applied to various shaft colors without changes.

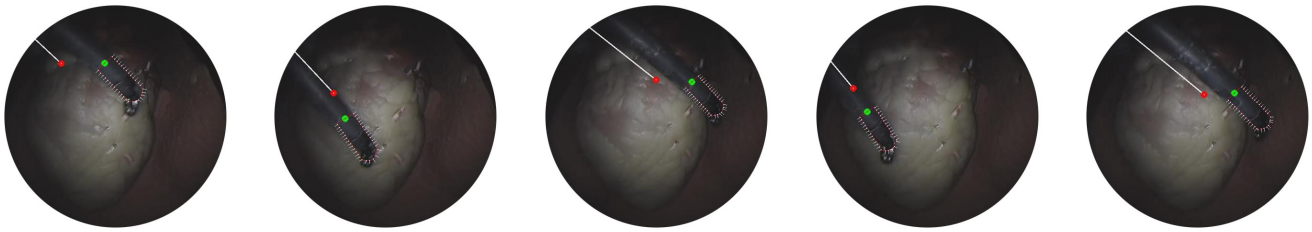


Figure 10. Tracking a grayish instrument shaft. As background, artificial skin and a heart model was used. In a real laparoscopic intervention the depth field would be more restricted, resulting in a brighter and more uniform illumination of the scene.

The performed experiments are very promising. Without the need for changing the model or program parameters, a blueish and a reddish instrument was tracked accurately. Problems in finding an adequate scaling factor can arise due to specular reflections, if the distance between instrument and light source is small. During a preprocessing step, this reflections could be removed, since their location and pixel values are well-known. In order to prevent a misplacement of the model at the distal end of the shaft, a non-uniform distribution of sampling normals could be introduced. After splitting up the model into an articulated model with two parts, one representing the shaft and one representing the rounded tip, the number of sampling points at each sub-model could be adjusted independently. As the statistics that are collected at the shaft would be weighted more than the statistics at the end, a shift could presumably be prevented. Also a simple blob detection that looks for the silver-colored forceps could be employed in addition.

ACKNOWLEDGMENT

This work is supported by the German Research Foundation (DFG) within the Collaborative Research Center SFB 453 on “High-Fidelity Telepresence and Teleaction”.

REFERENCES

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

[2] G. Guthart and J. Salisbury, J.K., “The intuitive^{TM}telesurgery system: overview and application,” in *Proceesings of the IEEE International Conference on Robotics and Automation*, vol. 1, 2000, pp. 618–621.

[3] H. Mayer, D. Burschka, A. Knoll, E. Braun, R. Lange, and R. Bauernschmitt, “Human-machine skill transfer extended by a scaffolding framework,” in *Proceesings of the IEEE International Conference on Robotics and Automation*, may 2008, pp. 2866 –2871.

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

[5] H. Wakamatsu, A. Tsumaya, E. Arai, and S. Hirai, "Manipulation planning for knotting/unknotting and tightly tying of deformable linear objects," in *Proceedings of the IEEE International Conference on Robotics and Automation*, April 2005, pp. 2505–2510.

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

[7] F. Nageotte, P. Zanne, C. Doignon, and M. de Mathelin, "Stitching planning in laparoscopic surgery: Towards robot-assisted suturing," *The International Journal of Robotics Research*, pp. 1303–1321, 2009.

[8] P. Hynes, G. Dodds, and A. Wilkinson, "Uncalibrated visual-servoing of a dual-arm robot for mis suturing," in *Proceedings of the IEEE/RAS-EMBS International Conference on Biomedical Robotics and Biomechatronics*, Feb. 2006, pp. 420–425.

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

[10] G.-Q. Wei, K. Arbter, and G. Hirzinger, "Real-time visual servoing for laparoscopic surgery. controlling robot motion with color image segmentation," *IEEE Engineering in Medicine and Biology Magazine*, vol. 16, no. 1, pp. 40–45, Jan.-Feb. 1997.

[11] A. Casals, J. Amat, and E. Laporte, "Automatic guidance of an assistant robot in laparoscopic surgery," in *Proceesings of the IEEE International Conference on Robotics and Automation*, vol. 1, Apr 1996, pp. 895–900.

[12] C. E. Reiley and G. D. Hager, "Task versus subtask surgical skill evaluation of robotic minimally invasive surgery," in *Proceedings of the International Conference on Medical Imaging and Computer Assisted Interventions*, 2009, pp. 435–442.

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

[14] D. R. Uecker, C. Lee, Y. F. Wang, and Y. Wang, "Automated instrument tracking in robotically-assisted laparoscopic surgery," *Journal of Image Guided Surgery*, vol. 1, pp. 308–325, 1998.

[15] C. Doignon, F. Nageotte, and M. de Mathelin, "The role of insertion points in the detection and positioning of instruments in laparoscopy for robotic tasks," in *Proceedings of the International Conference on Medical Imaging and Computer Assisted Interventions*, 2006, pp. 527–534.

[16] S. Voros, J.-A. Long, and P. Cinquin, "Automatic detection of instruments in laparoscopic images: A first step towards high-level command of robotic endoscopic holders," *The International Journal of Robotics Research*, vol. 26, no. 11-12, pp. 1173–1190, 2007.

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

[18] Z. Pezzementi, S. Voros, and G. D. Hager, "Articulated object tracking by rendering consistent appearance parts," in *Proceedings of the IEEE International Conference on Robotics and Automation*, May 2009, pp. 3940–3947.

[19] Y. Shiu and S. Ahmad, "Calibration of wrist-mounted robotic sensors by solving homogeneous transform equations of the form ax=xb," *IEEE Transactions on Robotics and Automation*, vol. 5, no. 1, pp. 16–29, Feb 1989.

[20] G. Welch and G. Bishop, "An introduction to the kalman filter," University of North Carolina at Chapel Hill, Department of Computer Science, Tech. Rep., 2006.

[21] G. Panin, A. Ladikos, and A. Knoll, "An efficient and robust real-time contour tracking system," *Computer Vision Systems, International Conference on*, vol. 0, p. 44, 2006.

[22] G. Panin, E. Roth, and A. Knoll, "Robust contour-based object tracking integrating color and edge likelihoods," in *VMV*, 2008, pp. 227–234.

[23] A. Ruf, M. Tonko, R. Horaud, and H.-H. Nagel, "Visual tracking of an end-effector by adaptive kinematic prediction," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 2, Sep 1997, pp. 893–899 vol.2.

[24] L. Itti and C. Koch, "Computational modelling of visual attention." *Nature reviews. Neuroscience*, vol. 2, no. 3, pp. 194–203, March 2001.

[25] B. Espiau, "Effect of camera calibration errors on visual servoing in robotics," in *The 3rd International Symposium on Experimental Robotics*. London, UK: Springer-Verlag, 1994, pp. 182–192.

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