2010 IEEE/ASME International Conference on
Advanced Intelligent Mechatronics
Montréal, Canada, July 6-9, 2010

Evaluating Remote Centers of Motion for Minimally Invasive Surgical Robots by Computer Vision

Jason T. Wilson, Tsu-Chin Tsao, Jean-Pierre Hubschman, M.D., Steven Schwartz, M.D.

Abstract— **This paper addresses the measurement of the location and precision of stationary points in the three dimensional space, commonly termed remote center of motion (RCM), of minimally invasive surgical robotic manipulators. Two-view computer vision is used for its versatility, portability, remote sensing, and cost effectiveness. Geometrical models of cylindrical tools are constructed from camera images. The RCM is computed from the vectors of the cylindrical tool center lines at multiple poses. To verify the approach, a gold standard spherical bearing is first used to measure the RCM location and precision. The same computer vision system and RCM method is then applied to evaluate the RCM of a commercially available laparoscopy surgical robot.**

I. Introduction

The remote center of motion (RCM) is one of the fundamental design requirements for minimally invasive robotic surgical manipulators. During minimally invasive surgery, the surgical instruments are inserted through small incisions or holes punctured through the surface tissue of the patient. Thus the centerline of the surgical instrument must be constrained by the robotic manipulator to pass through a point in space corresponding to the center of the incision. This point is known as the remote center of motion (RCM). There are two ways that robotic manipulators enforce a RCM; the surgical instrument may be constrained by the mechanical kinematics of the manipulator or by limiting allowable trajectories using software. There have been many research efforts focused on developing robotic manipulators capable of enforcing a RCM located on a surgical instrument.

Although not the first, the quintessential example of a mechanically enforced RCM for minimally invasive robotic surgery is the manipulator of the da Vinci Surgical System built by Intuitive Surgical, Inc. The da Vinci uses a double four bar linkage mounted to another rotating joint such that the stationary point of rotation from the linkage intersects with the rotating joint, thus creating the RCM. This mechanism has proven to be successful in a clinical setting to perform urological surgery, orthopaedic surgery, cardiovascular surgery and gynecological surgery to name a few [1], [2], [3], [4].

Research has been conducted to design other mechanically enforced RCM manipulators for minimally invasive surgery. One of the earliest implementations of the four bar linkage is the LARS robot [5]. The four bar linkage has been modified for percutaneous needle insertion [6]. Ocular surgical instruments mounted on semicircular tracks have replaced the double four bar linkage for intravascular ($\leq 70\mu m$) drug delivery, implantation of microdrainage devices and intraretinal manipulation of microelectrodes [7], [8]. Purely spherical manipulators have been designed for vitreoretinal surgery [9] and laparoscopic surgery [10].

Research involving software enforced RCM have led to other manipulator designs. A modified Stewart Platform based micromanipulator was developed as a solution to reduce the bulkiness of the spherical track serial mechanism design [11]. Since the inverse kinematics are straight forward to compute for parallel manipulators, it is simple to compute joint angles for trajectories of the surgical instrument that correspond to RCM enforcement. This design was used to cannulate retinal vessels in cats. The steady hand manipulator has been designed to cooperatively manipulate the surgical instrument with the surgeon [12]. The end effector of the steady hand manipulator has two rotational degrees of freedom which is then mounted to an x, y, z stage. By using software to coordinate the rotational motions with spherical cartesian translations, the RCM can be arbitrarily placed at a location on the surgical instrument. This device was successful in cannulating chorioallantoic membrane blood vessels in 12 day old chicken embryos which where used as retinal vessel phantoms.

Although the examples of RCM enforcing manipulators display micrometer surgical instrument tip precision, none of the studies report the precision and location of the RCM. The ability to measure the RCM precision and location in a quick and cost effective manner would be a valuable tool, especially in the research and development setting. For mechanically enforced RCM, this would allow the researcher to evaluate correctness of prototype manufacture and surgical instrument attachment; for software enforced RCM this would additionally verify correctness of coding/debugging.

Measuring the RCM is a difficult task using conventional measurement devices. It requires measuring the surgical instrument's centerline for many poses in some metric coordinate system. This would involve moving the surgical instrument in the workspace and taking 3-dimensional measurements. One device capable of makeing 3-dimensional measurements is a coordinate measurement machine (CMM). However using a CMM would require touching a surgical instrument, which has known geometry, and using those measurements to compute the centerline. This is not a practical approach due to cost and time consumption. It

Jason Wilson and Tsu-Chin Tsao are affiliated with the department of Mechanical and Aerospace Engineering at UCLA. jdubb@ucla.edu and ttsao@seas.ucla.edu

Jean-Pierre Hubschman and Steven Schwartz are affiliated with the Jules Stein Eye Institute at UCLA. hubschman@jsei.ucla.edu and schwartz@jsei.ucla.edu

might be difficult or even impossible to collocate the da Vinci and a CMM for measurement. For microsurgical manipulators, which typically have surgical instruments with small diameter (< 25 guage), the CMM may deform the shaft introducing measurement errors.

Passive DC type magnetic sensors are commonly used in the clinical setting for surgical guidance, however these sensors typically have millimeter rms resolution. Furthermore the sensors would need to be integrated into the robotic system such that the surgical instrument's centerline is measured accurately.

To overcome these difficulties we describe and implement a method to measure the RCM for a given surgical manipulator, using two cameras by using computer vision. The paper is organized as follows: Section II formalizes the definition of the location and precision of the remote center of motion of a surgical instrument for different poses in the manipulator workspace. Section III describes an algorithm to triangulate the centerline of a surgical instrument given two images taken from pinhole cameras with known relative pose. Section IV briefly describes camera calibration and two-view geometry. Section V describes the implementation on a Gold Standard and the result. Section VI reports the result of the same process applied to the da Vinci Surgical System followed by concluding remarks in Section VII.

II. RCM Evaluation

A. Problem Formulation

The RCM of an ideal surgical manipulator is the point in space, $\mathbf{p}$, that the centerline of the surgical instrument is constrained to pass through regardless of the pose in the workspace. However this condition is never satisfied in practice, thus the definition of the RCM is revised as follows. Given n poses of a surgical instrument, the RCM is defined as the point $\mathbf{p}_{RCM}$ that satisfies,

$$\mathbf{P}_{RCM} \doteq \underset{\mathbf{p}}{\operatorname{argmin}}\, J(\mathbf{d}_1^c, \cdots, \mathbf{d}_n^c) \tag{1}$$

where $\mathbf{d}_i^c$ is the Euclidean distance from $\mathbf{p}$ to the centerline of the surgical instrument at pose i. The cost function J must be chosen for the appropriate physical meaning.

B. Cost Function Choice

Different choices for J lead to different physical meanings of the RCM error. Since the purpose of the RCM is to avoid stress in the tissue at the incision, the most appropriate J is the maximum Euclidean distance from $\mathbf{p}$ to each centerline. This can be written in terms of the ∞-norm as,

$$J = \|(\|\mathbf{d}_1^c\|, \cdots, \|\mathbf{d}_n^c\|)^T\|_\infty = \min \max_i \|\mathbf{d}_i^c\| \tag{2}$$

The $\mathbf{p}$ associated with the solution to (2) minimizes the maximum stress applied to the surface tissue of the patient. The minimization of (2) can be formulated as a Second Order Cone Program as follows,

$$\min \alpha, \quad \text{subject to} \quad \|\mathbf{a}_i^c \tau_i + \mathbf{b}_i^c - \mathbf{p}\| \leq \alpha \tag{3}$$

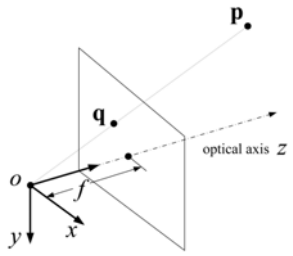

Fig. 1. Image of a Point by a Pinhole Camera

where $i = 1, \cdots, n$

Solving (3) returns the point that is the RCM given n poses of the surgical instrument. The residual is the normal displacements of the centerline from the RCM for the various poses. These displacements directly correspond to stress in the surface tissue.

III. Three Dimensional Centerline Reconstruction From Two Images

Computing the RCM as described in Section II requires measuring the surgical instrument's centerlines for many poses in some metric coordinate system. The method outlined below uses images of the surgical tool centerline, taken from two different cameras, to triangulate the position of the centerlines in $\mathbb{R}^3$

A. Centerline Image Formation

In order to triangulate the centerlines, the cameras must first take an image of the centerline. This is impossible to do directly since the centerline is hidden inside of the surgical instrument. However, the shaft of minimally invasive surgical instruments are typically cylindrical, regardless of the shape of the end effector. By using the line sections corresponding to the edges of the cylinder image, it is possible to compute the image of the centerline.

Image formation is typically modeled using the perspective projection or pinhole model [13]. This model is a mapping $\pi : \mathbb{R}^3 \to \mathbb{R}^2$ which projects 3-D points onto the 2-D image plane. As seen in Fig. 1, the image of a point $\mathbf{p}$ is the intersection of the vector that points at $\mathbf{p}$ and the image plane. For a point $\mathbf{p} = [x, y, z]^T$ represented in the coordinate frame of the camera, its image, $\mathbf{q} = [u, v]$, in the image plane coordinate frame is given by,

$$u = f\frac{x}{z}, \quad v = f\frac{y}{z} \tag{4}$$

To compute the image of the centerline of a cylinder, consider the edges of the image of the cylinder, which appear as line segments on the image plane. In the coordinates of the image plane, the line is represented as,

$$\mathbf{l} = \mathbf{a}\tau + \mathbf{b} \tag{5}$$

The set of points in $\mathbb{R}^3$ that give rise to the line $\mathbf{l}$ in the image plane is also a plane that passes through the origin of the camera, the line $\mathbf{l}$ and is also tangent to the cylinder. A normal vector representing the plane can be computed as

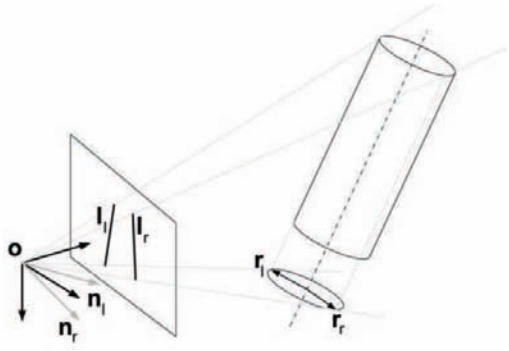

Fig. 2. Image, Preimage, and Coimage of Cylinder Image Edges

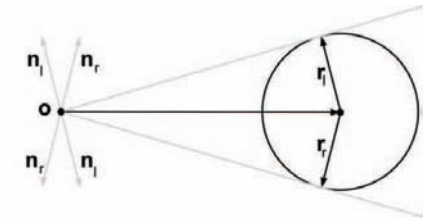

Fig. 3. Cross Section Orthogonal to Centerline

$$\mathbf{n} = \begin{bmatrix} \mathbf{a} \\ 0 \end{bmatrix} \times \begin{bmatrix} \mathbf{b} \\ f \end{bmatrix} \tag{6}$$

The plane defined by $\mathbf{n}$ is the preimage of the line l. The orthogonal compliment of the preimage is the coimage, which contains the normal vector $\mathbf{n}$. The preimage, coimage and thus $\mathbf{n}$ are all equivalent representations of the line $\mathbf{l}$ [14]. There are two lines segments associated with the right and left edges of the cylinder image, $\mathbf{l}_l$ and $\mathbf{l}_r$. Thus an equivalent representation of $\mathbf{l}_l$ and $\mathbf{l}_r$ are normal vectors in the span of the coimage, $\mathbf{n}_l$ and $\mathbf{n}_r$.

In general it is not true that the image of the surgical instrument's centerline, $\mathbf{l}_c$, is the average of $\mathbf{l}_l$ and $\mathbf{l}_r$. However, using the coimage representation, $\mathbf{n}^c$ is the average of $\mathbf{n}_l$ and $\mathbf{n}_r$. This can be seen geometrically by considering the cross section shown in Fig. 2. Fig. 3 shows the cross section such that the line of sight is parallel to the centerline. The cross section is chosen such that it is orthogonal to the center line and passes through the camera origin. Here the vector $\mathbf{b}^c$ is orthogonal to the centerline. From similar triangles, a $\mathbf{n}^c = (\mathbf{n}_l \pm \mathbf{n}_r)$, depending on the directions of $(\mathbf{n}_l, \mathbf{n}_r)$, so long as $\|\mathbf{n}_l\| = \|\mathbf{n}_r\|$.

An equation for $\mathbf{n}^c$ can also be derived algebraically. Note that

$$\mathbf{n}^c = \mathbf{a}^c \times \mathbf{b}^c \tag{7}$$
$$\mathbf{a}^c = \mathbf{n}_l \times \mathbf{n}_r \tag{8}$$

for any $\mathbf{a}^c$ and $\mathbf{b}^c$, which are the same as in (3) and for any $\mathbf{n}_l$ and $\mathbf{n}_r$. An equation for $\mathbf{b}^c$ can be derived as follows. Consider the cylinder $\mathbf{c}$,

$$\mathbf{c} = \mathbf{a}^c\tau + \mathbf{b}^c + \mathbf{r} \tag{9}$$

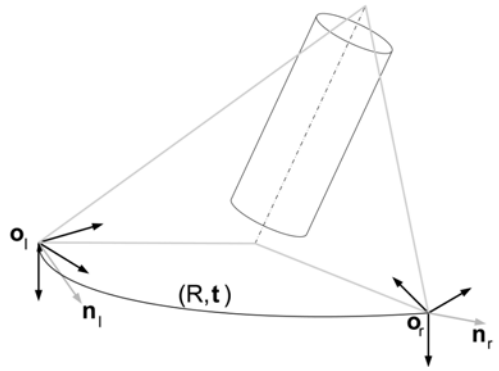

Fig. 4. Centerline Triangulation Using Plane Normal Vector

where $\|\mathbf{r}\| = r$ is the radial vector. Let $\tilde{\mathbf{n}}_l$ and $\tilde{\mathbf{n}}_r$ be unit length. For the plane to be tangent to the cylinder, $\tilde{\mathbf{n}}_l^T\mathbf{r}_l = \tilde{\mathbf{n}}_r^T\mathbf{r}_r = r$, for $(\tilde{\mathbf{n}}_l, \tilde{\mathbf{n}}_r)$ pointing away from the cylinder. This is true since $\mathbf{r}_l$ and $\mathbf{r}_r$ are the radial vectors pointing at the intersection of the planes, $\tilde{\mathbf{n}}_l$ and $\tilde{\mathbf{n}}_r$, with the cylinder, thus they must be parallel. This also means that $\tilde{\mathbf{n}}_l^T\mathbf{a}^c = \tilde{\mathbf{n}}_r^T\mathbf{a}^c = 0$. If $\mathbf{c}$ is a point that is imaged as a left or right edge, $\mathbf{c}_l$ or $\mathbf{c}_r$, then they are contained by the planes $\mathbf{n}_l$ and $\mathbf{n}_r$ respectively and, $\tilde{\mathbf{n}}_l^T\mathbf{c}_l = \tilde{\mathbf{n}}_r^T\mathbf{c}_r = 0$. Therefore multiplying (9) on the left by $\tilde{\mathbf{n}}_l^T$ and $\tilde{\mathbf{n}}_r^T$ results in,

$$\tilde{\mathbf{n}}_l^T\mathbf{b}^c + r = 0 \tag{10}$$
$$\tilde{\mathbf{n}}_r^T\mathbf{b}^c + r = 0 \tag{11}$$

A third constraint can be obtained by choosing the $\mathbf{b}^c$ perpendicular to $\mathbf{a}^c$

$$\mathbf{a}^{cT}\mathbf{b}^c = 0 \tag{12}$$

Solving (10), (11) and (12) for $\mathbf{b}^c$, and computing (7), returns the image of the centerline. Although the centerline isn't directly visible, it is possible to compute its image using only the edges of the cylinder image.

B. Centerline Triangulation

Fig. 4 shows that the centerline of the surgical instrument is the intersection of the preimage of the centerline in the left and right camera. Therefore, centerline triangulation can be achieved by computing $(\mathbf{a}^c, \mathbf{b}^c)$ in (3) using the centerline normal vectors $\mathbf{n}_l^c$ and $\mathbf{n}_r^c$. Assume that the coordinate transformation between cameras $(R, \mathbf{t})$ is known, so that that a point in the right camera $\mathbf{p}_r$ can be represented in the left camera following the rigid body transformation,

$$\mathbf{p}_l = R\,\mathbf{p}_r + \mathbf{t} \tag{13}$$

Since the vector $\mathbf{a}^c$ is in the intersection of both planes it is orthogonal to $\mathbf{n}_l^c$ and $\mathbf{n}_r^c$. Thus an $\mathbf{a}^c$ in the coordinate frame of the left camera is,

$$\mathbf{a}^c = \mathbf{n}_l^c \times R\,\mathbf{n}_r^c \tag{14}$$

To compute $\mathbf{b}^c$, consider the equation of a plane. For a point, $\mathbf{x}$ to be on a plane it must satisfy $\mathbf{n}^T(\mathbf{x} - \mathbf{x}_0) = 0$, where $\mathbf{n}$ is the normal vector and $\mathbf{x}_0$ is some point on the plane.

Since $\mathbf{b}^c$ is any point at the intersection of planes $\mathbf{n}_l^c$ and $\mathbf{n}_r^c$, it must satisfy.

$$\mathbf{n}_l^{cT}(\mathbf{b}^c - 0) = 0 \quad (15)$$

$$\mathbf{n}_r^{cT} R^T(\mathbf{b}^c - \mathbf{t}) = 0 \quad (16)$$

again choosing the left camera coordinate frame. For a third constraint, $\mathbf{b}^c$ can be chosen such that is orthogonal to $\mathbf{a}^c$ as in (12). Solving (15), (16) and (12) for $\mathbf{b}^c$ completes the triangulation of the surgical instrument centerline in the coordinates frame of the left camera. It should be noted that (14) equals zero when $\mathbf{a}^c \times \mathbf{t} = 0$. Physically this happens when the centerline of the surgical instrument is parallel to the line connecting the cameras, and should be avoided.

IV. Calibrated Two View Geometry

In the previous section we present a method to compute the centerline of the surgical instrument in $\mathbb{R}^3$ given its image in two distinct views from *pinhole* cameras with a *known* coordinate frame relationship $(R, \mathbf{t})$. However, in practice images are taken with *digital* cameras that have *unknown* coordinate frame relationships. This section describes which standard computer vision techniques were used to compute the geometric primitives in normalized pinhole cameras and compute the camera coordinate frame relationships. Detailed treatments of the topic are given by [14] and [15].

A. Camera Calibration

The following describes the camera model used to transform physical camera image primitives into normalized pinhole camera image primitives.

1) Calibration Matrix: The image formation modeled by a projection and the camera calibration matrix,

$$z \begin{bmatrix} u' \\ v' \\ 1 \end{bmatrix} = \begin{bmatrix} fs_x & 0 & o_x & 0 \\ 0 & fs_y & o_y & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \quad (17)$$

where f is the focal length, s_x and s_y are scaling factors to account for the rectangular pixels and o_x and o_y shift the coordinates to the upper left of the image. Equation (17) can be partitioned as $\lambda \mathbf{p}' = [K|0]\mathbf{p}$, where K is so called *calibration matrix*.

2) Radial Distortion: Commercially available cameras typically introduce some lens distortion. Here we only consider radial distortion. The model used is the same model used by [16]. For the image $\mathbf{q}_n$ of a point $\mathbf{p}$ taken by a normalized pinhole camera ($f = 1$), the distorted image $\mathbf{q}_d$ is

$$\mathbf{q}_d = (1 + \beta_1 r^2 + \beta_2 r^4)\mathbf{q}_n \quad (18)$$

where $r^2 = u_n^2 + v_n^2$ and β_1 and β_2 are the radial distortion coefficients. The final image $\mathbf{q}$ taken by the camera is $\mathbf{q} = K\mathbf{q}_d$.

B. Two View Geometry

Given more than eight noise free point correspondences, it is possible to compute the camera pose, $(R, \mathbf{t})$ using the epipolar constraint by implementing the eight-point algorithm [17]. However, this method performs poorly in the presence of noise. Instead, the images were considered to be noisy, $\tilde{\mathbf{x}}_1 = \mathbf{x}_1 + \mathbf{w}_1$ and $\tilde{\mathbf{x}}_2 = \mathbf{x}_2 + \mathbf{w}_2$, where $\mathbf{x}_i$ is the noise free image and $\mathbf{w}_i$ are errors in the physical image formation. Then the optimal camera pose was computed by minimizing the following,

$$\min \phi(\mathbf{x}_1, R, \mathbf{t}) = \sum_{j=1}^{n} \|\mathbf{w}_1^j\|^2 + \|\mathbf{w}_2^j\|^2 \quad (19)$$

$$= \sum_{j=1}^{n} \|\tilde{\mathbf{x}}_1^j - \mathbf{x}_1^j\|^2 + \|\tilde{\mathbf{x}}_2^j - \mathbf{x}_2^j\|^2$$

$$\text{subject to} \quad \mathbf{x}_1^T E \mathbf{x}_2 = 0, \quad \mathbf{x}_1^T e_3 = 1, \quad \mathbf{x}_2^T e_3 = 1 \quad (20)$$

where E is the essential matrix, $e_3 = [0, 0, 1]^T$ and $\mathbf{w}_i^j = [w_{i1}^j, w_{i2}^j, 0]^T$. The minimization process was initialized using the output of the eight-point algorithm. This type of minimization is also known as *bundle adjustment*.

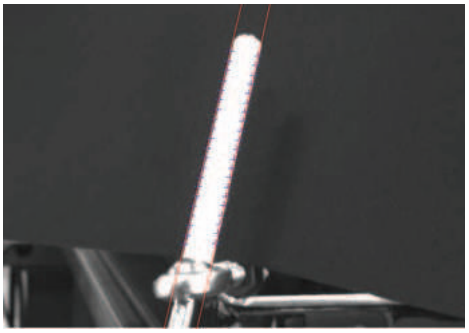

Fig. 5. Left and Right Images of Gold Standard for Pose $i = 1$

V. RCM Measurement; Gold Standard

The algorithm presented to evaluate the RCM described in Sections II, III and IV involve many steps, each with the potential of adding error, thus degrading the evaluation. It is also difficult to characterize the error accumulation because of many potential error sources. Therefore, in order to test this method's fitness for RCM evaluation, the process was first vetted using a *Gold Standard*. By Gold Standard it is meant that the algorithm was applied to a device known to have a very precise RCM. The device chosen was a 10mm precision shaft pressed into an all steel spherical bearing as shown from the right camera in Fig. 5. A RCM will be enforced at the center of the bearing so long as the ball is sufficiently round, and the centerline of the shaft intersects the center of the sphere, which is typically the case. For the components used, it is expected that the Gold Standard RCM would have sub-millimetric RCM error. Thus any error in the RCM evaluation for the Gold Standard should

TABLE I
RCM EVALUATION RESULTS

	Calibration	Gold Standard	da Vinci
error (max)	0.253mm	0.391 mm	7.086 mm
error (rms)	0.130mm	0.145 mm	3.010 mm

be attributed to errors introduced due to image acquisition or signal processing, and *not* mechanical RCM error. The camera configuration and shaft size of the Gold Standard evaluation was substantially identical to the da Vinci assessment. Therefore the error in the Gold Standard evaluation is representative of the errors introduced by the algorithm during the da Vinci assessment.

The Gold Standard was mounted to an optical table and cameras placed at approximately $90\,\mathrm{deg}$. The cameras used were both Pulnix TM-7EX monochrome cameras with image resolution of 640×480 pixels. Image capture was done using a National Instruments PXI-1409 analog image acquisition card.

The cameras were calibrated using the *Camera Calibration Toolbox for Matlab*[18]. The toolbox was configured to use a camera model that was the same as described in Section IV. This toolbox uses the corners of a calibration checkerboard seen from different angles as features for the calibration algorithm. In this case the checkerboard had $15\text{mm} \times 15\text{mm}$ squares and was 7×9 squares in overall dimension. The calibration was done using thirty images.

Twenty images of the same checkerboard were used to compute the relative pose of the cameras as described in Section IV. These images could have also been used to calibrate the cameras, however in order to get a better calibration, extra images were used to get feature points at more locations in the field of view the cameras. The eight-point algorithm and bundle adjustment were done using functions from the *UCLA Vision Lab Geometry Library (VLG)* which is available here [19]. The solution to (19) and (20) is solved up to a scale factor. Typically the solution normalizes $\mathbf{t}$, thus the units of the reconstruction is in camera translation units. In order to upgrade to metric units (mm), a least-squares minimization was performed as follows,

$$\min_{s} \|\mathbf{L}s - \mathbf{1}l\|_2 \tag{21}$$

where $\mathbf{L}$ is the length of the 3-D reconstructed checkerboard edges, l is the known checkerboard edge length ($l = 15\text{mm}$), s is the scaling factor and $\mathbf{1}$ is a vector of ones that is the same dimension as the rows of $\mathbf{L}$. After the metric upgrade the distance between the cameras was found to be $\|\mathbf{t}\| = 525.65\text{mm}$. The maximum and r.m.s error of the reconstruction is presented in Table I.

Once the camera calibration and pose was computed, seventy images of the Gold Standard were taken with different shaft angles. These angles were approximately evenly spaced throughout a $57\,\mathrm{deg}$ cone. Each image was then undistorted using the model from (18). A white stripe was put on the precision shaft and a black background was placed behind it to ensure a high image intensity gradient at the shaft's image edge for accurate edge detection. Points along each edge of the shaft images were extracted using an edge extraction function from [18]. This algorithm is based on a Harris corner detector and boasts sub-pixel resolution of 0.1 pixels. A line fit was then done to obtain $\mathbf{l}'_l$ and $\mathbf{l}'_r$, the prime denoting units of pixel. A representative example is shown on Fig. 5 for the right camera. The lines $\mathbf{l}'_l$ and $\mathbf{l}'_r$ are then computed in the normalized camera model using the camera calibration matrix,

$$\mathbf{l} = K^{-1}\mathbf{l}' \tag{22}$$

Once $\mathbf{l}_l$ and $\mathbf{l}_r$ are computed, the centerlines were triangulated as described in Section III. Finally (3) was solved for $\mathbf{p}_{RCM,\infty}$ using `CVX`, a package for specifying and solving convex programs [20]. The location of the RCM was computed to be $\mathbf{P}_{RCM} = (6.436\ 65.741\ 478.266)^T$ millimeters. The ∞-norm was chosen originally for its direct physical meaning for the RCM error. However, the 2-norm solution associated with the cost function $J = \|\mathbf{d}_1^c\|^2 + \cdots + \|\mathbf{d}_n^c\|^2$ was also computed for comparison. The solutions for the 2-norm and ∞-norm differ on the order of 10^{-8}. Therefore, for all intents and purposes the solutions are the same, which means that although the physical meaning is not the appropriate one, the 2-norm solution is a very good estimate of the RCM location. A single set of values is presented in Table I representing both the 2-norm and ∞-norm solution's RCM error.

The errors shown are representative errors for the RCM evaluation algorithm for the scenario described. These errors are similar to the errors from the calibration procedure. The errors from the calibration procedure should give a measure of how well a scene can be reconstructed assuming zero image processing and triangulation errors. Although not directly comparable, since that the Gold Standard errors (0.391 mm max and 0.145 mm rms) are close to the calibration errors (0.253mm max and 0.130mm rms), the RCM evaluation is as accurate as can be expected for the Gold Standard. It is possible to reduce the reconstruction error by reducing the scale of the imaging set up. This is due to the scale factor s that was computed in (21).

VI. RCM MEASUREMENT; DA VINCI SURGICAL SYSTEM

The da Vinci is a teleoperated robot, meaning the surgeon sits a master console and gives commands via joysticks which then map to motions of the robotic manipulator. The surgeon visualizes the surgical field via a stereoscopic endoscope, which provides the surgeon a 3-D view.

The same process described in Section V was repeated for the da Vinci Surgical System. Fig. 6 shows the camera configuration, and the trocar attached to the double four bar linkage. After the metric upgrade the distance between the cameras was found to be $\|\mathbf{t}\| = 551.94\text{mm}$. The maximum error of the reconstruction of the edges was 0.441mm and the r.m.s value was 0.166mm. Thus the configuration of the cameras and the calibration results are comparable

Fig. 6. Camera Configuration for the RCM Evaluation of the da Vinci Surgical System

between the Gold Standard evaluation and the da Vinci evaluation. The RCM location was computed to be $\mathbf{P}_{RCM} = (-2.802\ 37.876\ 420.767)^T$ millimeters. Again, this is similar to the Gold Standard location with respect to the cameras. The errors from the da Vinci RCM evaluation are shown in Table I.

The da Vinci surgical instrument manipulator tested showed sub-centimeter maximum RCM error. From the Gold Standard evaluation, the values for the da Vinci RCM error are likely to be correct within 0.441 mm. This is a reasonable tolerance for a manipulator designed to perform laparoscopic surgery, since the skin at the abdomen has some compliance.

VII. Conclusion

A method to measure the RCM for minimally invasive robotic surgical manipulators using two cameras was developed. This computer vision method enjoys the benefits of the definition of the RCM location, which was formalized using the 2-norm and ∞-norm. Two sets of experimental data suggest, that for the resolution of the RCM evaluation, the two solutions are equivalent. The 0.145 mm rms error for the spherical bearing Gold Standard RMS precision compares closely with the 0.130 mm camera calibration reconstruction error and it has confirmed the validity of the approach. The sub-centimeter level RCM error measured using the same method on the da Vinci manipulator provides a definite measure, as far as RCM is concerned, for determining the suitability of a particular minimal invasive procedure performed by that particular manipulator. The proposed method can also be useful for calibrating the RCM with respect to the surgical tools, instead of the tool carriages, if tool misalignment is a concern. For delicate surgery requiring higher RCM precision, the computer vision system's resolution can be improved over the sub-millimeter level presented in this paper by appropriately scaling the camera field of view arrangement and enhancing camera resolution.

VIII. Acknowledgments

The authors gratefully acknowledge the UCLA Center for Advanced Surgical and Interventional Technology (CASIT) for providing access to the Intuitive Surgical's da Vinci Surgical System used in the work presented in this paper.

References

[1] K. Cleary and C. Nguyen, "State of the art in surgical robotics: clinical applications and technology challenges," *Computer Aided Surgery*, vol. 6, no. 6, pp. 312–328, 2001.

[2] W. Chitwood, A. Jr., Kypson, and L. Nifong, "Robotic mitral valve surgery: A technologic and economic revolution for heart centers," *Am Heart Hosp J*, no. 1, pp. 30–39, 2003.

[3] B. WL, "Robots in orthopaedic surgery: Past, present, and future," *Clin Orthop Relat Res*, no. 463, pp. 31–36, 2007.

[4] D. Thiel and H. Winfield, "Robotics in urology: Past, present, and future," *J Endourol*, no. 22, pp. 825–830, 2008.

[5] B. Eldridge, K. Gruben, D. LaRose, J. Funda, S. Gomory, J. Karidis, G. McVicker, R. Taylor, and J. Anderson, "A remote center of motion robotic arm for computer assisted surgery," *Robotica*, vol. 14-1, pp. 103–109, Jan-Feb 1996.

[6] H. Bassan, R. Patel, and M. Moallem, "A novel manipulator for percutaneous needle insertion: Design and experimentation," *Mechatronics, IEEE/ASME Transactions on*, vol. 14, no. 6, pp. 746–761, Dec. 2009.

[7] A. Guerrouad and P. Vidal, "Smos: stereotaxical microtelemanipulator for ocular surgery," in *Engineering in Medicine and Biology Society, 1989. Images of the Twenty-First Century., Proceedings of the Annual International Conference of the IEEE Engineering in*, Nov 1989, pp. 879–880 vol.3.

[8] D. Yu, S. Cringle, and I. Constable, "Robotic ocular ultramicrosurgery," *Aust N Z J Ophthalmol.*, no. 26, May 1998.

[9] T. Ueta, Y. Yamaguchi, Y. Shirakawa, T. Nakano, R. Ideta, Y. Noda, A. Morita, R. Mochizuki, N. Sugita, M. Mitsuishi, and Y. Tamaki, "Robot-assisted vitreoretinal surgery: Development of a prototype and feasibility studies in an animal model," *Ophthalmology*, vol. 116, no. 8, pp. 1538 – 1543.e2, 2009.

[10] M. J. H. Lum, J. Rosen, M. N. Sinanan, and B. Hannaford, "Kinematic optimization of a spherical mechanism for a minimally invasive surgical robot," in *in Proc. 2004 IEEE Conf. Robotics and Automation*, 2004, pp. 829–834.

[11] K. Grace, J. Colgate, M. Glucksberg, and J. Chun, "A six degree of freedom micromanipulator for ophthalmic surgery," in *Proceedings., 1993 IEEE International Conference on Robotics and Automation*, 1993, pp. 630–635.

[12] B. Mitchell, J. K. M.D., I. Iordachita, P. Kazanzides, A. Kapoor, M. James Handa, G. Hager, and R. Taylor, "Development and application of a new steady-hand manipulator for retinal surgery," in *2007 IEEE International Conference on Robotics and Automation Roma, Italy*, April 2007, pp. 10–14.

[13] B. Horn, *Robot Vision*. MIT Press, 1986.

[14] Y. Ma, S. Soatto, J. Kosecka, and S. S. Sastry, *An Invitation to 3-D Vision: From Images to Geometric Models*. SpringerVerlag, 2003.

[15] R. I. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed. Cambridge University Press, ISBN: 0521540518, 2004.

[16] Z. Zhang, "Flexible camera calibration by viewing a plane from unknown orientations," *Computer Vision, IEEE International Conference on*, vol. 1, pp. 666–673, 1999.

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

[18] J.-Y. Bouget, "Camera calibration toolbox for matlab." [Online]. Available: http://www.vision.caltech.edu/bouguetj

[19] T. Lee, "Vision lab geometry library (vlg)." [Online]. Available: http://vision.ucla.edu/vlg/

[20] M. Grant and S. Boyd, "Cvx, a modeling system for disciplined convex programming," 2008. [Online]. Available: http://mloss.org/software/view/157/