Robust registration procedures for endoscopic imaging

W. Konen [1,*] , S. Tombrock [2], M. Scholz [3]

[1] Department of Informatics, University of Applied Sciences Cologne, 51643 Gummersbach, Germany

[2] Centre of Neuroinformatics at Ruhr University Bochum, 44801 Bochum, Germany

[3] Department of Neurosurgery, Ruhr-University Bochum, 44801 Bochum, Germany

Abstract

This paper presents a robust algorithm for calibration and system registration of endoscopic imaging devices. The system registration allows us to map accurately each point in the world coordinate system into the endoscope image and vice versa to obtain the world line of sight for each image pixel.

The key point of our system is a robust linear algorithm based on singular value decomposition (SVD) for estimating simultaneously two unknown coordinate transformations. We show that our algorithm is superior in terms of robustness and computing efficiency to iterative procedures based on Levenberg-Marquardt optimization or on quaternion approaches. The algorithm does not require the calibration pattern to be tracked.

Experimental results and simulations verify the robustness and usefulness of our approach. They give an accuracy of less than 0.7 mm and a success rate >99%. We apply the calibrated endoscope to the neurosurgical relevant case of red out, where in spite of the complete loss of vision the surgeon gets visual aids in the endoscope image at the actual position, allowing him/her to manoeuvre a coagulation fibre into the right position.

Finally we outline how our registration algorithm can be used also for standard registration applications (establish the mapping between two sets of points). We propose our algorithm as a linear, non-iterative algorithm also for projective transformations and for 2D-3D-mappings. Thus it can be seen as a generalization of the well-known Umeyama registration algorithm.

Keywords: registration, neuroendoscopy, visual navigation, markerless calibration, image processing.

Preprint of: W. Konen, M. Scholz, S. Tombrock: *Robust registration procedures for endoscopic imaging*, Medical Image Analysis, Volume 11, Issue 6, pp. 526-539, December 2007. http://dx.doi.org/10.1016/j.media.2007.04.006

---

[*] Corresponding author. Tel.: +49 2261 8196 275; fax: +49 2261 8196 15; konen@gm.fh-koeln.de. Postal adress: Prof. Dr. W. Konen, FH Köln, Am Sandberg 1, D 51643 Gummersbach, Germany.

1. Introduction

Image processing and image analysis play an important role in advanced surgery (CAS: computer aided surgery). Much work is devoted to 3D-reconstruction from CT-, MRI- or other volume data or to 3D-registration between different image modalities and the actual scene in the operating theatre. 3D-registration is an important prerequisite for navigation support systems which allow, for example, the surgeon to follow a predefined path during a surgical intervention.

Visual navigation in neuroendoscopy is a special kind of navigation employing only endoscopic images for navigational purposes. This technique can be utilized alone or in addition to conventional navigation systems based on magnetic resonance imaging (MRI) or computer tomography (CT) (Dorward et al., 1999; Rhode et al., 1998; Rhoten et al., 1997).

There is a large body of work in the area of virtual endoscopy (Bartz, 2005; Lemke et al., 2004) where radiological data (CT or MRI) are used to create virtual endoscopic views or movies, but those systems cannot react directly to observations made or situations happening in the operation theatre. An example is the case of red out, where a complete loss of vision in the endoscopic view due to bleeding occurs. With our system it is possible to give the surgeon precise visual aids in the endoscopic view to control a coagulation fibre (see Section 5.2) and stop the bleeding.

Digital image processing and image overlay *within (real) endoscopic images*, with the aim to provide the surgeon with navigational aids and/or measuring/tracking 3D-structures, has been addressed so far only by relatively few authors. Kosaka et al. (2000), Akatsuka et al. (2000) and later also Kawamata et al. (2002) and Shahidi et al. (2002) describe augmented reality systems in which previously recorded CT- or MRI-data (e.g. 3D-data of a tumour) can be overlaid to the actual endoscopic image. Camera calibration and system registration (physical-to-image-space registration) are vital building blocks for those systems. Koppel et al. (2002,2004) describe real-time tracking in endoscopic images which is used to estimate the "up" vector of the camera and to adjust the display for the surgeon. Camera calibration and measurement of the endoscope 3D-position is not part of the system.

The main roadblocks which prevent a wider range of applications for image processing and 3D-measurements within endoscopic images lie in our opinion (a) in the difficulty to provide robust algorithms for calibration and extraction of useful 3D-information from multiple images of a moving camera, (b) in strong distortions caused by the wide-angle endoscope lens system and (c) in close-to-real-time requirements for any of the image processing tasks.

We developed a new and robust algorithm for topic (a), the calibration and physical-to-image-space registration (system registration) of the endoscope. In this paper we describe this algorithm in detail and compare it to other algorithms and connect it to the somewhat different standard registration approaches, with the aim to share this knowledge with a wider audience and to bring endoscopic registration to more routine use in the operating theatre.

Topic (b) and (c) have been addressed in (Dey et al., 2002; Kawamata et al., 2002; Shahidi et al., 2002) as well as by us (Konen et al., 1997, 1998). We plan to describe more of our recent findings, laid out also in a patent application (Konen et al., 2006) in a follow-up paper (Konen et al., 2007).

Medical image registration is a very active research field due to its broad applicability to many imaging modalities; a recent overview is given by (Hill, 2001; Maintz and Viergever, 1998). The standard task of registration (or alignment) is to find the transformation T which transforms a set of points p_i as well as possible into a corresponding set of points q_i. In the case of rigid-body transformations the problem can be solved with methods from linear algebra, as it was first done by Green (1952), while Schoenemann (1966) was the first to use

singular value decomposition (SVD) for this problem. Later Arun et al (1987) rediscovered this method. Independently Farrell (1966) provided a solution that guaranteed a proper rotation, an idea which was rediscovered by Umeyama (1991). If the set of points is large, it may be too complex to set up the correspondence mapping manually. An automated method is the Iterative Closest Point (ICP) algorithm (Besl and McKay, 1992; Zhang, 1994), which can also be used to match a set of points to a free-form curve.

The problem of iterative algorithms is their starting value dependence which might result in a solution far away from the true solution. Koppel et al. (2002) describe a non-iterative, linear SVD-based solution, however for a somewhat different endoscope application where no positioning measurement system for the endoscope is used. With such a positioning system, the system registration contains 4 or 5 coordinate transformations, where 2 or 3 of them are unknown and have to be estimated simultaneously. The works of (Kosaka et al., 2000, Akatsuka et al., 2000; Kawamata et al., 2002) incorporate such a system registration, but do not describe which methods (iterative nonlinear vs. direct linear) are used. Shahidi et al. (2002) use a second tracking unit for a separately registered calibration pattern. Schwald et al. (2004) describe a linear method for solving the registration with 4 coordinate systems, which requires however a nonlinear quaternion mapping to ensure orthogonality of the rotation matrices. In this paper we describe a new direct linear method for system registration and compare it to other iterative possibilities, including a quaternion approach.

This paper is organized as follows: In Sec. 2 we will first present our system setup and formulate the general registration problem for endoscopic imaging. We will then develop in Sec. 3 different algorithms for solving the physical-to-image-space registration (some mathematical details are deferred to Appendix A and B). Sec. 4 presents results both from simulations and real registration experiments to verify the robustness and usefulness of our method, while Sec. 5 discusses the applicability to other registration task and shows some endoscopy medical applications.

2. The Registration Problem

2.1. System Setup

The rigid endoscope (outer diameter 5.9 mm, Camaert/Wolf GmbH) used in this work consists of a circular tube (6 mm diameter) where a colour CCD-camera at the rear end captures the image from the tip of the endoscope through a special lens system (distance tip - rear end: 380 mm). We developed a special device (see Konen et al. (1997) and Figure 1) mounted on the shaft of the endoscope which holds 3 infrared LEDs (light-emitting diodes).

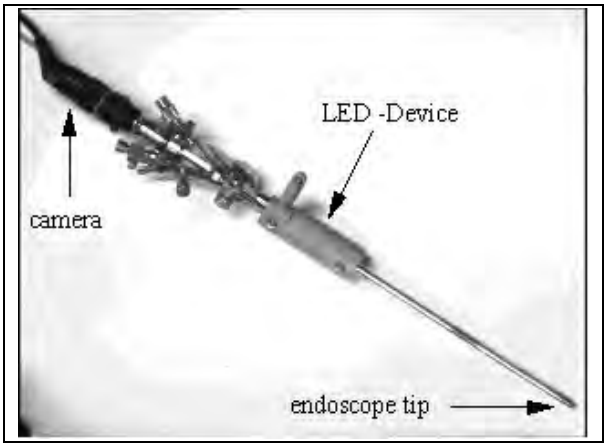

Figure 1: Rigid endoscope with LED-device

The positions of the LEDs are measured continuously by the OPMS (optical positioning measurement system) which is a part of the EasyGuide™ Neuro navigation system. (Philips Medical Systems, Best, The Netherlands). The OPMS basically consists of a stereo camera rig, stationary in the operating theatre. The system measures the 3D-position of the LEDs and determines the 6 degrees of freedom of the rigid endoscope in the coordinate system of the camera rig. The OPMS achieves a differential spatial resolution of 0.4-0.8 mm and an overall accuracy of 1-1.5 mm within its volume of operation.

The endoscope images were continuously registered at a frequency of 8 images per second, digitized on a frame grabber and stored as

a computerized dataset including the OPMS position data of the endoscope. We have reported results with this system concerning virtual image navigation (Scholz et al., 2000) and ergonomic aspects (Scholz et al., 2005, 2005b).

2.2. Camera Calibration

We briefly review our camera calibration procedure, which has been described in more detail in Konen et al. (1997, 1998).

The basic camera calibration procedure estimates 11 or 12 parameters: the 6 extrinsic parameters {**R, t**} which map a point from the world coordinate system into the camera coordinate system, the 4 intrinsic parameters of a camera (focal length f, piercing point (u,v), pixel scaling factor s_x) and finally 1 or 2 image distortion parameters (κ_1 or κ_1, κ_2). When the camera is moved, only the extrinsic parameters change, while the intrinsic and the distortion parameters remain constant.

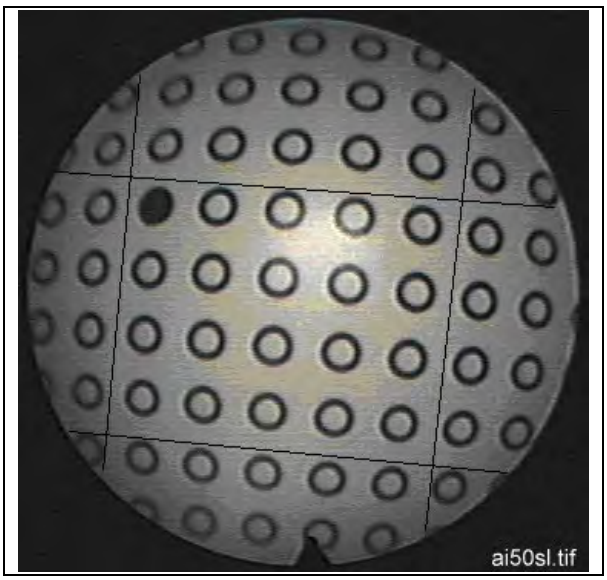

Figure 2: The calibration pattern as viewed through the endoscope (distorted image). The straight lines are not part of the image but superimposed to show the distortion effects.

We use the well-known camera calibration procedure from Tsai (1987) which provides a versatile and robust estimation of the camera parameters. As a calibration pattern we use a plane of rings placed on a regular lattice with 1.25 mm ring-to-ring distance (Figure 2). The algorithm of Tsai (1987) [31] is able to compute first order radial distortion correction (κ_1) which is highly necessary for the endoscope camera as Figure 2 shows. In Konen et al. (1997, 1998) also the inclusion of a higher order κ_2-term was tested, but the improvements were found to be negligible for an endoscope camera system. Thus our initial full camera calibration procedure delivers 11 parameters in total. Subsequent extrinsic camera calibrations as they will be used in Sec. 2.3 deliver only the 6 extrinsic parameters, they assume the intrinsic and distortion parameters to be the same. The overall calibration accuracy is better than 0.2 mm over the whole field of view.

2.3. Physical to Image Space Registration

Figure 3 depicts schematically the 4 different coordinate systems involved in our setup. Two of them, **W** and **O**, are at rest in the operating theatre: The **world coordinate system W** is attached to the patient (during surgery) or to the calibration pattern (during the registration process); the **OPMS coordinate system O** is attached to the stereo camera rig (Sec. 2.1). Obviously the relation between both coordinate systems remains fixed as long as stereo camera rig and calibration pattern (patient) are not moved, a prerequisite we assume to be valid during the registration process (and later during surgery).

Then there are two other coordinate systems which are usually moving in the operating theatre: The **endoscope coordinate system E**, whose origin is attached to the LED device (Sec. 2.1) and the **camera coordinate system C**, whose origin is attached to the camera centre and whose z-axis is pointing along the optical axis of the camera. Again, both coordinate systems are connected by a fixed relation, i.e. they move together – at least as long as nei-

ther the camera or the LED device is detached from the endoscopic shaft. But the relation is initially unknown.

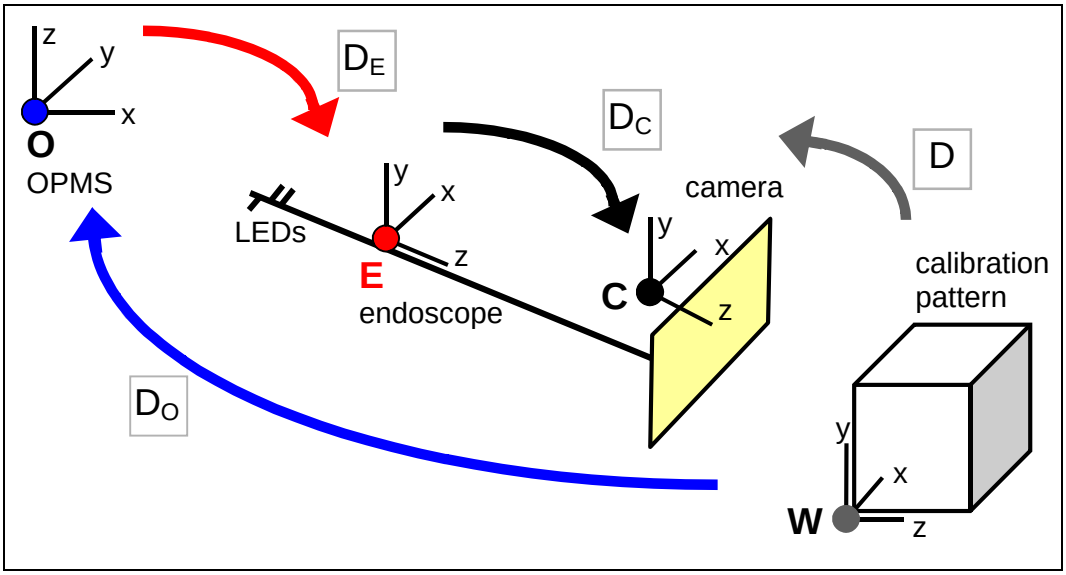

Figure 3: Coordinate systems for the system registration. The transform D is equivalent to the concatenated transform $D_C D_E D_O$.

Points in different Cartesian coordinate systems can be mapped into each other by a rotation followed by a translation (rigid body transform, a special case of affine transform). That is, if $\mathbf{r}_W$ is a point in **W**, represented by a 3x1 column vector, then its representation in **O** is given by

$$(1) \quad \mathbf{r}_O = \mathbf{R}_O \mathbf{r}_W + \mathbf{t}_O \equiv \mathbf{D}_O(\mathbf{r}_W)$$

where $\mathbf{R}_O$ is a 3x3 rotation matrix, $\mathbf{t}_O$ is the 3x1 translation vector (pointing from the **O**-origin to the **W**-origin) and $\mathbf{D}_O$ is just an abbreviation for this affine transformation. Likewise, the other coordinate transformations can be written as

$$(2) \quad \mathbf{r}_E = \mathbf{R}_E \mathbf{r}_O + \mathbf{t}_E \equiv \mathbf{D}_E(\mathbf{r}_O)$$

to map the **O**-representation into the **E**-representation and

$$(3) \quad \mathbf{r}_C = \mathbf{R}_C \mathbf{r}_E + \mathbf{t}_C \equiv \mathbf{D}_C(\mathbf{r}_E)$$

to map the **E**-representation into the **C**-representation. Note that if $\mathbf{r}_C=(x,y,z)^T$ is a point in the camera coordinate system **C**, then it will appear at location ((x/z)·f, (y/z)·f) on the camera target at focal length f, as follows from simple projective geometry. Likewise, if (X,Y) is a position on the camera target, it corresponds to 3D-points $z\cdot(X/f, Y/f, 1)^T$ for any depth value z.

To complete our transformation ring, the affine transformation

$$(4) \quad \mathbf{r}_C = \mathbf{R}\mathbf{r}_W + \mathbf{t} \equiv \mathbf{D}(\mathbf{r}_W)$$

maps a world point $\mathbf{r}_W$ directly into the **C**-representation (see Figure 3). Obviously this is equivalent to the concatenation of the above three transformations, i.e.

$$(5) \quad \mathbf{D}(\cdot) = \mathbf{D}_C(\mathbf{D}_E(\mathbf{D}_O(\cdot)))$$

Our goal is to be able to transform 3D world points into the camera coordinate system or vice versa to map points from the camera view into the corresponding lines of sight in the world coordinate system. The 3D world points are usually expressed in the OPMS coordinate system **O**. The 3D points may come for example from preoperative CT-data which have been transformed into the OPMS coordinate system. We have shown above all the equations

needed to transform between these coordinate systems, but unfortunately most of the parameters (rotation matrices, translation vectors) are unknown so far. It is the purpose of the system registration process, also termed "physical to image space registration", to find these parameters.

More specifically we note:

- $\mathbf{D_E}$ is known for each time instance, since the data ($\mathbf{R_E}$,$\mathbf{t_E}$) are delivered by the OPMS.
- $\mathbf{D_C}$ is the unknown transformation of primary interest, since it allows together with $\mathbf{D_E}$ to transform points from the camera coordinate system into **O**, a coordinate system at rest in the operating theatre. $\mathbf{D_C}$ is constant in time[1], thus it can be estimated beforehand in the registration process and then used during surgery.
- **D** and $\mathbf{D_O}$ are unknown, but only needed as auxiliary variables during system registration to find $\mathbf{D_C}$. (Future applications might also use $\mathbf{D_O}$, which is constant in time as long as the stereo camera rig and the calibration pattern (patient) are not moved, but currently we use only $\mathbf{D_E}$ and $\mathbf{D_C}$ in our applications.)

The important point is that the parameters of **D**, i.e. the rotation matrix **R** and the translation vector **t**, can be obtained by observing with the camera a calibration pattern of known geometry. This is the process of *extrinsic camera calibration* (Sec. 2.2) as it is described in more detail in Tsai (1987). Basically it means that a camera picture of the known calibration pattern and the known intrinsic properties of the camera are sufficient to determine *where* the camera is in relation to the calibration pattern (world) coordinate system **W**.

This gives us the following strategy for physical-to-image-space registration: The extrinsic camera calibration from Sec. 2.2 is used to get the transformation **D** from the calibration pattern to the camera system (Figure 3). The transformation $\mathbf{D}_E$ between the OPMS and the LEDs of the endoscope is delivered by the OPMS. Now we use the fact of Eq. (5) that a point in the calibration pattern coordinate system **W** can be transformed into the camera coordinate system **C** either by **D** or by the combined transformation $\mathbf{D_C D}_E \mathbf{D_O}$. The system registration process now proceeds as follows:

- Put the calibration pattern at a fixed point in space. That is, the transformation $\mathbf{D}_O$ between calibration pattern and the camera rig of the OPMS remains constant during registration.
- Take images i=1,...,n of the calibration pattern with the endoscope from different endoscopic viewpoints. Obtain the transformations $\mathbf{D}^{(i)}$ (extrinsic camera calibration) and $\mathbf{D}_E^{(i)}$ (OPMS) for each viewpoint.
- For each measurement the functional identity of Eq. (5) holds. This gives us a system of functional equations

$$\begin{aligned} \mathbf{D}^{(1)}(\cdot) &= \mathbf{D}_C\left(\mathbf{D}_E{}^{(1)}\left(\mathbf{D}_O(\cdot)\right)\right) \\ &\vdots \qquad\qquad \vdots \\ \mathbf{D}^{(n)}(\cdot) &= \mathbf{D}_C\left(\mathbf{D}_E{}^{(n)}\left(\mathbf{D}_O(\cdot)\right)\right) \end{aligned} \tag{6}$$

  which we need to solve for the unknown transformations $\mathbf{D}_O$ and $\mathbf{D}_C$.

We will describe in Sec. 3.1 how this set of functional equations (6) can be reduced quite easily to a set of usual equations. Also, Sec. 3 will give different methods to solve these equations in more detail.

To summarize we have outlined in this section a method to determine the unknown, constant transformations $\mathbf{D}_C$ (and $\mathbf{D}_O$) in a registration process which needs to be done only once before the surgical intervention. Thus all potentially time-consuming measurements and calcu-

[1] as long as the connections between LED device, camera and endoscope remain tight

lations can be done before surgery. During surgery we only use the constant $\mathbf{D}_C$ and the time-varying $\mathbf{D}_E$ (delivered by the OPMS) to map points from **C** to **O** and vice versa – a calculation which can be done in real time for many points.

2.4. Difference from Usual Registration Procedure

Our problem is thus somewhat different from the usual medical image registration procedures (e.g. the orthogonal Procrustes problem, see Hill (2001) for an overview), and the difference can be traced back to the fact that we perform here a markerless registration. In usual registration procedures (e.g. from CT system to the intraoperative system IO), a set of fiducial points (the markers) is traced in both systems, giving two sets of points $\{\mathbf{r}_{CT}^{(i)}|i=1,..,N\}$ and $\{\mathbf{r}_{IO}^{(i)}|i=1,..,N\}$. The goal is then to find the transformation $\mathbf{D}_{CT}$ which minimizes

$$\chi^2 = \sum_i \left\| \mathbf{r}_{CT}{}^{(i)} - \mathbf{D}_{CT}\left(\mathbf{r}_{IO}{}^{(i)}\right)\right\|^2$$

Many solutions to this problem exist in the literature, the most well-known of them being the algorithms by Arun (1987) and Umeyama (1991) which are rediscoveries of solutions published earlier by Green (1952), Schoenemann (1966) and Farrell et al. (1966). In our case we map calibration points $\mathbf{r}_C^{(i)} = \mathbf{D_C}(\mathbf{D}_E^{(i)}(\mathbf{r}_O)) = \mathbf{D_C}(\mathbf{r}_E^{(i)})$, but we do not know during the registration process, what the coordinates $\mathbf{r}_O$ are, we only know that some fixed transformation $\mathbf{r}_O = \mathbf{D_O}(\mathbf{r}_W)$ relates them to the world coordinate system of the calibration pattern. Contact to the above notation can be made by

$$\chi^2 = \sum_i \left\| \mathbf{r}_C{}^{(i)} - \mathbf{D}_C\left(\mathbf{r}_E{}^{(i)}\right)\right\|^2$$

but the difference is that the vectors $\mathbf{r}_E^{(i)} = \mathbf{D}_E^{(i)}(\mathbf{D_O}(\mathbf{r}_W))$ are themselves subject to an optimization in the unknown $\mathbf{D_O}$. This would be different if the calibration points (which are all the rings in the calibration pattern) were tracked by the OPMS: then we had for each calibration point a known vector $\mathbf{r}_O$. But there are 20-50 rings needed for the camera calibration. Thus to know each vector $\mathbf{r}_O$ it would required to build a calibration pattern containing 20-50 LEDs. These LEDs were measured simultaneously by the OPMS and the camera. But such a device is difficult to build and many OPMS can not handle the measurement of 20-50 LEDs simultaneously. In an alternative aproach, Shahidi (2002) has mounted *one* separate tracking unit on the calibration pattern. But then there still remains the problem of establishing the transformation between the grid of calibration rings and the tracking unit, a system registration in its own.

We therefore stay with the system of functional equations (6) and seek efficient methods to solve it in the following section.

3. Methods

A robust solution of Eq. (6) which works well in practical surroundings is not an easy task due to the following factors

- many variables (2x12 = 24 variables, 9 rotation matrix elements and 3 translation vector elements in both $\mathbf{D}_O$ and $\mathbf{D}_C$)
- *nonlinear* optimization, since the elements of $\mathbf{D}_O$ and $\mathbf{D}_C$ are coupled
- *constrained* optimization problem, since there are only 2x6 = 12 independent degrees of freedom within the 24 variables.

We show in Sec. 3.1 how to decompose Eq. (6) from a nonlinear problem with 24 unknowns into a linear 6-variable problem plus a nonlinear 18-variable problem. To solve the latter we show in Sec. 3.2 an iterative, unconstrained solution method, while in Sec. 3.3 we use qua-

ternions representation to get another iterative method with fewer variables. However, all these methods have the drawback that a good starting value is needed in order to obtain a solution, a severe drawback for a registration procedure which should run ideally fully-automated to the best solution. A new method from linear algebra (Sec. 3.4) shows that the problem can be solved by linear methods without iterations and thus without need for a starting value.

3.1. Decoupling translation from rotation

If the functional Eq. (6) is valid for any r_W, namely

$$\mathbf{D}^{(i)}(\mathbf{r}_W) = \mathbf{D}_C\left(\mathbf{D}_E{}^{(1)}(\mathbf{D}_O(\mathbf{r}_W))\right), \tag{7}$$

then the coefficient multiplying r_W and the constant term have to be the same on both sides of the equation. By substituting Eqs. (1)-(4) into Eq. (7) and comparing terms we obtain for i=1,...,n endoscopic viewpoints:

$$\mathbf{R}^{(i)} = \mathbf{R}_C\mathbf{R}_E^{(i)}\mathbf{R}_O \tag{8}$$

$$\mathbf{t}^{(i)} = \mathbf{t}_C + \mathbf{R}_C\mathbf{t}_E^{(i)} + \mathbf{R}_C\mathbf{R}_E^{(i)}\mathbf{t}_O \tag{9}$$

If all rotation matrices are known, we get the unknown translation vectors t_C and t_O from rearranging terms in Eq. (9)

$$\mathbf{t}^{(i)} - \mathbf{R}_C\mathbf{t}_E^{(i)} = \left(\mathbf{1} \;\; \mathbf{R}_C\mathbf{R}_E^{(i)}\right)\begin{pmatrix}\mathbf{t}_C\\ \mathbf{t}_O\end{pmatrix} \tag{10}$$

This is one of n linear equation systems with 3 equations in 6 unknowns $\mathbf{x}=(\mathbf{t_C},\mathbf{t_O})$. For n>2 we seek a solution for 3n equations in the least square sense

$$\chi^2 = \|\mathbf{Lx} - \mathbf{b}\|^2 = Min. \quad \text{with}$$

$$\mathbf{L} = \begin{pmatrix}\mathbf{1} & \mathbf{R}_C\mathbf{R}_E^{(1)}\\ \vdots & \vdots\\ \mathbf{1} & \mathbf{R}_C\mathbf{R}_E^{(n)}\end{pmatrix}, \quad \mathbf{b} = \begin{pmatrix}\mathbf{t}^{(1)} - \mathbf{R}_C\mathbf{t}_E^{(1)}\\ \vdots\\ \mathbf{t}^{(n)} - \mathbf{R}_C\mathbf{t}_E^{(n)}\end{pmatrix} \tag{11}$$

A robust solution for such a minimization problem is the well-known singular value decomposition (SVD) (see Press et al. (1992) for a good description)

Thus only the **reduced problem** in the rotation matrices

$$\begin{aligned}\mathbf{R}^{(1)} &= \mathbf{R}_C\mathbf{R}_E^{(1)}\mathbf{R}_O\\ &\vdots \qquad\qquad \vdots\\ \mathbf{R}^{(n)} &= \mathbf{R}_C\mathbf{R}_E^{(n)}\mathbf{R}_O\end{aligned} \tag{12}$$

(i=1,...,n endoscopic viewpoints) remains to be solved. This is a problem with 9n equations for 18 unknowns (the matrix elements of $\mathbf{R}_O$ and $\mathbf{R}_C$), but only 6 independent degrees of freedom. Once we have a solution for Eq. (12), the solution to the full problem is easily achieved with the help of Eq. (11).

In order to simplify our notation, we will use from now on the following abbreviations:

- $\mathbf{A}^{(i)} \equiv \mathbf{R}^{(i)}$ and $\mathbf{B}^{(i)} \equiv \mathbf{R}_E^{(i)}$ for the known matrices (i=1,...,n measurements),

o $\mathbf{X} \equiv \mathbf{R}_C$ and $\mathbf{Y} \equiv \mathbf{R}_O$ for the unknown matrices.

3.2. Iterative nonlinear optimization

A standard method for solving the nonlinear Eq. (12) is to transform it to a least square problem

$$(13)\quad \chi^2 = \sum_i \left\| \mathbf{A}^{(i)} - \mathbf{X}\mathbf{B}^{(i)}\mathbf{Y} \right\|^2 = \mathrm{Min.} \quad \text{with} \qquad \|\mathbf{M}\|^2 = \sum_{a,b} \mathrm{M}_{ab}{}^2$$

and solve for the unknowns **X** and **Y** by Levenberg-Marquardt optimization, a standard least square optimization technique (Press et al., 1992). This iterative method has a fast convergence on many nonlinear optimization problems, however it needs a good starting point and is not likely to escape from local minima.

Applied to our problem, the results in Sec. 4.1 show, that in about 25-35% of all cases the algorithm fails to find the global solution and gets stuck in a local minimum far away from the true solution, a clearly unacceptable result for a robust registration procedure. How can we improve on this?

An obvious criticism to the above Eq. (13) is that it does not include the orthogonality constraints which we have to impose on the matrices **X** and **Y** in order to make them proper rotation matrices. Therefore the optimization problem has more variables than necessary and is therefore more likely to get trapped in local minima.

3.3. Iterative quaternion optimization

A way to deal with the constraints is to change the representation of rotations: Instead of working with rotation *matrices* we can also work with **unit quaternions**. Unit quaternions are 4-dimensional vectors with norm 1 and they can be used to represent rotations. The advantage is that they have only 4 variables (with one constraint) instead of 9 variables (with 6 constraints), so the unconstrained optimization space is much smaller in the quaternion case.

The basic properties of quaternions are reviewed in Appendix A, showing that the quaternion vector product a**.**b is basically equivalent to the concatenation of rotation matrices. Given quaternions $a^{(i)}$, $b^{(i)}$, x, y corresponding to the matrices $\mathbf{A}^{(i)}$, $\mathbf{B}^{(i)}$, **X**, **Y**, resp., and given the quaternion representation **R**(q) of a rotation matrix, Eq. (19), we can reformulate Eq. (13) in the form

$$\chi^2 = \sum_i \left\| \mathbf{R}(a^{(i)}) - \mathbf{R}(x)\mathbf{R}(b^{(i)})\mathbf{R}(y) \right\|^2 = Min.$$

and solve this again by Levenberg-Marquardt optimization.[2] The derivatives needed for Levenberg-Marquardt are given by (x_q is the q-th component of quaternion x)

[2] Why do we not solve an optimization problem $\chi^2 = \sum_i \left\| \mathrm{a}^{(i)} - \mathrm{x}.\mathrm{b}^{(i)}.\mathrm{y} \right\|^2 = \mathrm{Min.}$ in the quaternion space directly? – Because the twofold ambiguity of quaternions makes the optimization functional discontinuous which leads to complications during iterative optimization.

$$\frac{\partial}{\partial x_q}\left(\mathbf{R}(x)\mathbf{R}(b^{(i)})\mathbf{R}(y)\right) = \frac{\partial \mathbf{R}(x)}{\partial x_q}\mathbf{R}(b^{(i)})\mathbf{R}(y)$$

$$\frac{\partial}{\partial y_q}\left(\mathbf{R}(x)\mathbf{R}(b^{(i)})\mathbf{R}(y)\right) = \mathbf{R}(x)\mathbf{R}(b^{(i)})\frac{\partial \mathbf{R}(y)}{\partial y_q}$$

Again we need a starting value for x and y. Although one might think that a lower dimensional optimization space (8 instead of 18 dimensions) should increase the success rate, the results in Sec 4.1 will show an even lower success rate of about 50%.

3.4. Direct linear algorithm

The question arises whether there is a way to avoid the nonlinearities altogether when solving Eq. (12). We can use to our advantage the fact that the true solution {**X**,**Y**} consists of two orthogonal matrices, i.e. $\mathbf{X}^T\mathbf{X} = \mathbf{Y}^T\mathbf{Y} = \mathbf{1}$, and thus we can rewrite Eq. (12) in the form

$$\begin{array}{l} \mathbf{A}^{(1)}\mathbf{Y}^{\mathrm{T}} - \mathbf{X}\mathbf{B}^{(1)} = 0 \\ \quad \vdots \qquad \vdots \qquad \vdots \\ \mathbf{A}^{(\mathrm{n})}\mathbf{Y}^{\mathrm{T}} - \mathbf{X}\mathbf{B}^{(\mathrm{n})} = 0 \end{array} \tag{14}$$

This can be transformed to a linear system

$$\begin{array}{l} \mathbf{C}\mathbf{z} = 0 \quad \text{with} \quad (9\mathrm{n}\times 18)\ \text{matrix}\ \mathbf{C}\ \text{and} \\ \mathbf{z}^{\mathrm{T}} = \left(\mathrm{x}_{11} \quad \mathrm{x}_{12} \quad \cdots \quad \mathrm{x}_{33} \quad \mathrm{y}_{11} \quad \cdots \quad \mathrm{y}_{33}\right) \end{array} \tag{15}$$

where the 18-element vector **z** is the concatenation of all rows from **X** and **Y**. The structure of matrix **C** is shown in Appendix B. This homogeneous system has of course the trivial solution **z**=0, but we are seeking non-trivial solutions from the nullspace of **C** with boundary condition $||\mathbf{z}||^2=6$. This necessary (but not sufficient) boundary condition is due to the fact that the true solution should represent two orthogonal matrices, i.e. **z** should be the concatenation of six 3D vectors, each having norm 1.

Each non-trivial solution is a vector in the nullspace of matrix **C**. A basis of the nullspace is readily found by SVD which decomposes

$$\mathbf{C} = \mathbf{U}\cdot\mathrm{diag}(\mathrm{w}_{\mathrm{j}})\cdot\mathbf{V}^{\mathrm{T}} \tag{16}$$

with orthogonal matrices **U** and **V**. It is shown in Press et al. (1992) that the columns j in matrix **V** with their $w_j = 0$ span the nullspace of **C**. We can distinguish three cases:

1. no $w_j=0$, the nullspace has dimension zero: then the measurements are inconsistent and do not allow a registration solution
2. exactly one $w_j=0$, the nullspace has dimension one: this should be the normal case for a sufficient number of measurements
3. more than one $w_j=0$, the nullspace has dimension two or more: there is no unique solution to the problem, this happens usually when there are too few ($n\leq 3$) or too redundant measurements.[3]

In the case of real measurements there will be noise, which prevents Eq. (15) from having exact solutions (beside the trivial one), thus no w_j is exactly zero. However, there will be in

[3] Measurements are *redundant* if the endoscope looks at the calibration pattern from virtually the same viewing angle. We found out in our experiments that changing the viewing angle by about 15^0 to 30^0 is usually good enough to have non-redundant measurements.

case 2 one linear independent solution **z** having $\|\mathbf{Cz}\|$ much smaller than for other **z** of the same length, corresponding to a w_j which is much smaller than all other w_i. Thus SVD robustly solves the problem in the best possible way: It returns that solution among all **z** with $||\mathbf{z}||^2=6$ which fulfils $\|\mathbf{Cz}\| = \mathrm{Min.}$ [4]. We refer to this solution somewhat incorrectly as being in the "nullspace" of **C**, although **C** has strictly speaking no nullspace but only a space with $\|\mathbf{Cz}\|$ (much) smaller than in any other direction. We give in Sec. 4.1 an empirical definition of this approximate "nullspace" for our case.

One might think at first glance, that a non-trivial solution **z** in Eq. (15) is very unlikely to fulfil the constraints of orthogonal matrices since they are not incorporated in Eq. (15), but the results in Sec. 4.1 show, that the contrary is usually the case: If we are in case 2 (nullspace has dimension 1, the normal case), then the solutions with the right length $||\mathbf{z}||^2=6$ will automatically fulfil $\mathbf{X}^T\mathbf{X}=\mathbf{Y}^T\mathbf{Y}=\mathbf{1}$.

The following argument shows that such a behaviour can indeed be expected: If we are given a set of consistent measurements, it means that they must be in agreement with a certain (but unknown) solution $\{\mathbf{X}^{\mathbf{true}},\mathbf{Y}^{\mathbf{true}}\}$. Thus $\{\mathbf{X}^{\mathbf{true}},\mathbf{Y}^{\mathbf{true}}\}$ is in the nullspace of **C**. *If* the nullspace is only 1-dimensional (case 2), then there is only (up to a minus sign) *one* vector **z** with $||\mathbf{z}||^2=6$, and it must therefore coincide with the set of orthogonal matrices $\{\mathbf{X}^{\mathbf{true}},\mathbf{Y}^{\mathbf{true}}\}$.

To summarize, we have the following **direct linear algorithm**:

1. Set up matrix **C** acc. to Eqs. (15) and (21)
2. Make its SVD decomposition acc. to Eq. (16) and pick the column $\mathbf{v}_j$ of matrix **V** corresponding to the smallest singular value w_j.
3. Set $\mathbf{z} = \sqrt{6}\mathbf{v}_j$ and extract from **z** the solution matrices {**X**,**Y**}. Check for the right sign of det(**X**) and det(**Y**).

It has the big advantage *that it does not need a starting value*, which means that the solution – if it exists – will be found with a success rate of 100%.

4. Results

4.1. Comparing the algorithms

We want to compare the 3 different algorithms of Sec. 3.2, 3.3 and 3.4 with respect to the following factors:

- robustness against start value variation
- robustness against measurement noise
- quality improvement with more measurements
- performance

We name the three algorithms of Sec. 3.2, 3.3 and 3.4 as "iterative", "quaternion" and "direct linear", resp., keeping in mind that the "quaternion" algorithm is of course also an iterative algorithm.

---

[4] Proof: If $\mathbf{v}_j$ is the column of **V** corresponding to the smallest singular value w_j of **C**, then $\mathbf{z} = \sqrt{6}\mathbf{v}_j$ is the desired solution, since $\|\mathbf{Cz}\|^2 = \left\|\mathbf{U}\,diag(w_i)\,\mathbf{V}^T\sqrt{6}\mathbf{v}_j\right\|^2 = 6\left\|\mathbf{U}_j w_j\right\|^2 = 6w_j\left\|\mathbf{U}_j\right\|^2 = 6w_j$ is smaller than $||\mathbf{Cz}||^2$ for any other **z** with the same length.

The evaluation procedure is as follows: We configure a registration problem by choosing rotation matrices $\mathbf{B}^{(i),true}$, $\mathbf{X}^{true}$,$\mathbf{Y}^{true}$ and calculating $\mathbf{A}^{(i),true}= \mathbf{X}^{true}\cdot\mathbf{B}^{(i),true}\cdot\mathbf{Y}^{true}$, where i=1,...,n denotes the number of measurements. To simulate the effects of measurement noise, we perturb the measurement matrices with $\mathbf{A}^{(i)}= \mathbf{A}^{(i),true}\cdot\boldsymbol{\varepsilon}(\varphi,\theta,\psi)$ where each $\boldsymbol{\varepsilon}(\varphi,\theta,\psi)$ is an independently drawn random rotation matrix. We obtain $\boldsymbol{\varepsilon}(\varphi,\theta,\psi)$ by drawing Euler angles φ,θ,ψ independently from the normal distribution $N(0,\sigma_{noise})$. The perturbed measurement has a slightly different true solution $\{\mathbf{X}^{true},\mathbf{Y}^{true}\}$ which is found by a quick least-square optimization of

$$\chi^2 = \sum_i \left\| \mathbf{X}^{true}\mathbf{B}^{(i)}\mathbf{Y}^{true} - \mathbf{A}^{(i)} \right\|^2$$

given the unperturbed $\{\mathbf{X}^{true},\mathbf{Y}^{true}\}$ as starting point.

For the iterative algorithms we choose also for each run starting values $\mathbf{X}^{(0)}$, $\mathbf{Y}^{(0)}$, which are independently drawn random rotation matrices (with flat distribution in the Euler angles).

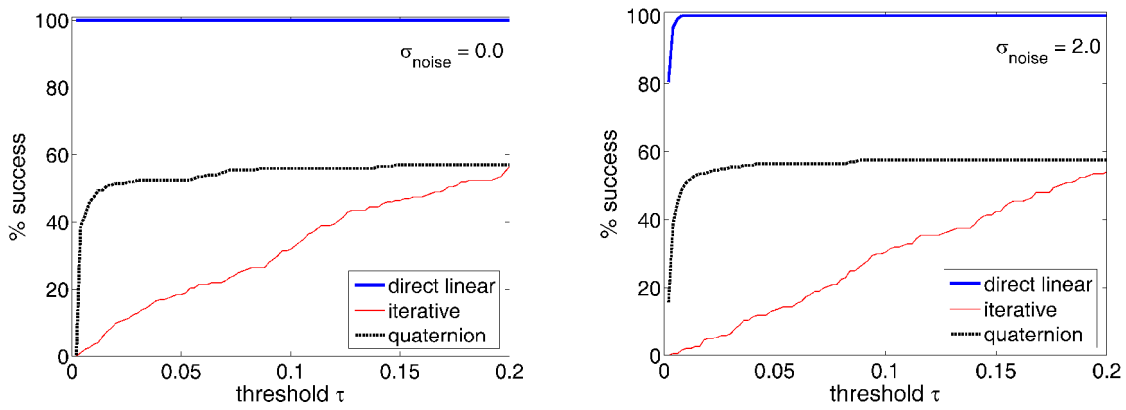

Figure 4: Robustness of the three different registration algorithms. The percentage of successes is shown as a function of threshold τ, the maximum allowable deviation between solution vectors and true vectors, see Eq. (17). Left: noise-free measurements, right: measurements with angular noise standard deviation $\sigma_{noise}=2^o$. Based on 200 simulation runs, each with n=4 measurements. For the two iterative algorithms a random start value is also chosen for each run. It is clearly seen that the direct linear algorithm outperforms the two iterative algorithms.

Then we pass the input $\{\mathbf{A}^{(i)}, \mathbf{B}^{(i)} | i=1,...,n\}$ into each of the three algorithms[5] (plus the start matrices $\mathbf{X}^{(0)}$, $\mathbf{Y}^{(0)}$ in the case of the iterative algorithms) which in turn respond with solution matrices $\mathbf{X}^{sol}$,$\mathbf{Y}^{sol}$. Each column $\mathbf{x}^{sol}_i$, i=1,..,3 of matrix $\mathbf{X}^{sol}$ is the image of the unit vector i under the rotation transform. The quality of the solution can be quantified by looking at the norm of the column difference vectors

$$(17)\qquad d_i = \|\boldsymbol{\delta}_i\| = \left\|\mathbf{x}_i^{sol} - \mathbf{x}_i^{true}\right\|$$
$$\Delta = \max(d_1, d_2, d_3)$$

Ideally, each d_i should be zero, but in the case that the $\mathbf{x}_i$ are arbitrary unit vectors, d_i can have values up to 2. We term a solution a **success**, if $\Delta \le \tau$ where τ is an appropriate threshold in the range [0.0, 0.2].

[5] In the case of the quaternion algorithm the start matrices are converted to quaternions via Eq. (20) and the results, which are quaternions, are converted back to rotation matrices via. Eq. (19).

We show in Figure 4 the results of 200 simulated runs on the three algorithms. Both iterative algorithms have a success rate of 60% or below for all shown thresholds, while only the direct linear algorithm has a 100% success rate. It is clear that all solutions {**X**,**Y**} from the direct linear algorithm must fulfil the orthogonality constraints $\mathbf{X}^T\mathbf{X}=\mathbf{Y}^T\mathbf{Y}=\mathbf{1}$ nearly to perfection, since otherwise they were not all successes even for small thresholds τ.

The success rate for the iterative optimization algorithm based on rotation matrices (red curve in Figure 4) is surprisingly low for small thresholds. A closer inspection of the results revealed the following reason for this: If $\{\mathbf{X}^{\mathbf{true}},\mathbf{Y}^{\mathbf{true}}\}$ is the true solution, the optimization problem as stated in Eq. (13) is invariant against the change

$$\mathbf{X} = k\mathbf{X}^{\text{true}}, \qquad \mathbf{Y} = \mathbf{Y}^{\text{true}} / k$$

for every scalar value $k\neq 0$. This is because the unconstrained optimization problem does not take into account the orthogonality constraints, thus leading to determinants deviating from 1:

$$\det(\mathbf{X}) = k^3, \qquad \det(\mathbf{Y}) = 1/k^3, \qquad \text{although} \qquad \det(\mathbf{XY}) = 1$$

This behaviour was found to be true for many (approx 75%) of the cases in the iterative optimization algorithm. The cure for this is quite simple, namely take the "normalized" solutions

$$(18) \quad \mathbf{X} \leftarrow \text{sgn}(\det(\mathbf{X}))\frac{\mathbf{X}}{\sqrt[3]{|\det(\mathbf{X})|}}, \qquad \mathbf{Y} \leftarrow \text{sgn}(\det(\mathbf{Y}))\frac{\mathbf{Y}}{\sqrt[3]{|\det(\mathbf{Y})|}}$$

which brings the determinant of both **X** and **Y** to 1 (although the matrices become not necessarily orthogonal).

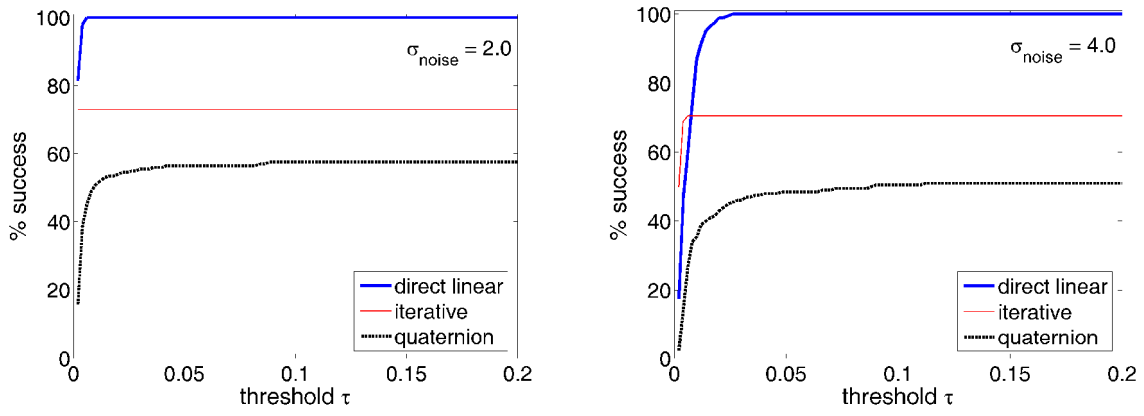

Figure 5: Same as Figure 4, but with an improved iterative algorithm acc. to Eq. (18). Noise levels are $\sigma_{noise}=2^o$ (left) and $\sigma_{noise}=4^o$ (right), the results for $\sigma_{noise}=0^o$ are very similar to the left picture. Now the iterative algorithm based on rotation matrices has a better success rate (70-75%) than the iterative quaternion algorithm (50-60%). All algorithms degrade quite gracefully in the presence of noise.

With this extra step in the first algorithm we get the results shown in Figure 5, much better than before for the red curve. All algorithms reach now very quickly a certain plateau, meaning that the solution is either a complete success or a complete failure. A complete failure is the typical behaviour we expect when an iterative algorithm gets trapped in a wrong local minimum, far away from the true solution. The direct linear algorithm shows a very robust behaviour: For reasonable large thresholds τ we get a 100% success rate, since the problem of local minima does not occur in this case. In other words: not a single complete failure was observed for the direct linear algorithm.

Both iterative algorithms have a failure rate of 25-50%. This can indeed be traced back to the starting point problem: If we had given them a starting point sufficiently close to the true solution, nearly all runs would have been successes. But the starting point is unknown in the real case, and a complete registration failure or "multiple guessing" would pose severe problems for the practical application in the operating theatre.

Note that after the improvement of Eq. (18), the first algorithm outperforms the quaternion algorithm (dashed black curve in Figure 5). We thus are led to the conclusion that the number of variables and the neglect of the orthogonality constraints are not the principal problem for finding a robust convergence behaviour. The quaternion algorithm reduces the dimension in optimization space (from 18 to 8), but the price for this is that the optimization function has a higher degree of nonlinearity which makes obviously the "basins of attraction" smaller. Thus the chance for getting trapped in a local minimum increases.

The direct linear algorithm is also superior when we compare performance (computing time): All three algorithms were implemented in MATLAB and executed on a Pentium® 1500 MHz. As Table 1 shows, the direct linear algorithm is 600-700 times faster than the quaternion algorithm and 60-80 times faster than the iterative rotation matrices algorithm. The reason that the quaternion algorithm is eight times slower than the iterative algorithm lies in the fact that it needs much more iterations in the Levenberg-Marquardt algorithm.

Table 1: Execution times and success rates for the different algorithms when solving the registration problem of Eq. (12) for different n (numbers of endoscopic viewpoints). Results are obtained for τ=0.1 and $\sigma_{noise}=4^o$.

		Algorithm		
	n	iterative (Sec. 3.2)	quaternion (Sec. 3.3)	direct linear (Sec. 3.4)
execution time	4	106 ms	886 ms	1.27 ms
	6	123 ms	940 ms	1.66 ms
	8	119 ms	1216 ms	2.02 ms
success rate	4	71%	51%	100%
	6	82%	47%	100%
	8	83%	41%	100%

Increasing n, the number of measurements (number of endoscopic viewpoints), leads to a roughly linear increase in execution time for the direct linear algorithm (Table 1). For the iterative algorithms there is considerable variation in the execution time for a single run, depending on the number of iterations needed. The success rate does not change very much, we have a saturating increase for the first iterative algorithm, and – surprisingly – a decrease for the quaternion algorithm as n grows. Why would the success rate of the quaternion algorithm decrease with growing n? The quaternion optimization function has a higher degree of nonlinearity of its optimization parameters (quartic terms instead of quadratic terms in Eq. (13), the iterative solution). We suspect that with growing n each additive term i in the χ^2-function contributes many local minima to the optimization surface. This can outperform the effect that the true minimum becomes deeper. The net effect may be that the "basin of attraction" of the true minimum becomes smaller.

Can we tell just from looking at the singular values (SV) of C in the direct linear algorithm whether we have a good solution or not? The situation is depicted in Figure 6 where the results from 200 simulated registrations for n=3 and and 200 for n=4 are shown: With n=4 measurements we have always a good solution with small errors, $\log_{10}(\Delta) \leq -1.8$. For n=3 we have a large fraction (34%) of bad solutions with $\log_{10}(\Delta) > -1.0$.

Table 2: Average of the two smallest singular values (SV) in percent of the largest SV λ_{max} for the direct linear algorithm. Averaging is based on 200 measurements.

	n=3	n=4	n=5
smallest SV	0.7%	0.8%	0.9%
2nd smallest SV λ_2	3.7%	8.7%	15.1%
λ_{max}	2.44	2.82	3.16

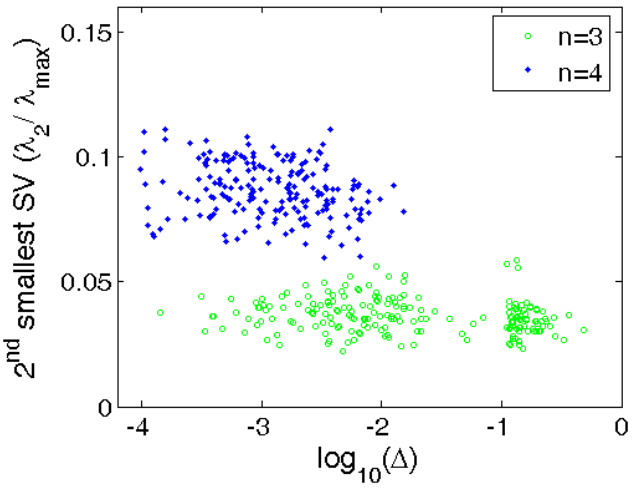

Figure 6: Error Δ vs. 2nd smallest SV λ_2. For λ_2/λ_{max} > 6% we can expect a good solution (Δ<0.01).

The y-axis of Figure 6 shows an important aspect about the spectrum of C. As said in Sec. 3.4 we never get an exact nullspace of C in the presence of noise. We only expect the smallest singular value (SV) λ_1 to be much smaller than all the others, especially smaller than the 2nd smallest SV λ_2. We use the following empirical definition: C has an *approximate "nullspace" of dimension one* if (a) $\lambda_1 \leq$ 2% λ_{max} AND $\lambda_2 \geq$ 6% λ_{max}, where λ_{max} is the largest SV of C. Table 2 shows that $\lambda_1 \leq$ 2% λ_{max} is on average valid for all n. But we see in Figure 6 a clear gap in λ_2: For n=3 we have λ_2<6% λ_{max} and we cannot separate good from bad solutions by just looking at λ_2. But for n=4 the value of λ_2 is considerably larger. If $\lambda_2 \geq$ 6% λ_{max} we can be quite sure to have a good solution.

4.2. Spin-Me-Around Accuracy

We performed several system registrations of the real endoscope (camera calibration already done) based on n=4 measurements. The calibration pattern is a plate with concentric rings which were automatically detected in the endoscopic image. The main result of the system registration is the transformation $\mathbf{D}_C$. With $\mathbf{D}_C$ we can map any point from the OPMS coordinate system into the endoscopic camera view for arbitrary positions of the endoscope, using the time-varying $\mathbf{D}_E$ from the OPMS. We can assess the quality of the system registration by the following spin-me-around test: We manually mark a certain landmark **r** in at least k=2 different camera views. Knowing the k camera positions from the OPMS, we can obtain the 3D-representation $\mathbf{r}_O$ in the OPMS-system using standard triangulation techniques (Longuet-Higgins, 1981). For any subsequent endoscopic view, we map $\mathbf{r}_O$ into the actual camera view and overlay it onto the endoscope image. If camera and system registration are correct, the overlaid mark will be always on top of the landmark **r**, no matter how we "spin around" the endoscopic viewpoint or where the landmark **r** appears in the endoscope image. Table 3 shows that the residual error $\langle\Delta f\rangle \approx 0.7$ mm is in the order of the intrinsic error of the OPMS (> 0.5mm), i.e. the accuracy of the registration is close to its theoretical limit.

Note that it is very important to have an accurate distortion model in order to achieve this accuracy. If we neglect distortion effects (by setting κ_1=0) and if **r** is far away from the piercing point, the error goes up to 3.5 mm instead of 0.7 mm.

Table 3: Average error <Δf> as residual distance between r and the overlay mark in the live endoscope image. Averaging based on 10 images where the landmark r appears at various distances from the piercing point. Δs is the approximate movement of the camera between the k multiple views to determine r_O. Δf_{OPMS} is the observable jitter of the overlay mark when the endoscope is fixed. This jitter stems from noise in repeated OPMS measurements and gives a lower bound on the intrinsic OPMS error. System registration done by *direct linear algorithm*.

k	Δs [mm]	<Δf> [mm]	Δf_{OPMS} [mm]
2	5	0.74	0.5
3	5	0.62	0.5
2	10	0.71	0.5
3	10	0.72	0.5

How often do we need to make a system registration? The transformation D_C remains constant as long as the elements {endoscope, LED-device, camera} stay together. D_C can change, when (a) the LED-device is detached from the endoscope shaft or (b) when the camera is detached from the endoscope shaft. In daily routine, however, it turns out that (a) is nearly never necessary (sterilization can be done with LED-device attached) and (b) that, after detaching and re-attaching the camera, the old D_C is still valid. Thus, we usually perform many surgical interventions with the same system registration.

Further results with the real system concerning tracking accuracy and 3D measurement accuracy will be given in a follow-up paper (Konen et al., 2007).

5. Applications

5.1. Applicability for standard registration

Is the direct linear algorithm also of use for standard registration tasks?

In standard registration procedures (cf. Sec. 2.4) we have two sets of points $\mathbf{x}^{(i)}$ and $\mathbf{y}^{(i)}$, i=1,…,n and seek a transformation $\mathbf{T} \in \mathcal{T}$ which minimizes $\sum_i \left\| \mathbf{T}(\mathbf{x}^{(i)}) - \mathbf{y}^{(i)} \right\|$. If $\mathbf{T} \in \mathcal{T}$ is drawn from the set of affine mappings, we have the case of Procrustes or rigid-body transformation, where the well-known algorithms of Schoenemann (1966) or Farrell (1966), later rediscovered by Arun (1987) and Umeyama (1991), provide a versatile and fast solution for the 2D-2D and 3D-3D-case.

But in many applications the modalities have different dimensions, e.g. 2D-3D, or the transformations must be drawn from a more general set of transformations, e.g. projective or bilinear. In these cases the mappings can be still expressed in matrix form with homogeneous coordinates, but the algorithm of Umeyama (1991) is no longer applicable. The direct linear algorithm will be still applicable to all these cases, and we show this for the example of the 2D-2D projective mapping (the other cases work similarly). Such a projective mapping is easiest to write down in homogeneous coordinates where each 2D-point $(x_1,x_2)^T$ is embedded in 3D by $(hx_1,hx_2,h)^T$ with a new parameter h. This allows a wider class of transforms to be expressed by linear maps. The projective mapping **T** should fulfil

$$\hat{\mathbf{y}}^{(i)} = \begin{pmatrix} hy_1 \\ hy_2 \\ h \end{pmatrix} = \mathbf{T}\mathbf{x}^{(i)} = \begin{pmatrix} t_{11} & t_{12} & t_{13} \\ t_{21} & t_{22} & t_{23} \\ t_{31} & t_{32} & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ 1 \end{pmatrix}$$

which – by eliminating h – can be written in the form of 2 equations

$$(t_{31}x_1 + t_{32}x_2 + 1)y_1 = t_{11}x_1 + t_{12}x_2 + t_{13}$$
$$(t_{31}x_1 + t_{32}x_2 + 1)y_2 = t_{21}x_1 + t_{22}x_2 + t_{23}$$

This is linear in the terms t_{ij} (although it is not linear in x_i and not linear in y_i), so it can be written in a form similar to Eq. (15), namely

$$\mathbf{Cz} = 0 \quad \text{with} \quad (2n \times 8)\ \text{matrix}\ \mathbf{C}\ \text{and} \qquad \mathbf{z}^T = \begin{pmatrix} t_{11} & t_{12} & \ldots & t_{32} \end{pmatrix} \tag{8'}$$

with a certain matrix **C** containing elements from $\mathbf{x}^{(i)}$ and $\mathbf{y}^{(i)}$. Eq. (8') can be solved with the same method as in Sec. 3.4.

Thus we have outlined a general registration method, which may work for a wider class of transformations (affine, projective, bilinear, 2D-2D, 2D-3D, 3D-3D) in the same robust manner. So far we have only applied it to our special 3D-3D-case of Sec. 3 and 4 above. Further research should be done to verify the applicability to the other cases mentioned here. Especially it should be investigated under which conditions the requirement is met that only one singular value (see Sec. 3.4,4.1) is approximately zero.

5.2. Neurosurgical applications

The properly registered endoscope can be used for many purposes: Certain landmarks, either from preoperative data (CT,MRI) – e.g. the location of a tumour – or intraoperatively marked 3D-points can be displayed as an overlay to the live endoscope image (as well as their direction vector when they are outside the field of view). By setting several landmarks it is possible to perform 3D-measurments in the live endoscopic image. Ergonomic aspects of all those functions have been studied in Scholz et al. (2005).

An important application is the coagulation support in the case of red out (Figure 7). We store intraoperatively the endoscopic images together with the 6D-rigid body data of the tracked endoscope. In the case of vision loss due to bleeding we can provide the surgeon with images corresponding to the current position (we display them as grey level images in order to make the distinction to the real camera image clear).

We do not only generate the views, but we also give the surgeon information how deep he has to put in the coagulation fibre in order to be close to the tissue surface. How do we get this? Assume that a point on the tissue surface has been marked at the beginning of surgery with clicks into (at least) two camera views. We do not know the depth yet, we only have lines of sight. But by transforming the lines of sight back into the O-system via $D_E^{-1}(D_C^{-1}(\cdot))$ we can intersect these two (or more) lines and – assuming that they are not all identical – calculate in the O-system the 3D coordinates r_O of the landmark. Once established, we can map this into any subsequent camera view using $r_C=D_C(D_E(r_O))$, see crosshairs in Figure 8, and give the surgeon at the same time hints about how to place the coagulation fibre, e.g. " advance 4 mm from norm position". As is described in more detail in (Scholz et al., 2000, 2005, 2005b), the surgeon can now make a coagulation to stop the bleeding.

Such a system requires accurate 3D physical-to-image-space registration, which is possible in a robust manner with the methods described here. The surgeon watches in general two monitors, one monitor always with the live image and one monitor with live or virtual images and graphic overlays. He can choose active coagulation as an alternative to conventional techniques like rinsing or careful suction. Those conventional techniques can disturb pressure, temperature and ional composition of cerebrospinal fluid and bear therefore the risk of further severe complications.

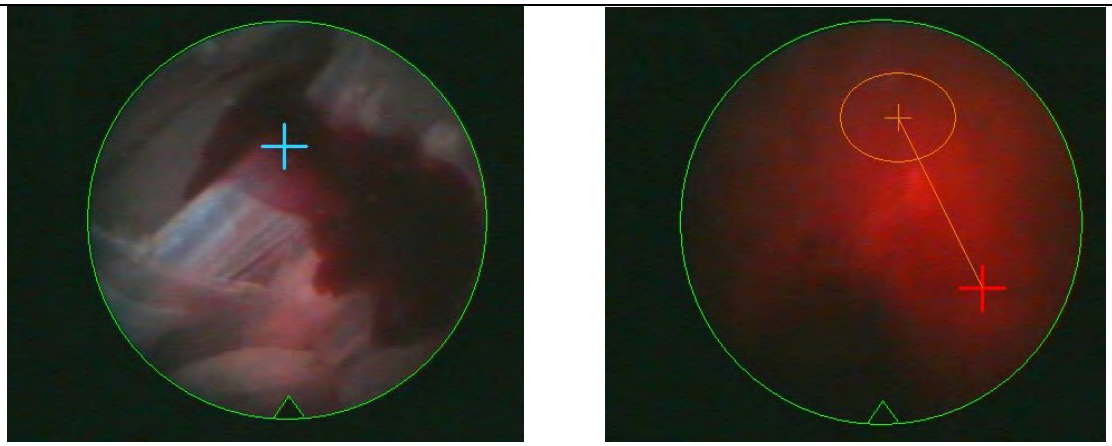

Figure 7: Example of a red out situation in a surgical created cavity in the groin of a rat. Left: Start of a very slow bleeding after damage of the vein. Right: Complete loss of vision due to red out after a few seconds.

Figure 8: Coagulation support for the case of a red out situation. The shown grey level images are previously stored images corresponding to the current position. Left: The orange graphic overlay of the coagulation fibre is far away from the coagulation point (blue mark). The red alarm signal on the side shows that the 3D distance is large. Right: The coagulation fibre has been brought into the correct depth and right x-y-location, the red alarm signal is low. Coagulation was successful and the bleeding stopped.

6. Conclusion

We introduced a new method for the system registration of surgical devices, which provide 2D-projections of the 3D-world seen by the surgeon during interventions, such as endoscopes. The registration method is *markerless*, i.e. it does not need fiducial points (with known extrinsic 3D position in the operating theatre), but only observations of a passive calibration pattern with known (intrinsic) geometry. Shahidi et al. (2002) solve the system registration by placing a second tracker on the calibration pattern. This approach requires, however, that the transformation between this tracker and the calibration grid is known precisely, which amounts to another registration task.

Our method is based on a direct linear algorithm which has been shown to be superior to other iterative, nonlinear approaches based on quaternions or Levenberg-Marquardt optimization. The distinctive advantage of the direct linear algorithm is that it does not need a starting value and cannot get trapped in local minima, thus leading to a robust behaviour in the calibration process (100% success rate instead of 50% or 70%). The direct linear algorithm is

60 times faster than the iterative approach and 600 times faster than the quaternion algorithm.

We have outlined how we expect the direct linear algorithm to work also for registration between modalities with different dimensions (2D-3D, 3D-3D) and for a wider class of transformations (namely projective mappings). It is possible that it is a generalization of the (also linear) registration algorithms by Schoenemann (1966) and Farrell (1966) . Further research should be done to verify this conjecture in real applications of medical image registration.

Appendix A: Quaternions

Quaternions (Shoemake, 1985) are 4D-vectors $q = (q_0, q_1, q_2, q_3) \equiv (q_0, \mathbf{q})$ with norm $\|q\|$ and a conjugated quaternion $\bar{q}$ defined as

$$\|q\|^2 = {q_0}^2 + \mathbf{q}^2$$
$$\bar{q} = (q_0, -\mathbf{q})$$

A vector product between quaternions, resulting in a new quaternion, is defined as:

$$q.p = \left(q_0 p_0 - \mathbf{qp}, q_0\mathbf{p} + p_0\mathbf{q} + \mathbf{q}\times\mathbf{p}\right)$$

A unit quaternion is a quaternion with norm $\|q\|$=1. Each unit quaternion can be written as

$$q = (\cos\frac{\theta}{2}, \sin\frac{\theta}{2}\mathbf{n})$$

with a 3D unit vector **n**, and since each rotation is uniquely specified by a rotation axis **n** and a rotation angle θ, the unit quaternion q is a representative for the rotation R(**n**,θ). The representation is *twofold*: both q and –q are unit quaternions which represent rotation R(**n**,θ).

The formulas relating a quaternion q with the corresponding rotation matrix **R** are:

Transformation from quaternion to rotation matrix:

$$\mathbf{R} = \mathbf{R}(q) = \begin{pmatrix} {q_0}^2 + {q_1}^2 - {q_2}^2 - {q_3}^2 & 2(q_1q_2 - q_0q_3) & 2(q_1q_3 + q_0q_2) \\ 2(q_1q_2 + q_0q_3) & {q_0}^2 - {q_1}^2 + {q_2}^2 - {q_3}^2 & 2(q_2q_3 - q_0q_1) \\ 2(q_1q_3 - q_0q_2) & 2(q_2q_3 + q_0q_1) & {q_0}^2 - {q_1}^2 + {q_2}^2 - {q_3}^2 \end{pmatrix} \quad (19)$$

Transformation from rotation matrix to quaternion: By suitable addition / subtraction of diagonal elements in Eq. (19) we get

$$4{q_0}^2 = 1 + r_{11} + r_{22} + r_{33}$$
$$4{q_1}^2 = 1 + r_{11} - r_{22} - r_{33}$$
$$4{q_2}^2 = 1 - r_{11} + r_{22} - r_{33}$$
$$4{q_3}^2 = 1 - r_{11} - r_{22} + r_{33}$$

One of these terms is the biggest, let us assume without loss of generality that it is $4{q_1}^2$. We get q by using the off-diagonal elements of Eq. (19):

$$q_1 = \frac{1}{2}\sqrt{1 + r_{11} - r_{22} - r_{33}}$$
$$q_2 = \frac{1}{4q_1}(r_{12} + r_{21}), \quad q_3 = \frac{1}{4q_1}(r_{13} + r_{31}), \quad q_0 = \frac{1}{4q_1}(r_{32} - r_{23}) \quad (20)$$

Here we have used the positive square root for q_1. With equal rights we could have taken the negative square root, yielding in turn the equivalent unit quaternion –q.

Appendix B: The structure of matrix C

The (k,l) matrix element for the ith matrix of Eq. (14) can be written as (i=1,...,n)

$$\left[\mathbf{A}^{(i)}\mathbf{Y}^{T}-\mathbf{X}\mathbf{B}^{(i)}\right]_{kl}=\mathbf{a}_k^{(i)}\cdot\mathbf{y}_l-\mathbf{x}_k\cdot\mathbf{b}_l^{(i)} \quad \text{with} \quad \begin{cases}\mathbf{a}_k^{(i)}: & \text{kth } \underline{\text{row}} \text{ of } \mathbf{A}^{(i)}\\ \mathbf{y}_l: & \text{lth } \underline{\text{row}} \text{ of } \mathbf{Y}\\ \mathbf{x}_k: & \text{kth } \underline{\text{row}} \text{ of } \mathbf{X}\\ \mathbf{b}_l^{(i)}: & \text{lth } \underline{\text{column}} \text{ of } \mathbf{B}^{(i)}\end{cases}$$

This is equivalent to

$$\text{(21)}\quad \mathbf{C}^{(i)}\mathbf{z}=\begin{pmatrix}-\mathbf{b}_1^{(i)} & 0 & 0 & \mathbf{a}_1^{(i)} & 0 & 0\\ -\mathbf{b}_2^{(i)} & 0 & 0 & 0 & \mathbf{a}_1^{(i)} & 0\\ -\mathbf{b}_3^{(i)} & 0 & 0 & 0 & 0 & \mathbf{a}_1^{(i)}\\ 0 & -\mathbf{b}_1^{(i)} & 0 & \mathbf{a}_2^{(i)} & 0 & 0\\ 0 & -\mathbf{b}_2^{(i)} & 0 & 0 & \mathbf{a}_2^{(i)} & 0\\ 0 & -\mathbf{b}_3^{(i)} & 0 & 0 & 0 & \mathbf{a}_2^{(i)}\\ 0 & 0 & -\mathbf{b}_1^{(i)} & \mathbf{a}_3^{(i)} & 0 & 0\\ 0 & 0 & -\mathbf{b}_2^{(i)} & 0 & \mathbf{a}_3^{(i)} & 0\\ 0 & 0 & -\mathbf{b}_3^{(i)} & 0 & 0 & \mathbf{a}_3^{(i)}\end{pmatrix}\begin{pmatrix}\mathbf{x}_1\\ \mathbf{x}_2\\ \mathbf{x}_3\\ \mathbf{y}_1\\ \mathbf{y}_2\\ \mathbf{y}_3\end{pmatrix}$$

where each entry in matrix $\mathbf{C}^{(i)}$ is a 1x3 row vector. Thus, $\mathbf{C}^{(i)}$ is a 9x18 matrix corresponding to one specific measurement i. The 9n x 18 matrix **C** is the concatenation of all $\mathbf{C}^{(i)}$, i=1,…,n. Each entry in 18x1 vector **z** corresponds to a 3x1 column vector.

Acknowledgements

We would like to thank the anonymous reviewers as well as G. Plassmann and H. Westenberger for helpful revision comments.

References

[1] Akatsuka Y., Shibasaki T., Saito A., Kosaka A., Matsuzaki H., Asano T., Furuhashi, Y. Navigation System for Neurosurgery with PC platform. Proc. Medicine Meets Virtual Reality 2000, pp 10-16, Newport Beach, CA, 2000.

[2] Arun K. S., Huang T. S., Blostein S. D. *Least-squares fitting of two 3D point sets*, IEEE Trans. on Pattern Analysis and Machine Intelligence, **9**(5), pp. 698-700 1987.

[3] Bartz D. Virtual Endoscopy in Research and Clinical Practice, *Computer Graphics Forum*, 24(1), 2005.

[4] Besl P. J., McKay N. D. *A method for registration of 3-D shapes.* IEEE Trans. Pattern Anal. Mach. Intell. **14**, pp. 239–56, 1992.

[5] Dorward N.L., Alberti O., Zhao J., Dijkstra A., Buurman J., Palmer J.D., Hawkes D., Thomas D.G. *Interactive image-guided neuroendoscopy: Development and early clinical experience*. Minim Invasive Neurosurg **42**[2], 74 – 78, 1999.

[6] Dey D., Gobbi D. G., Slomka P. J., Surry K. J. M., Peters T. M. Automatic Fusion of Freehand Endoscopic Brain Images to Three-Dimensional Surfaces: Creating Stereoscopic Panoramas. IEEE Trans. Med. Imaging 21(1): 23-30 (2002)

[7] Farrell J. L., Stuelpnagel J. C., Wessner R. H., Velman J. R., Brock J. E. *A least squares estimate of satellite attitude, Solution 65-1*, SIAM Rev., **8**, pp. 384-386, 1966.

[8] Green B.F. *The orthogonal approximation of an oblique structure in factor analysis*, Psychometrika, **17**, pp. 429-440, 1952.

[9] Hill D., Batchelor P. G., Holden M., Hawkes D. J. *Medical Image Registration*, Phys. Med. Biol. **46** (2001) R1–R45.

[10] Kawamata T., Iseki H., Shibasaki T., Hori T. *Endoscopic augmented reality navigation for endonasal transsphenoidal surgery to treat pituitary tumors: technical note*. Neurosurgery 50 (6) : 1393 – 1397, 2002.

[11] Konen W., Scholz M., Tombrock S., Tölg S., Brauckmann M., Schwarz A., Adams L. An image-based navigation support system for neuroendoscopic surgery. In : R. Ahlers (ed.) : *5. Symposium Bildverarbeitung*, Technische Akademie Esslingen, 1997.

[12] Konen W., Scholz M., Tombrock S. The VN-project : Endoscopic image processing for neurosurgery. *Computer Aided Surgery* **3**, No. 3, pp. 144-148, 1998.

[13] Konen W., Scholz M., Tombrock S. Verfahren zur endoskopischen Navigation und zur Eichung von Endoskopsystemen. German patent application. Deutsches Patentamt, DE 10 2005 012 295.7, filed March 2005, granted September 2006.

[14] Konen W., Scholz M., Tombrock S. *Robust real-time tracking and 3D-measurement in neuroendoscopic surgery,* in preparation, 2007.

[15] Koppel D., Wang Y.-F., Lee H. Viewing Enhancement in Video-Endoscopy. *Proc. Workshop on Applications Computer Vision*, Orlando, FL, 2002.

[16] Koppel D., Wang Y.-F., Lee H. Image-Based Rendering and Modelling in Video-Endoscopy. *Proc. IEEE International Symposium on Biomedical Imaging,* Arlington, VA, April, 2004.

[17] Kosaka A., Saito A., Furuhashi Y., Shibasaki T. Augmented Reality System for Surgical Navigation Using Robust Target Vision, IEEE Int. Conf. Computer Vision and Pattern Recognition (CVPR'00) 2, no. 2, pp. 187-194, 2000.

[18] Lemke A.-J., Schurig-Urbaniak A.M., Liebig T., Niehues S.M., Haberl H., Leh-mann T.-N., Felix R. Virtual MR endoscopy of the ventricles prior to neurosurgical interventional endoscopy – evaluation of different presentation techniques. *Fortschr Röntgenstr* 176, 1106 – 1113,2004.

[19] Longuet-Higgins, H.C. *A computer algorithm for reconstructing a scene from two projections*, Nature, 293, 133-135, 1981.

[20] Maintz J. B. A., Viergever M. A. *A survey of medical image registration.* Med. Image Anal. **2**, 1–36, 1998.

[21] Press W.H., Teukosky S.A., Vetterling W.T., Flannery B.P. *Numerical Recipes in C: The Art of Scientific Computing*, 2nd Edition, Cambridge Univ. Press, 1992.

[22] Rhode V., Reinges M.H., Krombach G.A., Gilsbach J.M.. *The combined use of image-guided frameless stereotaxy and neuroendoscopy for the surgical management of occlusive hydrocephalus and intracranial cysts.* Br. J. Neurosurg **12**(6) :531 – 538, 1998.

[23] Rhoten P., Luciano M., Barnett G. *Computer-assisted endoscopy for neurosurgical procedures: technical note*. Neurosurgery **40**, No. 3 : 632-638, 1997.

[24] Schoenemann P. H. *A generalized solution of the orthogonal Procrustes problem*, Psychometrika, **31**, pp. 1-10, 1966.

[25] Scholz M., Fricke B., Tombrock S., Hardenack M., Schmieder K., v. Düring M., Konen W., Harders A. Virtual image navigation: a new method to control intraoperative bleeding in neuroendoscopic surgery: Case Reports And Technical Notes. *Journal of Neurosurgery, Vol. **93**, August 2000.*

[26] Scholz M., Dick S., Fricke B., Engelhardt M., Tombrock S., Pechlivanis I., Harders A., Konen W. Consideration of ergonomic aspects in the development of a new endoscopic navigation system. *Br. J. of Neurosurgery, 2005.*

[27] Scholz M., Tombrock S., Konen W., Fricke B., Pechlivanis I., Engelhardt M., Schmieder K., Harders A. *Application of a new developed visual navigation system in humans. First results.* MinInvasive Neurosurg, Vol. 48 (2), 67–72, 2005.

[28] Schwald B., Seibert H. *Registration Tasks for a Hybrid Tracking System for Medical Augmented Reality,* Journal of WSCG (Winter School of Computer Graphics), Vol.12, No.1-3, February 2004.

[29] Shahidi R., Bax M. R., Maurer C. R., Johnson J. A., Wilkinson E. P., West J. B., Citardi M. J., Manwaring K. H., Khadem R. *Implementation, Calibration and Accuracy Testing of an Image-Enhanced Endoscopy System.* IEEE Trans. Med. Imaging 21(11): 1524-1535 (2002).

[30] Shoemake K. *Animating rotation with quaternion curves*, Computer Graphics (Proc. of SIGGRAPH '85) 19, no. 3, pp. 245-254, 1985. See also http://cgafaq.info/wiki/Quaternion for an introduction.

[31] Tsai R.Y. *A versatile Camera Calibration Technique for High Accuracy 3D Machine Vision Metrology Using Off-the-Shelf TV Cameras and Lenses,* IEEE Journal of Robotics and Automation, Vol. RA-3, No.4 , 323-344, 1987.

[32] Umeyama S. *Least-Squares Estimation of Transformation Parameters Between Two Point Patterns*, IEEE Trans. on Pattern Analysis and Machine Intelligence, **13**(4), pp. 376-380, 1991.

[33] Zhang Z. *Iterative point matching for registration of freeform curves and surfaces.* International Journal of Computer Vision, **13**(2):119--152, 1994.