RCM-SLAM: Visual localisation and mapping under remote centre of motion constraints

Francisco Vasconcelos, Evangelos Mazomenos, John Kelly, and Danail Stoyanov

***Abstract*— In robotic surgery the motion of instruments and the laparoscopic camera is constrained by their insertion ports, i. e. a remote centre of motion (RCM). We propose a Simultaneous Localisation and Mapping (SLAM) approach that estimates laparoscopic camera motion under RCM constraints. To achieve this we derive a minimal solver for the absolute camera pose given two 2D-3D point correspondences (RCM-PnP) and also a bundle adjustment optimiser that refines camera poses within an RCM-constrained parameterisation. These two methods are used together with previous work on relative pose estimation under RCM [1] to assemble a SLAM pipeline suitable for robotic surgery. Our simulations show that RCM-PnP outperforms conventional PnP for a wide noise range in the RCM position. Results with video footage from a robotic prostatectomy show that RCM constraints significantly improve camera pose estimation.**

I. INTRODUCTION

Some surgical procedures, specially in the treatment of prostate, kidney, and bladder cancer, are increasingly being performed with the assistance of a robot [2], [3], [4]. During a robotic intervention, the surgeon sits at a console with a stereoscopic display and operates on the anatomy through tele-manipulation of articulated tools and a laparoscopic camera. Although surgical robots have the potential of performing automated tasks, this capability has not yet translated to clinical practice. One of the main challenges is to accurately represent the vision system, surgical tools, and target anatomy in the same coordinate frame. Therefore, estimating the motion of the laparoscopic camera during surgery is a fundamental step towards developing robotic task automation and providing assisted navigation to the surgeon. Additionally, camera motion estimation is also a prerequisite for the development of augmented reality assisted intervention techniques that overlay pre-operative (MRI, CT) or intra-operative (Ultrasound) data on laparoscopic images [5]. Although camera motion can be estimated through forward kinematics of the robot joints, this has proven to be insufficient for surgical applications. The main reason is that the non-rigid anatomic site cannot be accurately represented in a camera with a single rigid transformation. Furthermore the large distance between the end-effector and the tip of the laparoscopic camera significantly amplifies hand-eye translation errors when mapping robot coordinates to the relatively narrow workspace of the camera [6]. Alternatively, camera motion can be estimated from visual input, a widely studied topic within the domain of Simultaneous Localisation and Mapping (SLAM) [7] and Structure-from-Motion (SfM) [8]. However, most of the developed SLAM and SfM methods are aimed at human made environments (streets, indoors, etc) and work unreliably when applied to laparoscopic surgical video. The presence of a dynamic scene that includes non-rigid tissues, scene occlusions by surgical tools and blood, and fast camera motions in a close range environment all contribute to make SLAM in surgery a very challenging task [9].

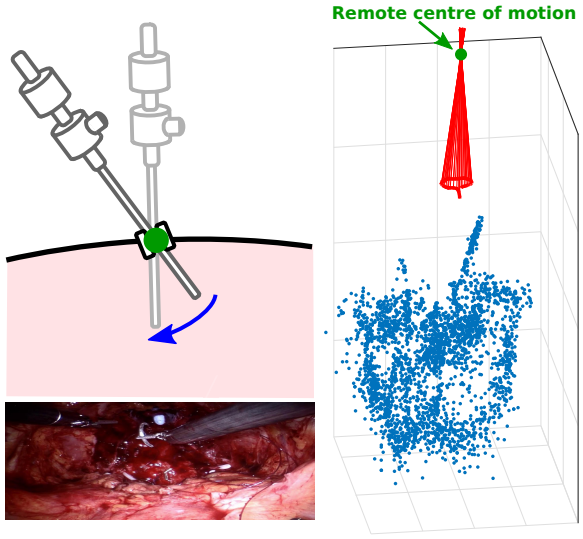


Fig. 1: In robotic surgery, the camera motion is constrained by a remote centre of motion (RCM). This reduces the range of feasible motions to 4 degrees of freedom.

In this paper we exploit remote centre of motion (RCM) constraints [10] for improving camera motion estimation in laparoscopy. In minimaly invasive surgery, both the instruments and the laparoscope are inserted inside the patient through small incisions that put strong boundaries on their motion range. RCM contraints have been used in the context of surgical robotics for localisation [11], control [12], and segmentation tasks [13]. Formulating a RCM constrained motion space simplifies the geometric relations between camera views and reduces the number of motion parameters to estimate. This approach has proven successful while estimating the relative pose between two camera viewpoints

Francisco Vasconcelos, Evangelos Mazomenos, John Kelly, and Danail Stoyanov are with Wellcome/EPSRC Centre for Interventional and Surgical Sciences (WEISS), University College London, United Kingdom f.vasconcelos@ucl.ac.uk

This work was supported by the Wellcome/EPSRC Centre for Interventional and Surgical Sciences (WEISS) (203145Z/16/Z), an Innovative Engineering for Health award by the Wellcome Trust WT101957, and Engineering and Physical Sciences Research Council (EPSRC) grants NS/A000027/1, EP/N027078/1, EP/P012841/1, EP/P027938/1, EP/R004080/1.

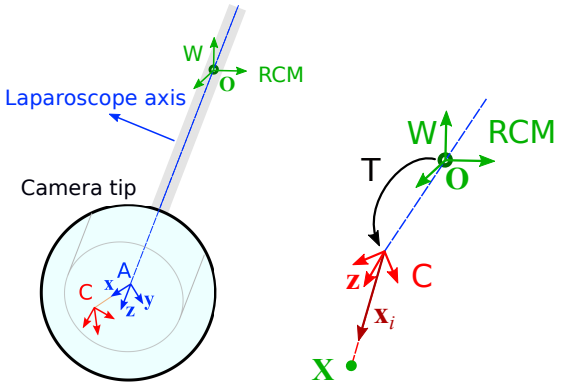


Fig. 2: (a) General RCM model; (b) Aligned axis assumption.

using 2D point correspondences [1]. However, the two-view constraints degenerate when there is no translation and therefore are not suitable for estimating incremental motions between consecutive frames of a video, where the translation is expected to be very small. In this scenario, SLAM methods typically establish correspondences between 2D image points and 3D reconstructed points in the scene (i. e. points that have been detected in at least three camera views). To achieve continuous motion estimation under RCM constraints, we go through the steps of a visual SLAM pipeline based on bundle adjustment [14] and adapt each of them to our proposed RCM formulation. Our contributions can be summarised as follows:

- A minimal solver for absolute camera pose using 2D-3D correspondences with RCM constraints (RCM-PnP). While with unconstrained motion this requires 3 correspondences (P3P [15]), in the RCM case only 2 correspondences are required. This not only eliminates a PnP degenerate case (3 points lying on a line), but also reduces combinatorics when eliminating outlier correspondences with RANSAC [16].
- A bundle adjustment optimisation framework that refines camera poses and 3D reconstructed points on the RCM constrained space (RCM-BA).
- A RCM-SLAM pipeline that uses RCM relative pose [1] for batch initialisation, RCM-PnP for incrementally adding new views and RCM-BA for motion refinement. This is tested on video footage from a prostatectomy performed with a Da Vinci® Si surgical robot, showing that RCM constraints significantly boost SLAM performance for monocular trajectory estimation.

II. RCM Formulation

Consider a camera C constrained by a RCM **O** (Fig. 2a). Without loss of generality we consider that **O** is located at origin of the world reference frame W. In the most general case the camera coordinate frame C can be located at any arbitrary position relative to the laparoscope axis A. This is specially the case with a stereo laparoscope (translation offset between camera and axis) and with angled tip laparoscopes (rotation offset between camera and axis). Assuming that A moves rigidly with C and that the z-axis of A is aligned with the RCM at the origin, the transformation between C and A has only 4 degrees of freedom. This is the case because A has two parameters that can be arbitrarily fixed: the distance to the RCM, and any rotation along the $z-axis$.

In [1] it was argued that if we consider C and A to be coincident (*aligned axis assumption*, Fig 2b), the approximation is still sufficiently accurate for relative motion estimation while using the stereo laparoscope of the Da Vinci® Si surgical robot. The main reason is that the distance between C and A is relatively small (around 2.5 mm) when compared to the distance between A and W (70-100 mm). Simulation results also showed resilience of the model to rotation misalignments, however, no test was performed with real data from an angled tip laparoscope.

We will show that the *aligned axis assumption* is also a good approximation for estimating absolute pose using correspondences between 2D image points and 3D reconstructed points (Perspective-n-Point).

III. Perspective-n-Pose under RCM (RCM-PnP)

The Perspective-n-Pose (PnP) problem consists in determining a camera pose T with rotation R and translation $\mathbf{t}$ using three or more correspondences between 2D image points $\mathbf{x}_i$ and 3D points $\mathbf{X}_i$ (Fig. 2b). For formulation convenience, we consider T to be the transformation mapping coordinates from W to C. Each 2D-3D correspondence establishes the following constraint

$$\mathrm{R}\mathbf{X}_i + \mathbf{t} = \lambda_i \mathbf{x}_i \tag{1}$$

where λ_i is the depth of $\mathbf{X}_i$ to the camera. Given that the $z-$axis of C is aligned with the RCM at the origin **O**, the translation $\mathbf{t}$ has the form $\begin{pmatrix}0 & 0 & -z\end{pmatrix}^\mathsf{T}$. Note that in the surgical context z has to be a positive value for the RCM to be behind the camera.

From equation 1 we can obtain

$$[\mathbf{x}_i]_\times \mathrm{R}\mathbf{X}_i + [\mathbf{x}_i]_\times \begin{pmatrix}0 & 0 & -z\end{pmatrix}^\mathsf{T} = 0 \tag{2}$$

where $[\mathbf{x}_i]_\times$ is a 3×3 skew symmetric matrix such that $[\mathbf{x}_i]_\times \mathbf{v} = \mathbf{x}_i \times \mathbf{v}, \forall \mathbf{v} \in \mathbb{R}^3$. This can be written as

$$\left(\mathbf{X}_i^\mathsf{T} \otimes [\mathbf{x}_i]_\times \quad [\mathbf{x}_i]_\times \begin{pmatrix} x_{2,i} \\ -x_{1,i} \\ 0 \end{pmatrix}\right) \begin{pmatrix} vec(\mathrm{R}) \\ z \end{pmatrix} = 0 \tag{3}$$

where $\otimes$ denotes the Kronecker product. From these three linear equations, only up to two are linearly independent by construction. Therefore, each correspondence between a 2D point $\mathbf{x}_i$ and a 3D point $\mathbf{X}_i$ puts two constraints on a linear system with 10 unknowns. With two correspondences (4 constraints) we can determine by SVD decomposition a 6D linear solution subspace with the form

$$\mathrm{R} = a_1\mathrm{R}_1 + a_2\mathrm{R}_2 + a_3\mathrm{R}_3 + a_4\mathrm{R}_4 + a_5\mathrm{R}_5 + a_6\mathrm{R}_6 \tag{4}$$

$$z = a_1 z_1 + a_2 z_2 + a_3 z_3 + a_4 z_3 + a_5 z_5 + a_6 z_6 \tag{5}$$

where a_j $(j = 1,...,6)$ are the new problem unknowns. Assuming R_6, z_6 correspond to the highest singular value of

TABLE I: Conventional and RCM constrained algorithms that can be used in a SLAM pipeline (Algorithm 1).

	Conventional	RCM-constrained
RELATIVEPOSE()	5-point [18]	4-point [1]
ABSOLUTEPOSE()	PnP [15], [19]	RCM-PnP (this paper)
REFINE()	Bundle adjustment [14]	RCM-BA (this paper)

the SVD decomposition and given that a zero matrix is not a valid rotation, we can safely assume that $a_6 \neq 0$. We divide equations 4, 5 by s_6 to express the scaled rotation $a_6^{-1}\mathsf{R}$ and the scalar $a_6^{-1}z$ in terms of 5 unknowns. Determining a scaled rotation from 5 unknown linear parameters is a quadratic system of equations with 8 discrete solutions, and a closed form solver for this problem has already been proposed [17]. After determining $a_6^{-1}\mathsf{R}$, a_6 is made such that $\mathsf{R}^\mathsf{T}\mathsf{R} = \mathtt{I}$ and z is computed from equation 5. Solutions with complex values, $\det(\mathsf{R}) = -1$ or $z < 0$ can be discarded.

Note that this method requires knowing the coordinates of 3D points $\mathbf{X}_i$ in the reference frame of the RCM. In the context of a SLAM pipeline this can be achieved once two or more views have been estimated, by intersecting their optical axes.

IV. RCM-SLAM PIPELINE

In this section we describe a basic visual SLAM pipeline based on bundle adjustment and highlight the steps that can be replaced by equivalent RCM constrained methods.

As we have seen in the previous section, absolute pose can only be used once 3D points in the scene are reconstructed. While with a stereo camera this information is available right from the first frame, the monocular case requires an initialisation process across the first few frames to create a starting 3D point map. This can be done via relative pose estimation between two views with sufficient displacement [18]. After the initialisation is done, each new frame can be added to the trajectory using correspondences with the 3D points already in the map. The basic structure of a visual SLAM pipeline is described in Algorithm 1. There are three functions in this process that can be replaced by RCM-constrained equivalents. RELATIVEPOSE$(\mathsf{M}_1, \mathsf{M}_2)$ with RCM-constraints was proposed in [1], ABSOLUTEPOSE(M, X) was addressed in the previous section, and REFINE$(\mathsf{T}_1, ..., \mathsf{T}_i, \mathsf{M}_1, ..., \mathsf{M}_i, \mathsf{X})$ will be addressed in the next section. The equivalence between conventional and RCM constrained methods is summarised in Table I.

Algorithm 1 SLAM pipeline

procedure MAIN$(\mathsf{I}_1, ..., \mathsf{I}_N, S)$
  $\mathsf{T}_1 = \mathtt{I}$
  $\{\mathsf{T}_2, ..., \mathsf{T}_S, \mathsf{X}\} =$ INITIALISE$(\mathsf{I}_1, ..., \mathsf{I}_S)$
  for $i = (S+1), N$ **do**
    $\mathsf{M}_i =$ MATCH2D-3D$(\mathsf{I}_i, \mathsf{X})$
    $\mathsf{T}_i =$ ABSOLUTEPOSE$(\mathsf{M}_i, \mathsf{X})$
    $\mathsf{X} =$ TRIANGULATE$(\mathsf{T}_1, ..., \mathsf{T}_i, \mathsf{M}_1, ..., \mathsf{M}_i)$
    $\{\mathsf{T}_1, ..., \mathsf{T}_i, \mathsf{X}\} =$ REFINE$(\mathsf{T}_1, ..., \mathsf{T}_i, \mathsf{M}_1, ..., \mathsf{M}_i, \mathsf{X})$
  end for
end procedure
function INITIALISE$(\mathsf{I}_1, ..., \mathsf{I}_S)$
  $\mathsf{M}_S =$ MATCH2D-2D$(\mathsf{I}_1, \mathsf{I}_S)$
  $\mathsf{T}_S =$ RELATIVEPOSE(M_S)
  $\mathsf{X} =$ TRIANGULATE$(\mathsf{T}_S, \mathsf{M}_S)$
  for $i = 2, (N-1)$ **do**
    $\mathsf{M}_i =$ MATCH2D-3D$(\mathsf{I}_1, \mathsf{I}_i)$
    $\mathsf{T}_i =$ ABSOLUTEPOSE$(\mathsf{M}_i, \mathsf{X})$
  end for
end function

V. BUNDLE ADJUSTMENT UNDER RCM (RCM-BA)

Whenever a new camera pose and new 3D points are added to a SLAM map, these parameters can be refined by minimising camera re-projection errors of the form

$$e_{i,j} = ||(\mathsf{K}_j \frac{1}{\lambda_i}(\mathsf{R}_j\mathbf{X}_i + \mathbf{t}_j) - \mathbf{h}_{i,j}||_2 \tag{6}$$

where K_j is the intrinsic matrix of camera j, and $\mathbf{h}_{i,j} = \mathsf{K}_j\mathbf{x}_{i,j}$ is the ith 2D image point in pixel coordinates detected by camera j. In the context of this paper we do not consider non-linear lens distortion, however, this can be easily added to the re-projection error model. In its most basic implementation, the bundle adjustment optimisation problem can be formulated as

$$\min_{\mathbf{t}_i, \mathbf{q}_i, \mathbf{X}_j (i \neq 1)} \sum_{i=1}^{N} \sum_{j=1}^{M} \sigma_{i,j} e_{i,j} \tag{7}$$

where $\mathbf{q}_i$ is a quaternion representation of the rotation R_i and $\sigma_{i,j}$ is a boolean function that takes 1 when $\mathbf{X}_j$ is visible in camera i and 0 otherwise. In order to fix the world reference frame, the first view is not refined (there are other approaches to deal arbitrary scale gauge that are not considered here [14]).

We now assume that our SLAM map is represented in the reference frame W, with the RCM position at the origin and also that the first camera rotation is aligned with W. If we do not refine the translation of the first camera $\mathbf{t}_1 = \begin{pmatrix} 0 & 0 & z_1 \end{pmatrix}^\mathsf{T}$ then the distance between the RCM and the first camera will be constant, which is to say, the RCM position is fixed relative to the camera trajectory. On the other hand, if we do refine z_1, this is equivalent to refining the RCM position relative to the camera trajectory. This implicit representation of the RCM avoids gauge ambiguities that would be caused by using an additional translation vector to refine the RCM location. Our RCM-BA optimisation formulation is therefore the following:

$$\min_{z_1, z_i, \mathbf{q}_i, \mathbf{X}_j (i \neq 1)} \sum_{i=1}^{N} \sum_{j=1}^{M} \sigma_{i,j} e_{i,j} \tag{8}$$

When compared to conventional bundle adjustment, our formulation refines over $4(N-1)+1+3M$ degrees of freedom instead of $6(N-1)+3M$.

VI. EXPERIMENTS

We tested our RCM-PnP minimal solver in simulation to investigate how much noise in the RCM position it is able

to endure, as well as the complete SLAM pipeline in video footage of a robotic prostatectomy performed with the Da Vinci® Si system. The simulation conditions are made as close as possible to the camera parameters of the real scope.

A. RCM-PnP with Simulated Data

We developed a simulation environment with a 1024×768 resolution camera and intrinsics

$$\mathsf{K} = \begin{pmatrix} 900 & 0.01 & 500 \\ 0 & 890 & 360 \\ 0 & 0 & 1 \end{pmatrix} \tag{9}$$

With modern chip on tip cameras the intrinsic parameters remain fixed, and thus we work under this assumption in simulation. The camera detects 3D points using a pinhole model. We do not consider radial lens distortion in simulation, since in the experiment with real data we undistort the images after camera calibration. The camera distance to the RCM location is randomly generated between 40 and 80 mm. As the distance to the RCM decreases the *aligned axis assumption* becomes less valid. Since with real data this distance is at least 70 mm, our simulations are on average biased towards unfavourable conditions. The camera rotation is randomly generated within a 22.5° cone facing towards the 3D points in the scene, which are randomly generated within a $30 \times 30 \times 30$ mm cube centred 200 mm away from the RCM. Both image point detections and the RCM position are injected with Gaussian noise.

In a first experiment we compare RCM-PnP against P3P [15] using 3 points (the minimum for PnP), for different levels of image and RCM noise (Fig. 3a, 3c, 3b,3d). With high image noise RCM-PnP is better than PnP, since it uses the additional RCM information that does not depend on image points. On the other hand, when the RCM noise is too high, PnP outperforms our approach. The break even point is around 2.5 mm RCM noise for 1 pixel image noise, and 6.5 mm for 2.5 pixel image noise.

In a second experiment we generate 100 points, inject 60% of outlier correspondences and compare the same algorithms inside a RANSAC [16] robust estimator (Fig. 3e, 3f). RANSAC generates random solutions by sampling the smallest possible amount of points. While PnP requires sampling at least 3 points, RCM-PnP only needs 2 points. This makes our approach to generaly find a suitable solution in fewer RANSAC iterations. In this case, which is closer to real input data, our estimated rotation outperforms PnP for all tested RCM noise levels (up to 8 mm), while for translation it compensates to use our method up to 4 mm RCM noise.

With real data we use a stereo camera with 5 mm baseline and we expect the laparoscope axis to be roughly in between the two cameras. Therefore, we are expecting the *aligned axis assumption* to add an offset error to the RCM of around 2.5 mm. Even if we account for further RCM estimation errors, this is likely within the noise range where RCM-PnP outperforms PnP.

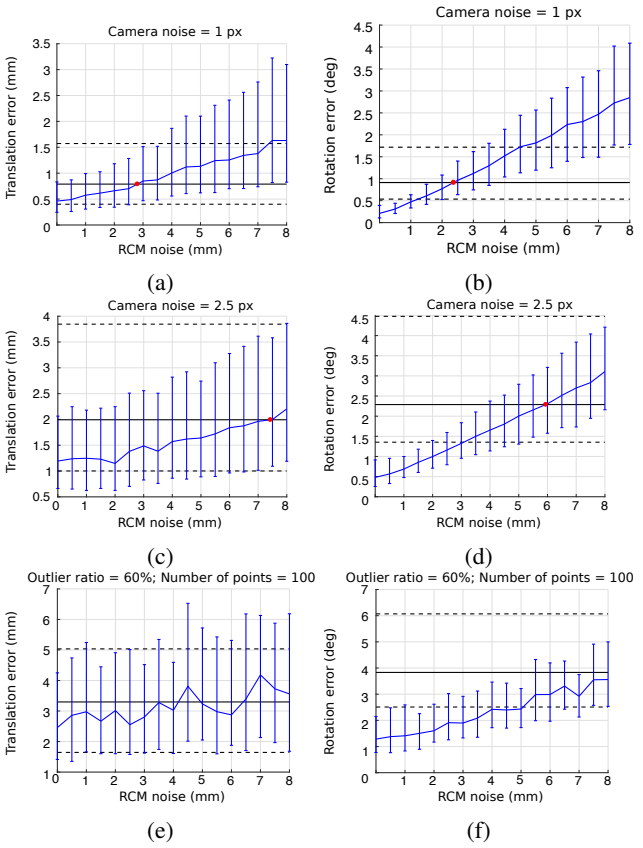


Fig. 3: Simulation results. PnP results (black) do not depend on RCM noise and thus are represented by a single distribution in each plot with bold line as median and dashed lines as quartiles. RCM-PnP results (blue) depend on RCM noise. The outlier-free experiments (a), (b), (d), (e) represent 1000 estimation trials with PnP and RCM-PnP, while the outlier contaminated experiments (c), (f) represent 100 estimation trials of PnP and RCM-PnP used with RANSAC.

B. RCM-SLAM with Robotic Prostatectomy Data

We estimate camera motion on a *in-vivo* video sequence from a robotic prostatectomy performed with the Da Vinci® Si system. Although its laparoscope is a stereo camera, we are interested in validating our monocular SLAM pipeline on a single channel, while using stereo data for comparison purposes. All monocular algorithms are tested on the right channel of the stereo scope. Kinematic data was not available, and camera calibration data was acquired after the procedure. We use a sequence of 44 frames (sub-sampled from a sequence of 88 frames) that performs a circular trajectory with the camera and goes back to roughly the initial point. The viewed scene is a prostate before being removed and the operating tools are in position to start the operation. We test the following algorithms on this sequence:

- **Monocular:** We follow the structure of Algorithm 1, using the conventional routines for unconstrained motion.
- **RCM-Monocular:** We follow the structure of Algorithm 1, using the routines for RCM constrained motion.

- **Stereo:** Stereo motion pipeline that follows Algorithm 1 for each channel using the methods for unconstrained motion. Additionally, stereo consistency is enforced during the bundle adjustment refinement step. This is done by representing all left camera poses in terms of right camera pose parameters, i. e. $\mathsf{T}_{i,L} = \mathsf{T}_{i,R}\mathsf{T}_s^{-1}$, where T_s is the stereo transformation from right to left camera.
- **ORB-SLAM2:** We use the monocular version of this open source method [20]. This is a significantly more sophisticated pipeline than Algorithm 1, that includes key-frame management, different local/global bundle adjustment threads, and a place recognition module for re-localisation.

Except for **ORB-SLAM2**, all algorithms are our own implementations. They all rely on SIFT descriptors [21] for feature matching (MATCH2D-2D,MATCH2D-3D functions). TRIANGULATION is performed using a classic SVD solution [22]. Only the triangulated points with less than 1.5 pixel re-projection error in all views are assigned to X and used in the following MATCH2D-3D call. All motion estimation algorithms (4-point, 5-point, RCM-PnP, PnP) are used inside a RANSAC robust estimator with the appropriate number of samples per iteration (respectively 4, 5, 2, 3). Given the very large amount of matched points over the video sequence, and for achieving a tractable computational effort on a macbook pro machine, we further filter the 3D points used during REFINE : (1) we only use 3D points that were matched consecutively for 5 or more frames; we only use 3D points that were matched against either the previous frame or 5 frames ago. While (1) ensures that the most reliable points are used, (2) selects matches from two distant frames to minimise drift. All the implemented methods are both tested in open loop and closed loop where the latest frame is matched against the closest frame in the previous trajectory. **ORB-SLAM2** has its own loop closing implementation.

The reconstructed trajectories for all methods are displayed in Fig. 5. Additionally we also plot for each trajectory, the projected laparoscope axis assuming the *aligned axis assumption*, i. e. along the $z-$axis of the camera. Given these projected axes we compute their distance to the closest RCM intersection point and display its distribution on Fig. 4. The **Stereo** trajectory (black), due to having a fixed baseline between the cameras is used as reference. It is worth noticing that we purposefully project the RCM axes on the right camera of the stereo system instead of stereo baseline midpoint, where it would be more likely located. We do this to show that the *aligned axis assumption* is still reasonably valid and all axes are still close to intersect. The **Monocular** implementation (blue) has the worst performance due to drift in rotation, translation, and scale. Its trajectory in open loop is also the one that least resembles a RCM constrained trajectory. Its loop closure is not able to sufficiently correct the initial large drift. We also note that due to the stochastic nature of RANSAC, methods can have different results on different runs. **Monocular** was the only one to have

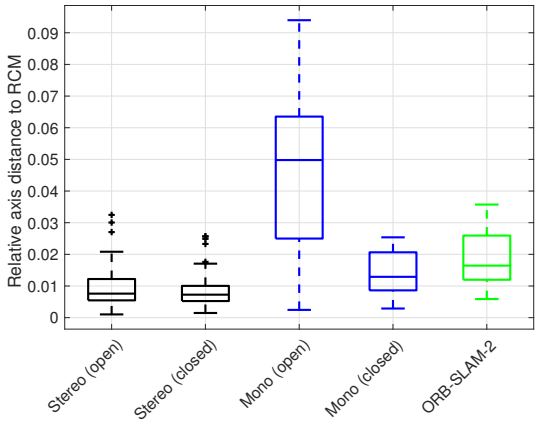
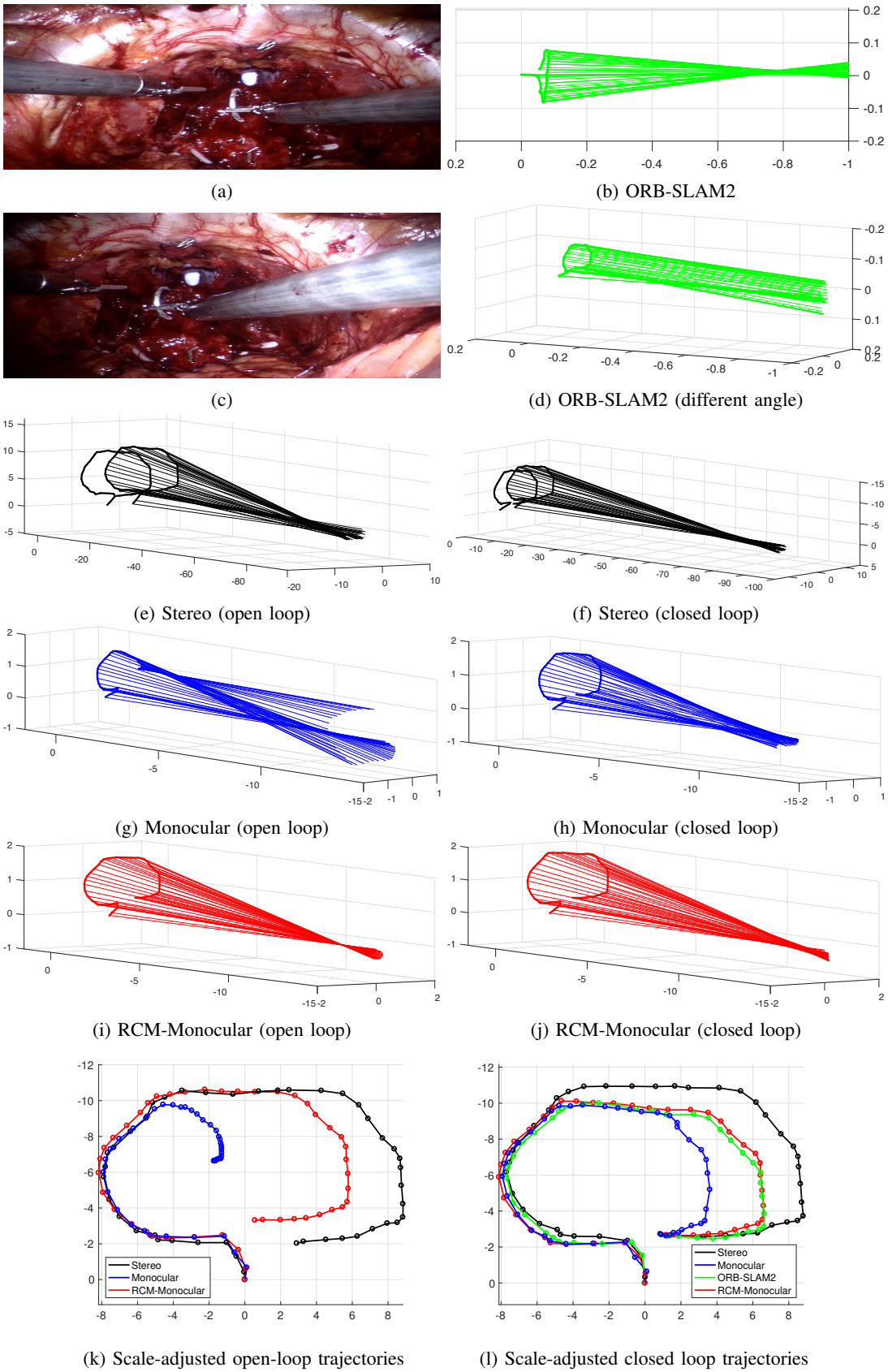


Fig. 4: Measuring how close each trajectory is to the *aligned axis assumption* by the distance between camera axis in each pose and the estimated RCM position. The distances are divided by the RCM distance to origin (for comparison of arbitrary scale reconstructions). With **RCM-Monocular** RCM alignment is strictly enforced and thus this distance is zero for all lines

significant differences, and to occasionally completely fail to perform loop closure. The displayed result is among the best obtained. **RCM-Monocular** (red) and **ORB-SLAM2** (green) have the most similar trajectories (with the former slightly closer to **Stereo**). However, if we analyse the projected RCM axes, **ORB-SLAM2** has an infeasible trajectory that does not comply with RCM constraints. It is worth noticing that the RCM is still very well constrained along the x-axis of the camera, but very badly along y-axis. The causes of this anisotropic behaviour are unknown. The significant contrast between **Monocular** and **RCM-Monocular** is only due to enforcing RCM constraints, as this is the only implementation difference between them. Additionally, for both **Monocular** and **Stereo**, the loop closure refinement makes the trajectory closer to the RCM constraint.

VII. CONCLUSIONS

We propose a SLAM pipeline for robotic surgery that estimates camera motion under RCM constraints. Following [1] we use an approximation that assumes perfect alignment between the camera and laparoscope axes, and that greatly simplifies the geometric relations between point correspondences across images. This enables building simple closed-form solutions and formulating optimisation problems with fewer variables. We further confirm that, in practice, the gains obtained by solving easier problems out-weight the induced approximation errors for a reasonable range of realistic conditions. Despite these results we believe that using more general RCM models (Fig. 2a) will be important to obtain increasingly more accurate reconstructions. In this sense, the *aligned axis assumption* is a very useful tool for formulating closed form solutions that provide reasonably accurate initial solutions that can be further refined in a subsequent step.



(a) (b) ORB-SLAM2

(c) (d) ORB-SLAM2 (different angle)

(e) Stereo (open loop) (f) Stereo (closed loop)

(g) Monocular (open loop) (h) Monocular (closed loop)

(i) RCM-Monocular (open loop) (j) RCM-Monocular (closed loop)

(k) Scale-adjusted open-loop trajectories (l) Scale-adjusted closed loop trajectories

Fig. 5: SLAM results for a prostatectomy camera trajectory. (a), (b) are two sample frames. In (e), (f) the axes units are millimitres, while in all monocular trajectories (b), (d), (g), (h), (i), (j) the units are an arbitrary scale up to the SLAM implementation choice. In (k), (l) the arbitrary scales of monocular trajectories are adjusted to the mm scale of stereo trajectories for visualisation purposes.

REFERENCES

[1] F. Vasconcelos, E. Mazomenos, J. Kelly, S. Ourselin, and D. Stoyanov, "Relative pose estimation from image correspondences under a remote center of motion constraint," *IEEE Robotics and Automation Letters*, 2018.

[2] A. Tewari, J. Peabody, R. Sarle, G. Balakrishnan, A. Hemal, A. Shrivastava, and M. Menon, "Technique of da vinci robot-assisted anatomic radical prostatectomy," *Urology*, vol. 60, no. 4, pp. 569–572, 2002.

[3] G. H. Ballantyne and F. Moll, "The da vinci telerobotic surgical system: the virtual operative field and telepresence surgery," *Surgical Clinics*, vol. 83, no. 6, pp. 1293–1304, 2003.

[4] R. H. Taylor, J. Funda, B. Eldridge, S. Gomory, K. Gruben, D. LaRose, M. Talamini, L. Kavoussi, and J. Anderson, "A telerobotic assistant for laparoscopic surgery," *IEEE Engineering in Medicine and Biology Magazine*, vol. 14, no. 3, pp. 279–288, 1995.

[5] L.-M. Su, B. P. Vagvolgyi, R. Agarwal, C. E. Reiley, R. H. Taylor, and G. D. Hager, "Augmented reality during robot-assisted laparoscopic partial nephrectomy: toward real-time 3d-ct to stereoscopic video registration," *Urology*, vol. 73, no. 4, pp. 896–900, 2009.

[6] K. Pachtrachai, F. Vasconcelos, F. Chadebecq, M. Allan, S. Hailes, V. Pawar, and D. Stoyanov, "Adjoint transformation algorithm for hand–eye calibration with applications in robotic assisted surgery," *Annals of biomedical engineering*, vol. 46, no. 10, pp. 1606–1620, 2018.

[7] N. Karlsson, E. Di Bernardo, J. Ostrowski, L. Goncalves, P. Pirjanian, and M. E. Munich, "The vslam algorithm for robust localization and mapping," in *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on*. IEEE, 2005, pp. 24–29.

[8] N. Snavely, S. M. Seitz, and R. Szeliski, "Photo tourism: exploring photo collections in 3d," in *ACM transactions on graphics (TOG)*, vol. 25, no. 3. ACM, 2006, pp. 835–846.

[9] P. Mountney, D. Stoyanov, and G.-Z. Yang, "Three-dimensional tissue deformation recovery and tracking," *IEEE Signal Processing Magazine*, vol. 27, no. 4, pp. 14–24, 2010.

[10] R. H. Taylor, J. Funda, D. D. Grossman, J. P. Karidis, and D. A. LaRose, "Remote center-of-motion robot for surgery," Mar. 14 1995, uS Patent 5,397,323.

[11] L. Dong and G. Morel, "Robust trocar detection and localization during robot-assisted endoscopic surgery," in *Robotics and Automation (ICRA), 2016 IEEE International Conference on*. IEEE, 2016, pp. 4109–4114.

[12] N. Aghakhani, M. Geravand, N. Shahriari, M. Vendittelli, and G. Oriolo, "Task control with remote center of motion constraint for minimally invasive robotic surgery," in *Robotics and Automation (ICRA), 2013 IEEE International Conference on*. IEEE, 2013, pp. 5807–5812.

[13] C. Doignon, F. Nageotte, and M. De Mathelin, "Segmentation and guidance of multiple rigid objects for intra-operative endoscopic vision," in *Dynamical Vision*. Springer, 2007, pp. 314–327.

[14] B. Triggs, P. F. McLauchlan, R. I. Hartley, and A. W. Fitzgibbon, "Bundle adjustmenta modern synthesis," in *International workshop on vision algorithms*. Springer, 1999, pp. 298–372.

[15] B. M. Haralick, C.-N. Lee, K. Ottenberg, and M. Nölle, "Review and analysis of solutions of the three point perspective pose estimation problem," *International journal of computer vision*, vol. 13, no. 3, pp. 331–356, 1994.

[16] M. A. Fischler and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, 1981.

[17] F. Vasconcelos, D. Peebles, S. Ourselin, and D. Stoyanov, "Similarity registration problems for 2d/3d ultrasound calibration," in *European Conference on Computer Vision*. Springer, 2016, pp. 171–187.

[18] D. Nister, "An efficient solution to the five-point relative pose problem," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 26, no. 6, pp. 756–770, June 2004.

[19] F. Moreno-Noguer, V. Lepetit, and P. Fua, "Accurate non-iterative o (n) solution to the pnp problem," in *Computer vision, 2007. ICCV 2007. IEEE 11th international conference on*. IEEE, 2007, pp. 1–8.

[20] R. Mur-Artal and J. D. Tardós, "Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras," *IEEE Transactions on Robotics*, vol. 33, no. 5, pp. 1255–1262, 2017.

[21] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vision*, vol. 60, pp. 91–110, November 2004.

[22] R. Hartley and A. Zisserman, *Multiple view geometry in computer vision*. Cambridge Academic Press, 2003.