Tissue Tracking and Registration for Image-Guided Surgery

Michael C. Yip*, David G. Lowe, Septimiu E. Salcudean, Robert N. Rohling, and Christopher Y. Nguan

***Abstract*—Vision-based tracking of tissue is a key component to enable augmented reality during a surgical operation. Conventional tracking techniques in computer vision rely on identifying strong edge features or distinctive textures in a well-lit environment; however endoscopic tissue images do not have strong edge features, are poorly lit and exhibit a high degree of specular reflection. Therefore, prior work in achieving densely populated 3-D features for describing tissue surface profiles require complex image processing techniques and have been limited in providing stable, long-term tracking or real-time processing. In this paper, we present an integrated framework for accurately tracking tissue in surgical stereo-cameras at real-time speeds. We use a combination of the STAR feature detector and binary robust independent elementary features to acquire salient features that can be persistently tracked at high frame rates. The features are then used to acquire a densely-populated map of the deformations of tissue surface in 3-D. We evaluate the method against popular feature algorithms in *in vivo* animal study video sequences, and we also apply the proposed method to human partial nephrectomy video sequences. We extend the salient feature framework to support region tracking in order to maintain the spatial correspondence of a tracked region of tissue or a medical image registration to the surrounding tissue. *In vitro* tissue studies show registration accuracies of 1.3–3.3 mm using a rigid-body transformation method.**

***Index Terms*—Feature tracking, image-guided surgery, image registration, salient features, stereoscopy, surface reconstruction.**

I. Introduction

A. Clinical Problem

IMAGE-GUIDED minimally-invasive surgery has received significant interest in recent years due to rapid progression in the fields of robot assisted surgery, medical imaging technology, and augmented reality. Augmented reality enables the identification and augmentation of subsurface tissue (e.g., lesions or vasculature) in the surgical camera images and can provide stable and persistent medical image registrations, improve surgical margins, and reduce the time required in the operating room [51].

As examples, partial nephrectomy and radical prostatectomy are two surgical procedures that would benefit from providing a persistent medical image registration to the surgeon's view. During partial nephrectomy, the kidney is dissected from the surrounding tissue in order to allow a surgeon to clamp the renal artery and stop blood flow prior to tumor resection. A laparoscopic ultrasound image or an external ultrasound image can be acquired intraoperatively and registered to the kidney. Since the kidney is mobile, a method for maintaining a registration would allow for subsurface tumor boundaries to be maintained in the camera images during resection. In radical prostatectomy, an exposed prostate is imaged with transrectal ultrasound and then dissected from surrounding tissues prior to resection. Since contact is lost between the ultrasound transducer and the prostate after mobilization and new ultrasound images cannot be attained, it is critical to maintain the ultrasound-image/prostate registration over time. Therefore, local tissue tracking is essential for maintaining such registration in a surgical environment (Fig. 1).

The following are the requirements for tissue tracking and medical image registration.

1) A dense set of trackable locations on the tissue surfaces is required to describe local tissue deformation and movement. Tracking should run in real-time.
2) Tracked locations must be repeatedly found in the endoscopic images in the presence of camera movement, human motion, and instrument–tissue interaction.
3) Tissue tracking should enable medical images to stay registered to the endoscopic tissue images over time while avoiding drift.

B. Prior Work on Endoscopic Tissue Tracking and Registration

Tracking algorithms developed for natural scenes and urban environments rely on the assumptions that the environment

Manuscript received May 05, 2012; revised July 25, 2012; accepted July 31, 2012. Date of publication August 09, 2012; date of current version October 26, 2012. This work was supported in part by the Natural Sciences and Engineering Research Council of Canada (NSERC), in part by the Canadian Institutes of Health Research (CIHR), and in part by the CIHR Knowledge Translation Fund, as well as from the C. A. Laszlo Chair held by Professor Salcudean. *Asterisk indicates corresponding author.*

*M. C. Yip is with the Electrical and Computer Engineering Department, University of British Columbia, Vancouver, BC V6T 1Z4 Canada (e-mail: myip@ece.ubc.ca).

D. G. Lowe is with the Computer Science Department, University of British Columbia, Vancouver, BC, V6T 1Z4 Canada (e-mail: lowe@cs.ubc.ca).

S. E. Salcudean are with the Electrical and Computer Engineering Department, University of British Columbia, Vancouver, BC, V6T 1Z4 Canada (e-mail: tims@ece.ubc.ca).

R. N. Rohling are with the Department of Electrical and Computer Engineering and the Department and Mechanical Engineering, University of British Columbia, Vancouver, BC, V6T 1Z4 Canada (e-mail: rohling@ece.ubc.ca).

C. Y. Nguan is with the Urology Department, Vancouver General Hospital, Vancouver, BC, V5Z 1M9 Canada (e-mail: chris.nguan@ubcurology.com).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TMI.2012.2212718

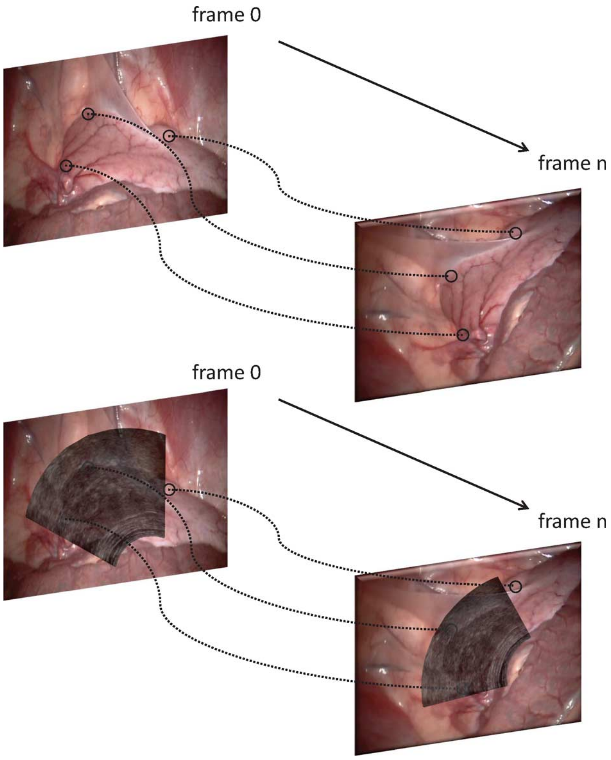


Fig. 1. Repeatedly identifying and tracking tissue locations in endoscopy (top), and maintaining an image registration based on tracked movement (bottom).

exhibits strong edge features, low shadowing and lighting changes, visually distinctive textures, and is generally not deformable [57]. However, tissue surfaces are deformable and are constantly affected by patient motion due to breathing and heartbeat; furthermore, interactions with surgical instruments cause significant tissue deformation. Tissues have visually nondistinctive textures and therefore image patches are difficult to distinguish from their local environment. Finally, tissue images exhibit considerable specular reflection caused by the wet tissue surfaces, and endo-cavity lighting creates large shadow effects and dynamic lighting conditions.

The first major challenge is identifying trackable features from endoscopic images [46], [47]. Correlation-based tracking has been proposed [13], [32], [33], [31], [42], but has been limited by the lack of sensitivity to poor texturization. B-spline polynomial fitting [18], homography transformation [2], and radial basis functions [36], [38]–[40], [35] have been shown to be effective in capturing the low-order deformations of a small region of the beating heart at real-time speeds (50 Hz); however, higher order-deformation tracking using these methods becomes nontrivial and computationally expensive. In their current state, they have been found to only perform accurate tracking in a short term of approximately 3 s without reinitialization [37]. In order to capture high-order deformations, a large number of locations need to be individually tracked over time. Lightweight corner and region detectors such as the Shi–Tomasi features [43] and features from accelerated segment test (FAST) features [41] have been proposed for endoscopic images [27], [29], [12] and can perform at real-time speeds; however, the correlation methods used to match and localize feature points between consecutive images are prone to drift. The features that exhibited the highest density and temporal persistency are the scale invariant feature transform (SIFT) and speeded up robust features (SURF) [21], [22], [26], [28]. However, these methods were unable to achieve the real-time speeds required for an operating room setting. To maintain real-time performance, methods involving tracking only a small region of the image have been proposed [52]. Other methods such as the ones described in [24] used GPU acceleration to acquire SIFT and SURF features for tissue reconstruction. However, existing GPU implementations still rely on trading computations between the GPU and the CPU [9], [56], [44], and real-time performance on high-resolution, feature-dense images is still difficult [45].

The second major challenge is long-term tracking. Features found in previous frames are eventually obscured by camera motion, occlusion, illumination change, specular reflection, and instrument interaction with tissues. Although the use of camera motion estimation for iterative localizations and tracking of features [27], [29], [12] has achieved some success, it relies on assumptions of a nondeforming scene. Otherwise, the tissue tracking literature has shown success only in tracking short tissue sequences (e.g., [37], [27]). Wengert *et al.* [54] and Wang *et al.* [53] maintained a database of tracked and untracked features over multiple frames, and used camera position estimates to preserve the feature sets in subsequent frames. Grasa *et al.* [12] developed a history-preserving feature tracking method using the FAST detector; however, only sparse clouds were tracked (45 features tracked per frame) at 9 frames/s. These methods required camera pose estimation as well as correlation-based template matching. Therefore, a tissue tracking framework for long-term tracking that does not depend on camera modeling or pose estimation is necessary, and different feature detectors for the purpose of long-term tracking need to be extensively evaluated.

The third major challenge is maintaining medical image registration through tissue tracking. There has been a gap in the literature between the application of tissue tracking in surgical environments and the efforts in medical image registration. Fiducial or marker-based systems [30], [11], [49], [48] and magnetic tracking systems [25] provide registration and tracking techniques, but rely on the depositing of artificial markers onto the tissue. Burschka *et al.* [5] showed that endoscopic image features could be used to maintain registrations in nasal surgery; however, *in vitro* tissues were marked using ink in order to create distinguishable landmarks for tracking in the endoscopic camera. Therefore, there is much room for improvement for maintaining medical imaging registrations to endoscopic images, and a noninvasive method would be very beneficial.

C. *Contributions*

This paper addresses the challenges of tissue tracking and registration through the following contributions.

1) We present the use of the STAR detector and the binary robust independent elementary features for real-time dense tissue tracking. We provide evaluations of performance (speed, stability, and accuracy) of popular feature detectors for both 2-D and stereoscopic 3-D tissue tracking.

2) We develop a history-preserving framework for tracking tissue, and evaluate the feature detectors for long-term tracking.
3) We extend the history-preserving framework for maintaining a medical image registration in the endoscopic images over time. We present preliminary data on maintaining a registration for various tissue types.

II. Methods

A. Choice of Feature Detector

In order to approach real-time performance while preserving feature saliency, we chose a modified version of the Center Surrounded Extremas for Real-time Feature Detection (CenSuRE) feature detector [1] called STAR [55] and the binary robust independent elementary feature (BRIEF) feature descriptor [6]. The CenSuRE feature detector and the BRIEF feature descriptor are especially fast salient feature algorithms as they are both intensity-based, binary methods that evaluate square patches of an image. Furthermore, binarized methods reduce floating point operations and can use boolean operations for feature definition and feature matching. Comparisons with CenSuRE and BRIEF against popular feature descriptors in natural scenes and urban environment are provided in their original papers [1], [6]. The STAR detector is simply an overlay of a CenSuRE kernel with a 45° rotated kernel, which better approximates the Laplacian of Gaussians kernel, improving robustness at little cost to performance. Furthermore, by varying the STAR kernel size, we can identify patches of varying scales at which BRIEF descriptors can be extracted, effectively making the feature scale-invariant. The choice of STAR over other detectors such as Shi–Tomasi [43] and FAST [41] is based on the report that Laplacian of Gaussian estimators have better feature repeatability and saliency [50]. We have shown previously that STAR and BRIEF are able to track tissue in endoscopic images [59].

B. Temporal Feature Matching

There are six main steps to feature detection and matching (Fig. 2). They are described in [59] and are outlined below.

1) **Capture [Fig. 2①]:** Capture an image from the surgical camera.
2) **Gaussian Smoothing [Fig. 2②]:** Perform a preprocessing step with a 3 × 3 Gaussian smoothing kernel.
3) **Feature Extraction and Detection [Fig. 2③]:** Extract image features from the current frame that can be used as local landmarks for tracking.
4) **Feature Matching [Fig. 2④]:** Match the features to a list of features extracted from previous frames, in order to determine their movement in the scene. Various techniques can be applied to narrow down the number of possible matches, reducing unnecessary descriptor comparisons and improving performance, as well as reducing the possibility of incorrect matches [7], [20], [57]. Feature descriptors are compared only if they have similar characteristic scales, similar characteristic orientations, and are in close proximity. Furthermore, two features are only considered matched if the distance between their descriptors is significantly smaller than other possible matches. Neighboring features are expected to move in similar directions and distances due to the the nature of deforming tissue, and therefore false positive matches of temporal feature matches can also be filtered by examining neighborhood feature movement. The details of implementing these filters are given in the Appendix A.
5) **Updating of Feature Lists [Fig. 2⑤]:** Features are independently tracked over time by maintaining a list of features that have been previously found. All previous features that were matched are updated with the new feature location and descriptor, and the remaining new, unmatched features are appended to the list.
6) **Deletion of Lost Features [Fig. 2⑥]:** Whenever the ratio between the number of times a feature has been found and the number of frames since first detection falls beneath a threshold, the feature within the list is deleted. This ensures that the list only maintains features that are deemed to be persistent within the scene. Since we do not want to throw away new features as quickly as they appear, we wait until 10 frames after their first detection to evaluate whether they are to be deleted. This wait time was chosen by preliminary experiments, as was the threshold for deletion, set to approximately 40%, above which there was a significant drop-off of the stability of feature points.

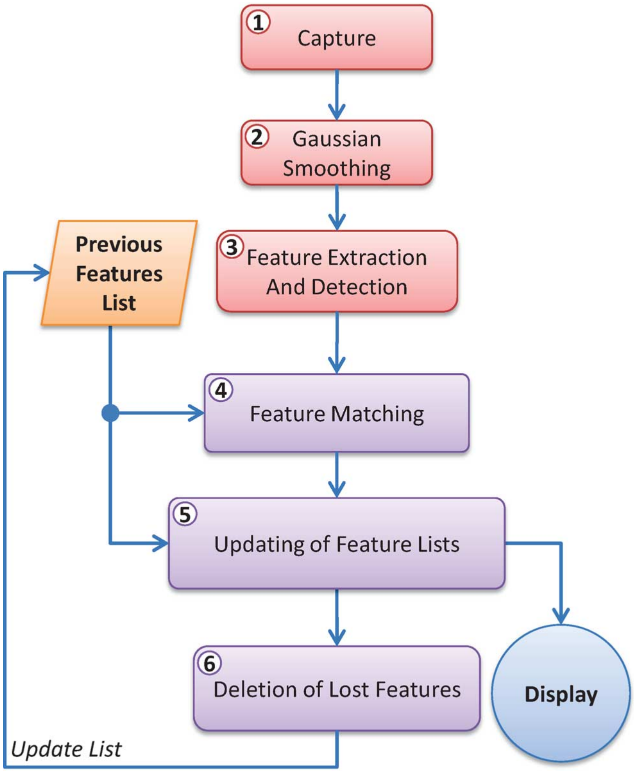

① Capture
② Gaussian Smoothing
③ Feature Extraction And Detection
Previous Features List
④ Feature Matching
⑤ Updating of Feature Lists
⑥ Deletion of Lost Features
Display
Update List

Fig. 2. Flowchart depicting the proposed feature tracking framework on a single image. Features are extracted in the current frame, and matched to features extracted in previous frames. Matched features are updated, and new features are saved. Feature that are not stable are deleted.

C. 3-D Depth Estimation

Given the feature tracking solution we proposed, we extend its function into stereo depth estimation and dense 3-D point localization for reconstructing depth maps of the scene. Because we have two dense feature populations for the left and right stereoscopic channels, we match features between the channels to establish a stereo-triangulated point. The strategies described for temporal feature matching can be used again for stereo matching during depth estimation (filtering by scale, orientation, proximity, and ratio of descriptor distances, as described in detail in Appendix A). Features can be tracked in 3-D by first performing 2-D tracking on one channel, and then taking the tracked features and performing stereo-matching with the other channel's features.

D. Region Tracking and Registration

We first assume that a medical image (or volume) registration is already performed within a localized region in the stereoscopic images. This can be achieved using a method such as the one described in [58]. A region is then selected on the tissue surfaces that appears closest to the center of the registered image/volume, and stereoscopically-matched features that are found within this region are saved to a separate feature list, as are their 3-D locations. For the sake of clarity, we will call this list the object feature list, as the methods we propose are similar to those used for object detection and tracking.

We present a visual flowchart of our proposed region tracking and registration framework in Fig. 3, depicting a single iteration. The steps are presented below.

1) **Object Feature Matching [Fig. 3①]:** In order to match the selected region's features to the features in the scene, two strategies can be used. First, we can rely on the matching parameters described in II-B-4 to match object features to features that have been temporally tracked and stereoscopically matched. This method provides redundancies in temporal and stereoscopic matches that can be used for outlier rejection. A second strategy is to give each scene feature a unique ID that it retains as it is tracked from frame to frame. Then, temporal tracking of scene features will provide the motion of the object features through their identification via the scene features' IDs, which is more computationally efficient. An attempt is made to match any object feature that is not matched temporally with stereo-matched scene features as described previously, based on scale, orientation, proximity, descriptor distance ratio, and neighborhood consistency.

2) **Acquire Registration [Fig. 3②]:** A temporal registration T requires a set of object feature locations in 3-D, denoted by X, to be matched to another set of 3-D feature locations in the current frame, denoted by Y, such that $Y = TX$. We use Random Sample Consensus [8] (RANSAC) in order to perform outlier rejection and identify the best registration using only a subset of the N points [53]. We consider outliers to be those feature points that fall outside the registration model by over five pixels of error in stereo-matching. The least squares method to find T was performed using Horn's quaternion-based method [14]. A minimum of three points are required in Horn's registration algorithm in order to acquire a rigid transformation. For simplicity, we make the assumption that the tissue is moving rigidly resulting in a homogeneous transformation.

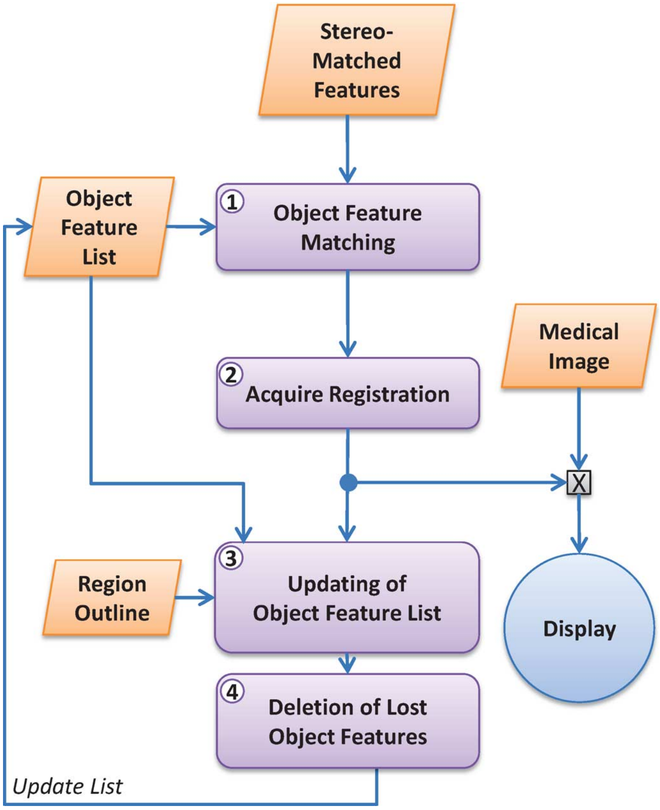


Fig. 3. Proposed object tracking and registration framework. An object is defined by the user or a detection method in frame 0, and features within the region are saved as object features. Features tracked in subsequent frames are matched to the object features to keep the object features up to date, and the new features in the tracked region are appended to the object feature list. All object features, registered images, and region outlines are kept in the frame 0 coordinate system to avoid drift from successive transformations.

3) **Updating of Object Feature List [Fig. 3③]:** To avoid registration drift, we need to update object features differently than when tracking features temporally. Given a set of object features in frame 0 and their locations X_0, we can identify their locations in the current frame n, X_n such that $X_n = {}^{n}T_0 \cdot X_0$. We do not update the feature locations based on the matches from X_n; rather, we only update their feature descriptor vectors, and the transformation ${}^{n}T_0$. By doing so, we are able to maintain a base frame of reference for all subsequent frames. Therefore, for the following frame at $n+1$, we first transform the features from the original object frame to the last frame n using ${}^{n}T_0$. Then we calculate the incremental transformation ${}^{n+1}T_n$ to transform the object features to the new frame. Therefore, ${}^{n+1}T_0 = {}^{n+1}T_n \cdot {}^{n}T_0$.

With time, features that were originally used as anchors will eventually be lost due to image noise and erroneous matches; therefore, the outline of the image patch that originally identified the anchor features is also transformed from frame to frame, and any new feature that appears within the outline will be added as an anchor point. We calculate the inverse transform $^{0}T_{n+1} = {}^{n+1}T_{0}^{-1}$ in order to reproject them into the original object frame n. This avoids the drift of previously saved object features caused by successive transforms.

4) **Deletion of Lost Object Features [Fig. 3④]:** Since the list of object points will grow indefinitely without applying some mechanism for feature deletion, we set a threshold number of anchor points to track. In practice, we only keep the most stable 500 features and their 3-D locations.

III. Experiments

A. Temporal Tracking

To evaluate the proposed method in tissue tracking, we compare the STAR+BRIEF algorithm to two of the *de facto* standards for salient feature detectors in the literature, SIFT and SURF. We follow the formulation of the BRIEF feature descriptor described in [6], choosing a feature vector length of $N = 256$ bits and a patch size of $S = 25$, and we follow the original papers [20] and [3] for the implementations of SIFT and SURF respectively. We use three octaves with three levels each for the SIFT and SURF implementations with the first octave being the original image resolution. We chose a Hessian threshold value of 50 for the SURF implementation. For the STAR detector, we constructed the two successive center-surrounded bi-level kernels to have an outer edge length of eight pixels, an inner edge length of four pixels, and a scale space pyramid of nine levels between 1.0–5.0 in increments of 0.5. The operating parameters of the algorithm were experimentally chosen as a Harris corner response threshold of 2, a nonmaximal extrema suppression of 5, and a line suppression ratio of 10.

Evaluation of the salient feature algorithms will be based on several criteria.

1) **Speed of algorithm:** the time required for feature detection, descriptor extraction, and matching with previously found features.
2) **Average number of features found in each frame:** this can vary significantly depending on video characteristics (e.g., size and resolution, noise, motion blurring, shading, etc.). However, it is useful in that it provides an idea of a feature detector's ability to densely characterize a tissue surface, and it allows for comparisons of densities between different feature detectors.
3) **Percent of features matched between consecutive frames:** Calculated as the ratio between number of features found and the number of features matched to previously saved features in each frame.
4) **Average lifetime of feature:** The number of frames between a feature's first detection and its deletion.
5) **Percentage of time features are found:** Evaluated as the ratio of frames in which a certain feature is found to the number of frames since its first detection. Given that features will often flicker in and out of an image due to video artifacts and noise, this evaluation will provide a measure of the temporal stability of features.
6) **Average size of static feature list:** The number of saved features that new features will be matched to.
7) **Localization accuracy and drift of selected features:** Since there is no ground truth available for the *in vivo* video sequences, we used the approach of Kalal *et al.* [15] to calculate the forward–backward error of select features. Features are tracked in frames as they move forwards in time, and at time n, the video frame sequences are reversed and the features are subsequently tracked backwards in time until the beginning of the video. Performing this forward–backward tracking essentially allows us to investigate how likely a feature is to drift within the image sequences due to feature mismatches in frames. A perfect feature correspondence would be achieved when the position of a feature moving forward and backwards in time line up at every time step. At any point in time, if a feature is mismatched to a different location, it is likely that this error is continued into subsequent frames resulting in drift.

B. 3-D Depth Estimation

In order to evaluate the feature tracking framework for stereoscopic depth estimation, we will evaluate the following parameters.

1) **Speed of the stereo matching:** The time required to identify matches between features found in the left and right frames given the two lists of features.
2) **Number of features matched across channels:** This value will differ depending on the video sequence photometric properties.
3) **Percentage of features stereo-matched from a single frame:** Evaluated as the ratio of the number of features that were matched between the left and right camera frames, and the number of features found in the left camera frame.

C. Region Tracking and Registration

For tissue region tracking experiments, we set up an *in vitro* experiment on three different tissue types: kidney (bovine), heart (bovine), and liver (porcine). On each tissue, we placed a 2-mm-diameter steel bead fiducial. These fiducials are used to represent locations within the tissue that can be segmented clearly in the camera images and will be used to test the accuracy of registration.

We placed an endoscopic stereo camera approximately 5–10 cm away from the tissue such that the fiducials could be seen in both stereo channels. We then moved the tissue, with rotation and translation in all three dimensions, taking care to keep the fiducials continuously seen in all three channels.

Since we do not want the fiducials to provide any help in determining a registration, we automatically remove all the extracted features whose templates would overlap with the fiducial locations, based on their patch location and size. This ensures that the fiducials, which represent a strong trackable location, will not improve the registration. In the first frame of tracking, we manually identified the fiducial locations in the left and right camera images and we acquired their locations in 3-D. In subsequent frames, we applied the chain of transformations to the original fiducial locations in order to predict the motion of the fiducials over time. We compared this to tracking of each fiducial individually, using a 31 × 31 normalized cross-correlation

TABLE I
PARAMETERS FOR TEMPORAL TRACKING, STEREOSCOPIC MATCHING, AND OBJECT TRACKING

		Value		
Parameter	Symbol	Temporal Tracking	Stereo Matching	Object Tracking*
Scale Threshold	κ	$\log(2.0)$	$\log(\sqrt(2))$	$\log(2.0)$
Orientation Threshold	Θ	$\pi/18$	$\pi/12$	N/A
Proximity Threshold	δ	0.2*rm image_width	0.5*image_width on x-axis 0.05*image_height in y-axis	0.5*image_width
Descriptor Distance Ratio	λ	0.5	0.5	0.5
Difference in movements	γ	$2 * \log(1.5)$	N/A	$2 * \log(1.5)$
Difference in angles	ϵ	$\pi/18$	N/A	$\pi/18$

* only uses these parameters when trying to re-establish a registration after it has been lost; otherwise use feature IDs

window over the fiducial location, to estimate their locations in each stereo channel and in 3-D.

D. Apparatus and Test Data

We used a PC with an Intel Core i7 CPU 960 at 3.20 GHz with 12 GB RAM, on the Windows 7 64-bit platform. No GPU acceleration was performed, and all the processing was kept on the system memory and processor. Video sequences were read frame-by-frame from the hard disk.

To evaluate temporal tracking and 3-D depth estimation, we used a range of videos of intraoperative laparoscopic porcine studies from an Imperial College London *in vivo* dataset, which can be found online, along with associated camera calibration files [16]. These videos represent a wide array of scene motions that cover camera translation, scale change, camera rotation, multiple viewpoints (i.e., affine transformation), and tissue deformation. The videos are as follows.

1) **Translation**: Abdominal cavity just after insufflation. The surgeon moves the endoscopic camera to approximate translation. Resolution: 320 × 240 (upscaled to 640 × 480 through linear interpolation in order to detect small features); Length: 1050 frames.
2) **Rotation**: Abdominal cavity just after insufflation. The surgeon rotates the endoscopic camera. Resolution: 640 × 480; Length: 710 frames.
3) **Series**: Abdominal cavity just after insufflation. The surgeon moves the endoscopic camera in a series of movements involving translation and scaling. Resolution: 640 × 480; Length: 1200 frames.
4) **Heartbeat**: Open-chest procedure with an exposed heart with significant instrument footprint in images. The endoscopic camera is held stationary, imaging a rapid heartbeat. Resolution: 360 × 288 (upscale to 720 × 576); Length: 650 frames.

We also tested that the temporal tracking of tissue works with human patient data. The videos used for evaluation were captured during partial nephrectomy operations. Patients were recruited with signed consent after approval from the University of British Columbia Clinical Research Ethics Board (Certificate Number H08-02798). The video sequences are as follows.

1) **Regular Motion**: A partially exposed kidney is viewed by a stationary camera. A regular heartbeat results in a regular kidney motion of a few millimeters. Resolution: 640 × 480.
2) **Deformation**: The exposed kidney with a visible tumor is repeatedly deformed by the surgical instruments. Resolution: 640 × 480.
3) **Cauterization**: The surgeon performs cauterization on the several parts of the surface of the kidney, causing smoke to cloud and waft through the surgical images. Resolution: 640 × 480.
4) **Postdissection**: After tumor dissection, the kidney is repositioned by the surgical instruments to show the dissection location. More blood obscures the tissue surfaces resulting in the loss of tissue texture information and an increase in specular reflection. Instrument occlusion also covers larger areas than in other videos. Resolution: 640 × 480.

The videos used for region tracking and registration evaluation were captured using a da Vinci SI laparoscopic stereo camera. The videos that were used are of the kidney (250 frames ≈8 s), heart (650 frames ≈20 s), and liver (700 frames ≈23 s) undergoing translation, rotation, and scaling. Videos were cropped to a resolution of 560 × 352. These datasets can be found online [17]. We aim to keep these videos online as a repository in order to allow other researchers to compare their algorithms using common data.

We performed a two sample t-test $(p < 0.05)$ for each measured evaluation criteria to compare STAR+BRIEF to SURF, STAR+BRIEF to SIFT, and SURF to SIFT. Camera calibration was performed using the Matlab calibration toolbox by Bouguet [4]. Our camera calibration of the da Vinci SI cameras was found to have approximately a 5 mm baseline and near-identical left and right camera orientations.

Table I summarizes the parameters used for tissue tracking, stereoscopic matching and object tracking. The motivation behind selecting the value of each parameter is given in the Appendix B.

IV. RESULTS

A. Temporal Tracking Results

Fig. 4 shows samples of each video test performed with an overlay of extracted and matched features. SIFT features are shown to be the most numerously detected, and although the percentage of features matched temporally is less than that of the other feature types, it still offers a higher density of tracked features within each frame (Fig. 5). Both STAR+BRIEF and SURF have similar feature densities, but STAR+BRIEF is seen to have a slightly higher percentage in temporal matching (Fig. 6) while

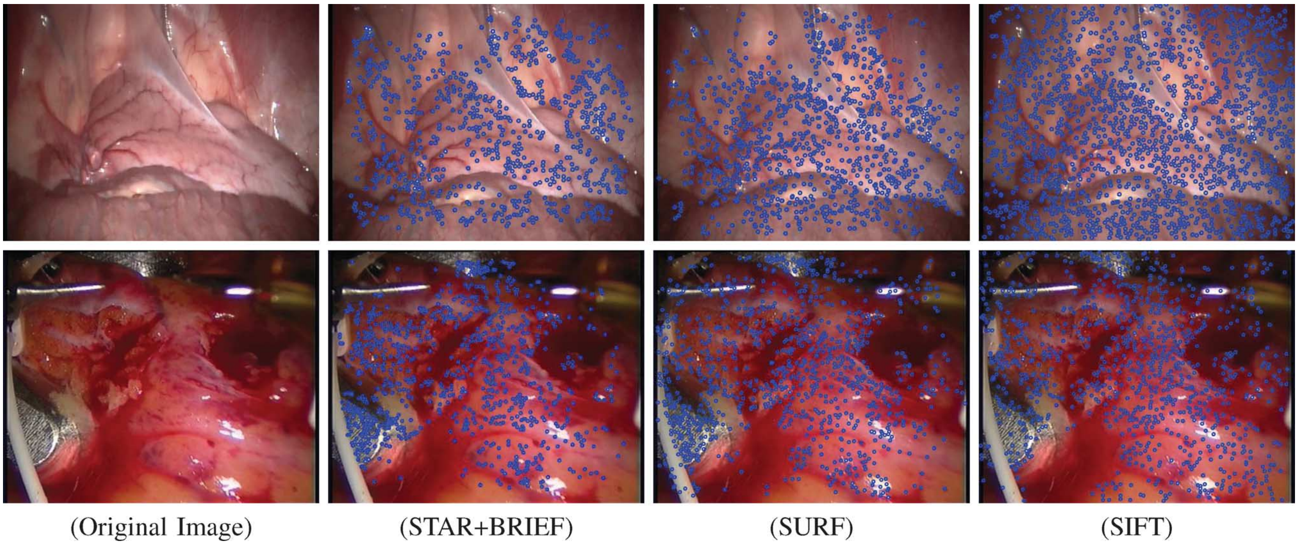


Fig. 4. Screen captures of frame 20 of the series (top row) and heartbeat video (bottom row). Circles represent feature locations.

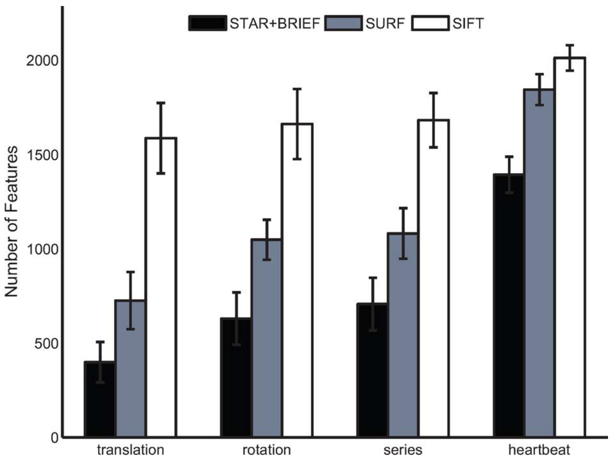


Fig. 5. Number of features found per frame.

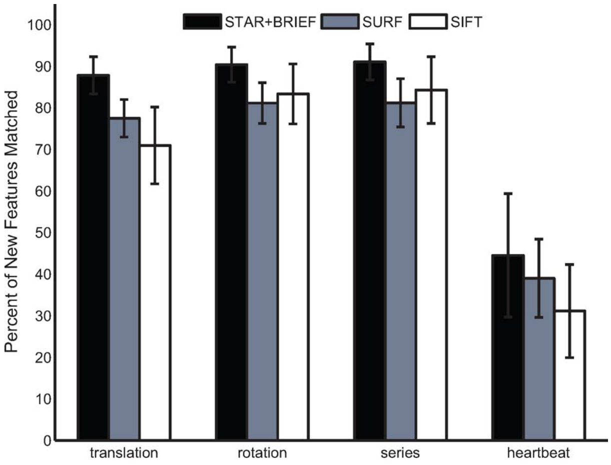


Fig. 6. Percentage of these features that are matched to a previously detected feature.

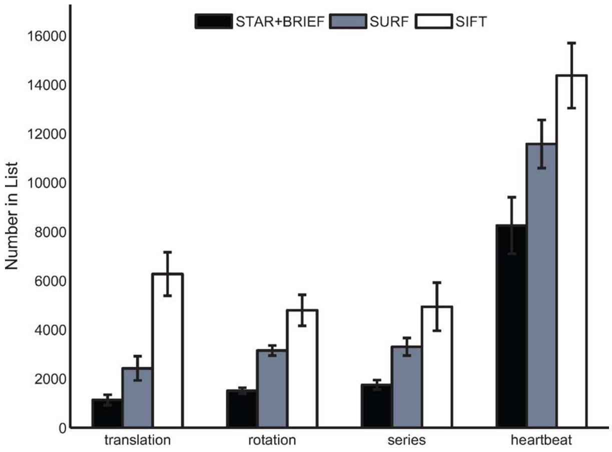


Fig. 7. Number of features found previously that is kept in a history-preserving feature list.

keeping track of the least number of previously matched features (Fig. 7). This reduces the necessary computations required for matching between features as thus reduces the computation time.

Fig. 8 shows the average percentage of features that are deleted at every frame. A higher percentage represents the loss of history for a previously saved feature; if the same feature is found in subsequent frames, it will be considered a brand new feature. A reduction of this value will improve the longterm persistency of features within a video sequence. STAR+BRIEF is shown to have the lowest percentage of features deleted, followed by SURF, and subsequently SIFT.

To elucidate the prevalence of features among consecutive frames, we need to investigate just how often they are found and subsequently matched in following frames after their first detection and extraction. In order to do this, we generate a normalized histogram shown in Fig. 9, depicting the number of features during runtime that are found in at least a certain percentage of the frames following initial detection. Since features are deleted from the list if they are found in less than 40% of all the frames, we only keep track of features that are found in greater than 40%

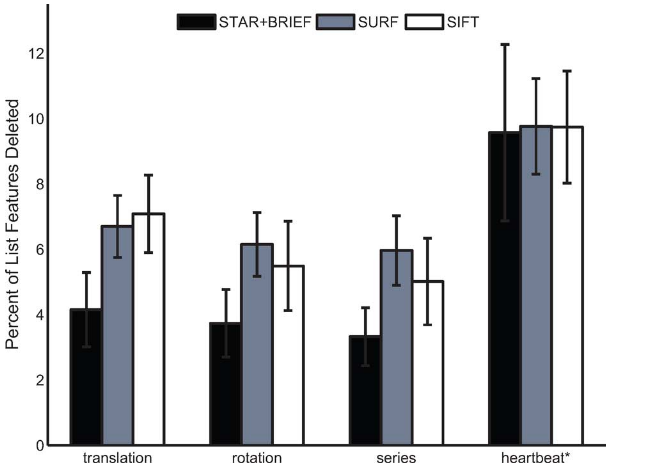
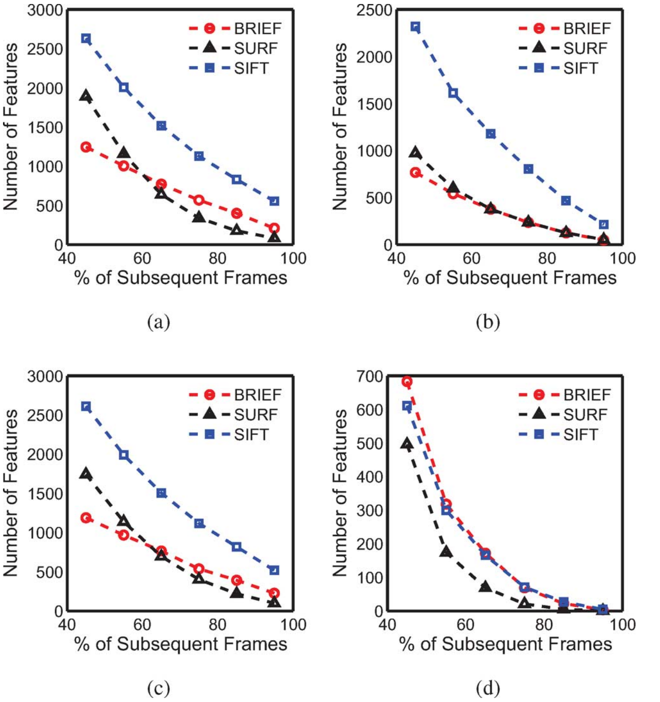


Fig. 8. Percentage of features from the saved list that are removed every frame due to a low frequency of finding a suitable temporal match.



(a)

(b)

(c)

(d)

Fig. 9. Histograms depicting the number of features that are found in a certain percentage of subsequent frames. The graph is cumulative such that feature numbers drop off as the ratio between times found and total lifetime increases. Evaluations are performed on the (a) translation, (b) rotation, (c) series, and (d) heartbeat videos.

of all the frames. STAR+BRIEF features are found to be consistently prevalent in scenes in a similar manner as SIFT, whereas SURF features are found to be less persistent.

Fig. 10 shows an example of a feature tracked in both forwards and backwards time for each tracking algorithm. The feature chosen to be tracked is one that appeared at highest frequency moving forward in time and therefore suggestive of the persistency of the feature in the video sequence from the beginning to the end. Fig. 10 shows that each tracking algorithm manages to track the features well; matching forwards and backwards motions will fall on the exact same pixel location in both forwards and backwards time. Fig. 11 shows the proposed STAR+BRIEF feature tracked in Fig. 10 over the entire sequence of the videos at 100 frame intervals. The feature remains true to its original location over the duration of tracking.

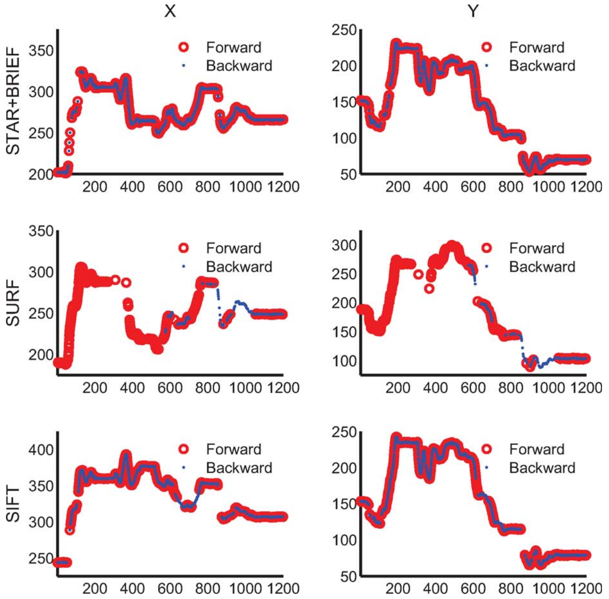


Fig. 10. Sample feature location is chosen for each feature tracking algorithm, and the feature's pixel-location is tracked from the beginning to the end of the video (red). Subsequently, starting at the end of the video, the feature is tracked backwards towards the start of the video (blue). Due to frames where a feature is momentarily lost, there are gaps within the tracking in both directions. X-axes represent time (sec) and Y-axes represent pixel location.

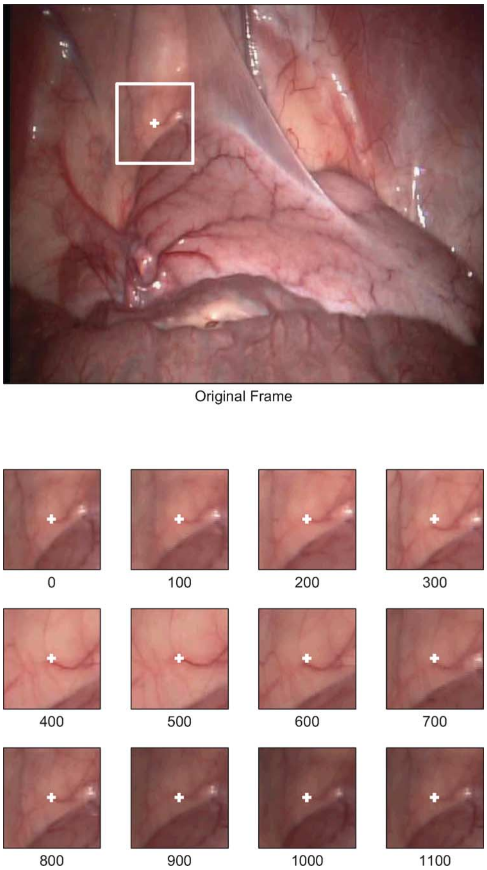


Fig. 11. Example of a STAR+BRIEF feature being tracked over time.

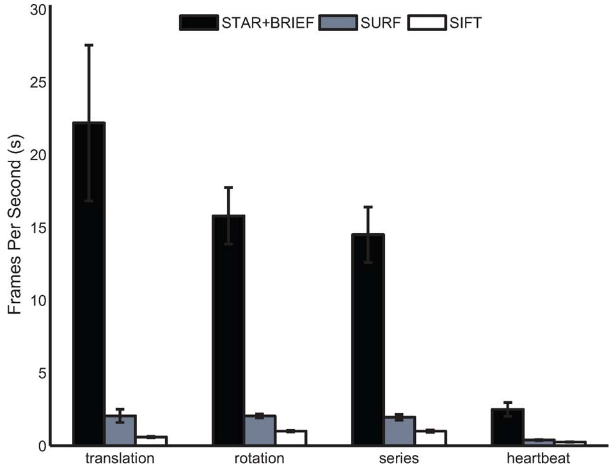


Fig. 12. Speed of the feature tracking framework for STAR+BRIEF, SURF, and SIFT feature types.

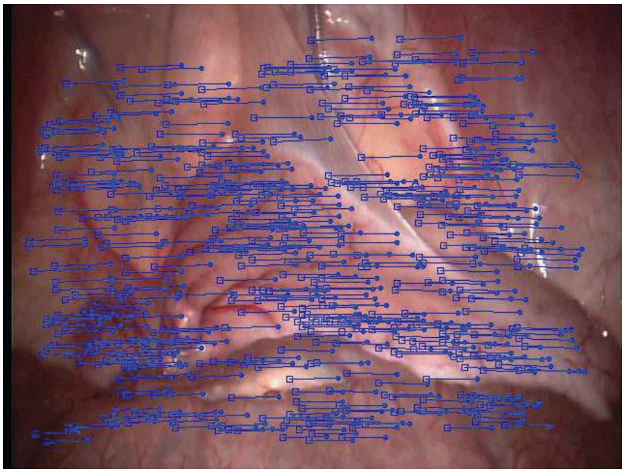


Fig. 13. Sample image of matched left and right channel feature. The image above is the left channel, populated by features (●) found in the current frame. They are connected to the locations on which the matching right features (□) would be located.

As expected, using STAR+BRIEF features is far more efficient in speed than SURF and SIFT, achieving nearly real-time frame rates (15–20 Hz in most scenes, Fig. 12). There is minimal difference in tracking between scene rotation and translation, which is likely due to the fact that inter-frame movement is small. Scenes with higher dynamic motion (i.e., heartbeat) show a reduction in speed; we hypothesize that because of the dynamic environment, poor textures and highly specular reflections, a larger number of features are kept in the feature list, resulting in a linearly increasing number of feature match evaluations required.

B. 3-D Depth Estimation Results

Fig. 13 shows an example of the matching that occurs between left and right camera frames. The features within the left channel are matched to the features in the right channel (using the STAR+BRIEF method), and both locations as well as an interconnecting line are overlayed on top of the left channel frame. This image is a typical representation of the feature density and of the disparities between the left and right channel features.

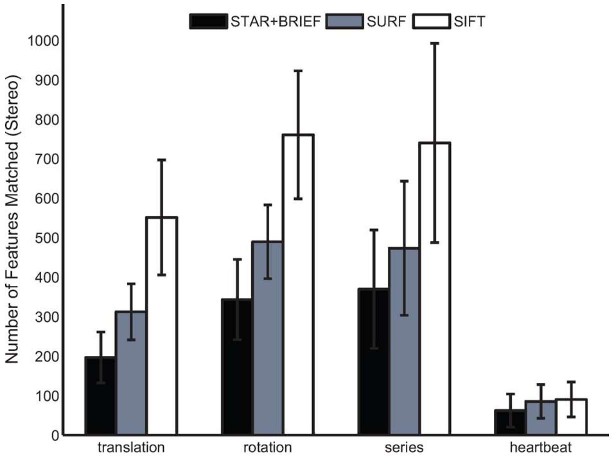


Fig. 14. Number of features matched between the stereo camera channels.

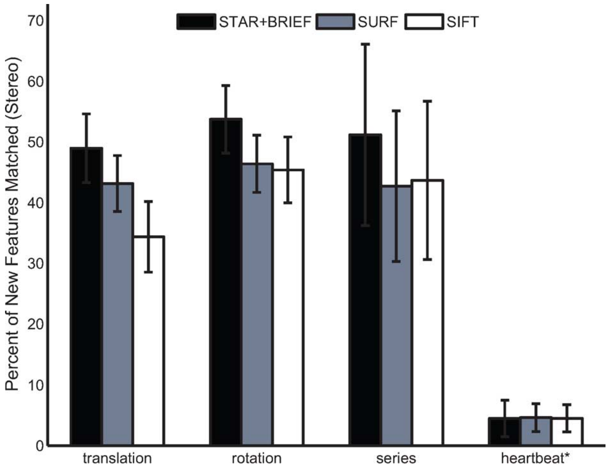


Fig. 15. Percent of features matched between the stereo camera channels.

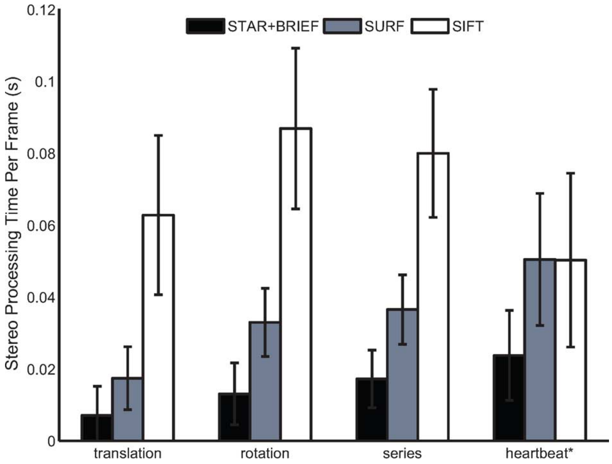
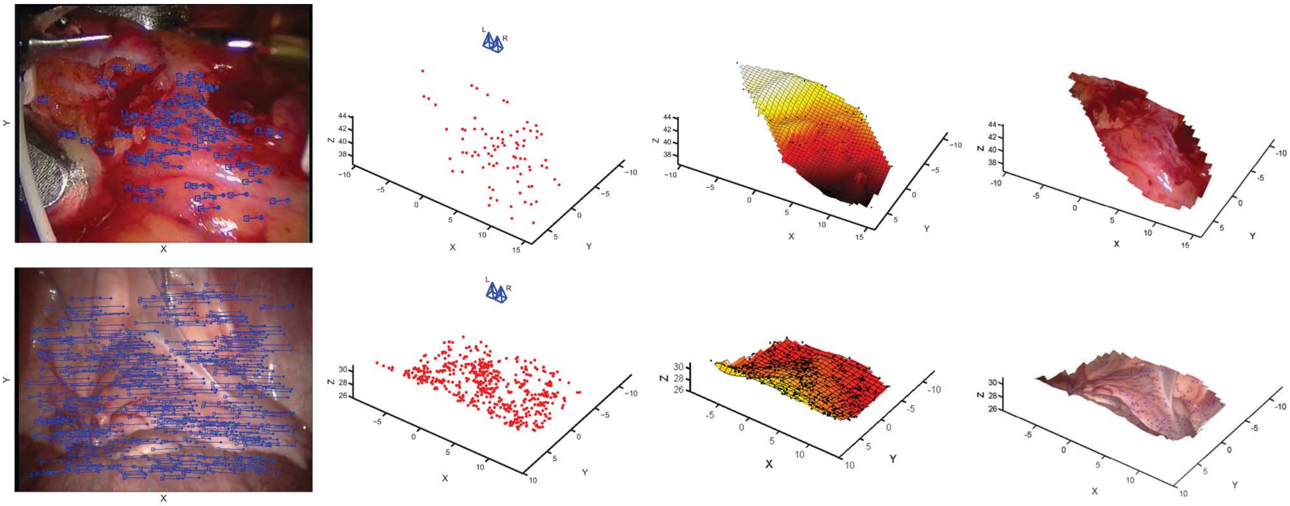


Fig. 16. Time required to process a pair of stereo frames for feature matching.

During stereoscopic feature matching, STAR+BRIEF is seen to match the least number of features as compared to SURF, and SIFT is seen to match many more features than the other feature tracking algorithms (Fig. 14). This is to be expected, since STAR+BRIEF and SURF both find fewer numbers of features



(a) (b) (c) (d)

Fig. 17. Depth estimation of *in vivo* stereo video. (a) Stereo matches in the left frame. (b) Stereo-triangulated points and estimated camera position. (c) Interpolated depth map. (d) Reprojected image.

within each frame than SIFT. We see that STAR+BRIEF features exhibit the highest percentage of matching (approximately 50%), followed by SURF and SIFT (Fig. 15).

One interesting effect that is immediately noticeable is that the video of the porcine heartbeat has significantly fewer features that are stereo-matched from frame to frame. This can be attributed to the fact that during systole there are high frequencies contributing to the motion (100 Hz and above) that cannot be accurately captured with a 30 f/s camera. Motion blur results in less saliency among features. When the heart is in diastole, we are able to see a great deal more stereo-matched features.

Fig. 16 shows the time required to perform feature matching between left and right channel features. Due to the heavy filtering that is applied prior to performing descriptor vector matching, the processing time for each is significantly shorter than that of temporal tracking. We can see that stereo-matching for STAR+BRIEF is extremely fast and can be added to temporal tracking with little added cost to performance.

We provide two visualizations of the 3-D feature depth maps in Fig. 17, one of the beating heart and one of the insuflated abdomen. We first take the feature pairs matched in Fig. 17(a), and we perform stereo-triangulation in order to acquire a 3-D cloud of points (Fig. 17(b), camera locations are shown in the figure). We present a cubic interpolation over these points in Fig. 17(c) to create a contiguous depth map, and we reproject the points among the depth map to visualize their depth consistency. We can see that the 3-D point locations follow closely the smoothed contiguous depth map. Finally, we reproject the image onto the depth map in Fig. 17(d), showing that the extracted contours are congruent with the perceived contours of the tissue.

C. Region Tracking and Registration Results

Fig. 18 shows fiducials registered and tracked over time for the kidney, the heart, and the liver. Tracking from the first frame until the final frame, the Euclidean Error between the estimated true final fiducial position and the fiducial position from registration is 3.31 mm (heart), 2.02 (kidney), and 1.27 mm (liver). We provide the frame-by-frame tracking of the fiducials in Fig. 18 to show the tracking progression over time.

D. In Vivo Human Partial Nephrectomy Results

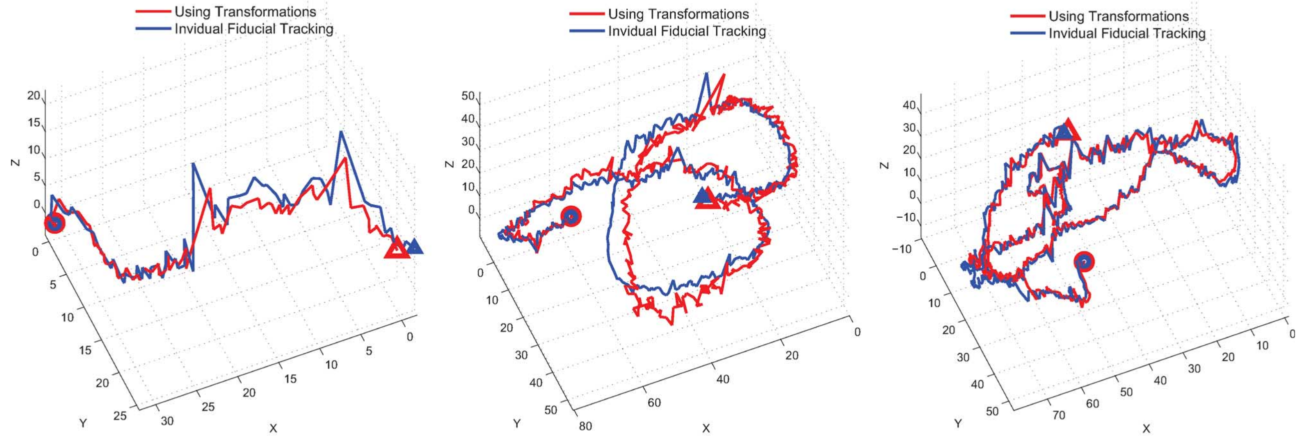
We applied the STAR+BRIEF framework to laparoscopic videos of human partial nephrectomy studies. Table II shows a summary of the performance of the algorithm on different portions of partial nephrectomy. For all tests, the videos tracked approximately 500–800 features in every frame in order to provide a dense description of surface movement and deformation. This can be seen in Fig. 19, which gives a depiction of the persistent features that are tracked during regular motion, induced tissue deformation, surface cauterization, and postdissection manipulation of the kidney. Cauterization is shown to reduce the number of features. We observed that this only occurs during continuous periods cauterization [as shown in the Fig. 19(c)], where there was enough smoke to fill a substantial portion of the endoscopic scene. Thin bands of cauterization smoke wafting through the image (similar to blowing out a match) caused primarily small features to be obscured and to lose tracking. However, these features were found and tracked again after the smoke clears from the image (approximately 2–3 s after cauterization halts).

Temporal matching between frames was again found to be approximately 90% for all video instances, and on average, 1300–2000 features were saved in order to be matched in subsequent frames (Table II).

V. Discussion

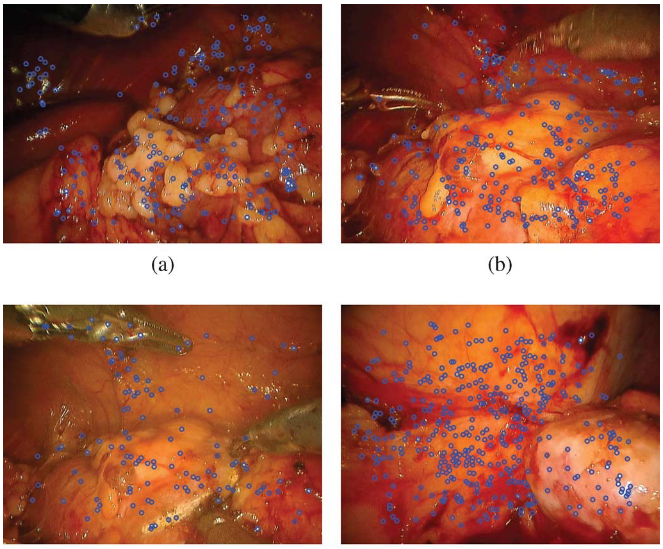
In this paper, we have aimed to address significant clinical needs for tissue tracking and medical image registration.

Our experiments have shown that it is possible to provide real-time tracking of *in vivo* tissue surfaces using efficient salient feature detectors and descriptors. We were able to attain densely-populated feature maps that can be accurately stereo-matched and triangulated into 3-D space.



(a) (b) (c)

Fig. 18. Registration and tracking combination. Circles represent starting location of a surface fiducial, and triangles represents the final position of the surface fiducial after tracking through the videos. Tests were performed *in vitro* on (a) kidney, (b) heart, and (c) liver.



(a) (b) (c) (d)

Fig. 19. Features tracked for over 80% of the frames during human partial nephrectomy. These images are samples of (a) regular motion, (b) deformation, (c) cauterization, and (d) postdissection appearance and motion.

In order to track tissue for long periods of time, we presented a novel history-preserving framework for tracking previously-matched features over time. It allows for features to be tracked individually over consecutive frames, and also allows them to enter and exit the scene for periods of time.

Finally, we extended our tissue tracking framework for long-term tracking of a region of tissue. To the best of our knowledge, we have provided the first evaluation of error in maintaining a registration using tissue tracking, through measuring the fiducial registration errors of tracked tissue under controlled rigid motion. We have shown that registrations can be maintained with an average error of 1.3–3.3 mm.

The proposed tracking framework does not identify or handle the effects of instrument occlusions and shading, and therefore a next research step would be to investigate possible methods to track the instruments and mask their effects from the salient feature trackers. Potential approaches include color filtering and other techniques such as straight edge detection. Specular reflection is often a significant contributing factor to poor tissue tracking, and methods such as those described in [34], [19], and [40] could be used to reduce the effect of specular reflection. Investigation of other salient features and filtering techniques may improve tracking robustness, such as the affine-invariant anisotropic regions detector and extended kalman filtering methods presented in [10]. Varying the threshold limits (see Appendix B) can be used to increase or decrease the number of features and robustness of tracking as needed for different video sequences. Statistical analysis of feature tracking for intraoperative video data (including effects of cauterization, instrument occlusion and instrument–tissue interaction) will a focus of future work.

Implementing multi-threading and GPU acceleration for selected subprocesses (such as dividing each level of the pyramidal scale space to a separate thread for feature extraction) has been shown to be effective in significantly improving the speed of feature detection [9], [45], [44], [56], allowing for more computationally intensive feature detectors to be used in real-time. In the implementations described by Sinha [45] and Wu [56], a number of feature extraction and feature tracking computations were left for CPU processing as they were found to be more efficient. Nonetheless, the trade off between CPU processing and GPU acceleration for feature list management and feature matching is nontrivial. We plan to investigate that in future work.

Our evaluation of maintaining a medical image registration based on object-tracking techniques developed in this paper has been shown as a proof of concept rather than a definitive method for maintaining a registration. Nonetheless, we believe that by using camera-based registration methods such as ones described in [23] or [58], we may be able to maintain a registration with no extra equipment and little effort to the surgeon. Maintaining image registration accuracies within 3 mm may be accurate enough for coarse tumor localization in certain applications.

TABLE II
PERFORMANCE OF THE STAR+BRIEF FEATURE ALGORITHM ON *IN VIVO* HUMAN PATIENT STUDIES DURING PARTIAL NEPHRECTOMY

	Regular Motion	Cauterization	Deformation	Post dissection
Number of Features	549 ± 36	730 ± 64	721 ± 53	489 ± 86
Number in List	1279 ± 110	1952 ± 221	1905 ± 202	1330 ± 228
Percent of New Features Matched	93.28 ± 2.57	89.77 ± 3.71	89.66 ± 3.89	88.43 ± 5.11
Percent of List Features Deleted	2.85 ± 0.83	3.74 ± 0.85	3.82 ± 0.98	4.10 ± 1.27
Frames Per Second	14.03 ± 1.03	9.08 ± 0.88	9.40 ± 0.91	13.87 ± 2.09

For partial nephrectomy (where the kidney is mobilized and clamped so that it does not experience significant motion effects) and for radical prostatectomy (where a mobilized prostate experiences negligible motion effect from patient breathing and heartbeat), a rigid registration would likely suffice.

Even without medical image registration, object tracking can still be very useful during a surgical procedure. As examples, the use of object tracking can provide persistent cues of tumor boundaries throughout the operation, even as they move in and out of the endoscopic view. Region tracking techniques presented in this paper can provide surgeons the ability to create and track boundaries of suspect tissue regions throughout a biopsy, and automatically find and revisit them with relative ease. Beating heart operations could benefit from tissue region tracking in that tracking of the heart tissue motions can be a means for enabling motion compensated surgical instruments. Another open area for research is in full scene reconstructions, which can be used for camera-based 3-D organ modelling or for the purpose of offline virtual endo-cavity exploration.

VI. CONCLUSION

In summary, we have presented an overall framework for tracking tissue in 3-D within an *in vivo* surgical environment and maintaining a medical image registration. We provide three key contributions within this paper.

1) An extensive evaluation of popular salient features algorithms (SIFT, SURF, and a combination of STAR and BRIEF) for acquiring dense, stable, accurate and fast tracking of *in vivo* tissue. The STAR+BRIEF algorithm tested in the paper can reach real-time tracking speeds while still maintaining high feature density and tracking accuracy.
2) A novel framework for history-preserving feature management that enables features to be tracked for long periods of time, without the use of camera pose estimation. Features can be recognized and continuously tracked despite having been lost or having left the scene momentarily. 3-D deformations of the tissue can be fully captured and accurately tracked since each feature is individually tracked in space.
3) A novel method for maintaining an image registration using our history-preserving framework. Feature tracking is used to maintain the registrations over time while avoiding drift. We present the first accuracy measurements for maintaining a registration for a variety of tissue types.

Future work will investigate techniques involving GPU acceleration for improved speed, methods for handling specular reflection and instrument occlusion, and nonrigid registration methods for maintaining higher registration accuracies over time.

APPENDIX

A. Filtering Possible Feature Matches

This section describes the methods used during feature matching that reduce the number of unnecessary descriptor distance evaluations and reduces false positives in matches. These methods are generalized for all temporal, stereo, and object feature matches.

Given the ith feature from list 1, $f_i(x_i, y_i, k_i, \theta_i)$, and the jth feature from list 2, $f_j(x_j, y_j, k_j, \theta_j)$, where x_i, y_i, x_j, and y_j are their pixel location in the original image, k_i and k_j are their characteristic scales, and θ_i and θ_j represents their orientation, we will only include within the set of matching possibilities those pairs of features that satisfy the criteria listed below.

- *Scales:* The characteristic scales of features are compared, where the maximum ratio κ of scales is

$$\left|\log\left(\frac{k_i}{k_j}\right)\right| < \kappa. \tag{1}$$

- *Orientations:* The characteristic orientation of SURF and SIFT features are compared, where the maximum allowable orientation difference Θ is

$$|\theta_i - \theta_j| < \Theta. \tag{2}$$

- *Proximity:* We do not expect to see large motions from consecutive frames; therefore, we can limit our search to a width δ_x and height δ_y centered about a feature's last known location

$$\begin{aligned} |x_i - x_j| &< \delta_x \\ |y_i - y_j| &< \delta_y. \end{aligned} \tag{3}$$

- *Descriptor Distance Ratio:* Given a subset of possible matching features, we let d_{first} represent the descriptors' distance for the best match and d_{second} represent the descriptors' distance for the second best match. Given a confidence threshold λ, a match is only considered valid if

$$\frac{d_{\text{second}}}{d_{\text{first}}} < \lambda. \tag{4}$$

- *Neighborhood Consistency:* One of the characteristics of *in vivo* surgical video is that the soft-tissue within the scene deforms with the adjacent tissues and depth discontinuities within the image are sparse. Therefore the spatial movement of a feature will be similar to those of neighboring features.

A feature's neighbors are defined as features that are within a distance of 20% of the image width. Given a set of two neighboring feature locations in frame n, $\{x_{1,n}, y_{1,n}\}$ and $\{x_{2,n}, y_{2,n}\}$, and their matched locations in the previous frame, $\{x_{1,n-1}, y_{1,n-1}\}$ and $\{x_{2,n-1}, y_{2,n-1}\}$, we consider their movements to be significantly different if

$$\left|\log\left(\frac{\delta x_1^2 + \delta y_1^2}{\delta x_2^2 + \delta y_2^2}\right)\right| > \gamma \tag{5}$$

where $\{\delta x_1, \delta y_1\} = \{x_{1,n} - x_{1,n-1}, y_{1,n} - y_{1,n-1}\}$ and $\{\delta x_2, \delta y_2\} = \{x_{2,n} - x_{2,n-1}, y_{2,n} - y_{2,n-1}\}$, where γ represents the maximum allowable ratio of squared distances of movements between the neighboring features.

Furthermore, we consider the dot product of their movement vectors, and evaluate the direction of movement of neighboring features to be significantly different if

$$\Delta\theta = \operatorname{acos}\left(\frac{\delta x_1 \cdot \delta x_2 + \delta y_1 \cdot \delta y_2}{\sqrt{\delta x_1^2 + \delta y_1^2}\sqrt{\delta x_2^2 + \delta y_2^2}}\right) > \epsilon \tag{6}$$

where ϵ is the maximum allowable difference in the direction of movement.

In practice, we only check against ϵ and γ if there is a temporal displacement of five pixels for each matched feature, as smaller displacements results in coarser distances and orientation values, and therefore would reduce the efficacy of the methods.

B. Selection of Parameter Values

This section describes the parameter values for the tissue tracking experiments that were used.

$\kappa = 2.0$ matches features that are at most one octave apart over a single timestep; feature scales are not expected to change significantly between the left and right stereo channels, and therefore $\kappa = \log(\sqrt{2})$. Feature orientations are not expected to change more than $\Theta = \pi/18$ radians during a single timestep; between the left and right cameras, $\Theta = \pi/12$ to account for slightly different viewpoints. $(\delta_x, \delta_y = 0.2 * \text{image\_width})$ in order to capture large tissue motions during temporal tracking. The epipolar constraint in stereoscopic matching limits the feature matches to within a narrow band $(\delta_x = 0.5 * \text{image\_width}, \delta_y = 0.05 * \text{image\_height})$. A wide search of object features $(\delta_x, \delta_y = 0.5 * \text{image\_width})$ accounts for situations when tracked region leaves the image scene. $\lambda = 0.5$ is chosen such that the best feature match has half the error of the second best match. $\gamma = 2 * \log(1.5)$ and $\epsilon = \pi/18$ were chosen to restrict the movement of features bundles to represent smooth and consistent motion, which should be characteristic of *in vivo* tissue.

Acknowledgment

The authors would like to thank Prof. G.-Z. Yang and the Hamlyn Centre for Robotic Surgery at Imperial College of London for open access to their *in vivo* data sets.

References

[1] M. Agrawal, K. Konolige, and M. Blas, "Censure: Center surround extremas for realtime feature detection and matching," in *Computer Vision ECCV 2008*. Berlin: Springer, 2008, vol. 5305, Lecture Notes in Computer Science, pp. 102–115.

[2] W. Bachta, P. Renaud, E. Malis, K. Hashimoto, and J. Gangloff, "Visual servoing for beating heart surgery," in *Visual Servoing via Advanced Numerical Methods*, G. Chesi and K. Hashimoto, Eds. Berlin, Germany: Springer, 2010, vol. 401, Lecture Notes in Computer Science, pp. 91–114.

[3] H. Bay, T. Tuytelaars, and L. Van Gool, "Surf: Speeded up robust features," in *Computer Vision—ECCV 2006*. Berlin, Germany: Springer, 2006, vol. 3951, Lecture Notes in Computer Science, pp. 404–417.

[4] J. Bouguet, Camera Calibration Toolbox for Matlab 2010.

[5] D. Burschka, M. Li, M. Ishii, R. H. Taylor, and G. D. Hager, "Scale-invariant registration of monocular endoscopic images to ct-scans for sinus surgery," in *Medical Image Computing and Computer Assisted Intervention (MICCAI)*, 2004, pp. 413–421.

[6] M. Calonder, V. Lepetit, C. Strecha, and P. Fua, "BRIEF: Binary robust independent elementary features," in *Computer Vision—ECCV 2010*. Berlin, Germany: Springer, 2010, vol. 6314, Lecture Notes in Computer Science, pp. 778–792.

[7] E. Delponte, F. Isgr, F. Odone, and A. Verri, "Svd-matching using sift features," *Graph. Models*, vol. 68, no. 5–6, pp. 415–431, 2006.

[8] M. A. Fischler and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Commun. ACM*, vol. 24, pp. 381–395, Jun. 1981.

[9] J. Fung and S. Mann, "Openvidia: Parallel GPU computer vision," in *ACM Int/ Conf. Multimedia—MULTIMEDIA 2005*, 2005, pp. 849–852.

[10] S. Giannarou, M. V. Scarzanella, and G.-Z. Yang, "Probabilistic tracking of afne-invariant anisotropic regions," *IEEE Trans. Pattern Anal. Mach. Intell.*, to be published.

[11] R. Ginhoux, J. Gangloff, M. de Mathelin, L. Soler, M. M. A. Sanchez, and J. Marescaux, "Beating heart tracking in robotic surgery using 500 Hz visual servoing, model predictive control and an adaptive observer," in *IEEE Int. Conf. Robot. Automat.*, 2004, pp. 274–279.

[12] O. G. Grasa, J. Civera, and J. M. M. Montiel, "Ekf monocular slam with relocalization for laparoscopic sequences," in *Int. Conf. Robot. Automat. (ICRA2011)*, 2011, pp. 4816–4821.

[13] M. Groeger, T. Ortmaier, W. Sepp, and G. Hirzinger, "Tracking local motion on the beating heart," in *SPIE Med. Imag. 2002: Visualizat., Image-Guided Procedures, Display*, 2002, vol. 4681, pp. 233–241.

[14] B. K. P. Horn, "Closed-form solution of absolute orientation using unit quaternions," *J. Opt. Soc. Am. A*, vol. 4, no. 4, pp. 629–642, Apr. 1987.

[15] Z. Kalal, K. Mikolajczyk, and J. Matas, "Forward-backward error: Automatic detection of tracking failures," in *Proc. IEEE 20th Int. Conf. Pattern Recognit.*, 2010, pp. 2756–2759.

[16] Imperial College London, Laparoscopic Video Data Set [Online]. Available: http://hamlyn.doc.ic.ac.uk/vision url:

[17] Univ. British Columbia, Laparoscopic video data set [Online]. Available: http://rcl.ece.ubc.ca/content/laparoscopic-videos

[18] W. Lau, N. Ramey, J. Corso, N. Thakor, and G. Hager, "Stereo-based endoscopic tracking of cardiac surface deformation," in *Medical Image Computing and Computer-Assisted Intervention MICCAI 2004*. Berlin/Heidelberg: Springer, 2004, vol. 3217, Lecture Notes in Computer, pp. 494–501.

[19] B. Lo, A. Chung, D. Stoyanov, G. Mylonas, and G.-Z. Yang, "Real-time intra-operative 3-D tissue deformation recovery," in *IEEE Int. Symp. Biomed. Imag.*, 2008, pp. 1387–1390.

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

[21] K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 10, pp. 1615–1630, Oct. 2005.

[22] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. Gool, "A comparison of affine region detectors," *Int. J. Comput. Vis.*, vol. 65, pp. 43–72, 2005.

[23] D. Mirota, R. H. Taylor, M. Ishii, and G. D. Hager, "Direct endoscopic video registration for sinus surgery," in *Proc. SPIE*, 2009, vol. 7261, pp. 91–99.

[24] M. Moll, H.-W. Tang, and L. Van Gool, "Gpu-accelerated robotic intra-operative laparoscopic 3-D reconstruction," in *Information Processing in Computer-Assisted Interventions*. Berlin, Germany: Springer, 2010, vol. 6135, Lecture Notes in Computer Science, pp. 91–101.

[25] K. Mori, D. Deguchi, K. Akiyama, T. Kitasaka, C. Maurer, Y. Suenaga, H. Takabatake, M. Mori, and H. Natori, "Hybrid bronchoscope tracking using a magnetic tracking sensor and image registration," in *Medical Image Computing and Computer-Assisted Intervention—MICCAI 2005*. Berlin, Germany: Springer, 2005, vol. 3750, Lecture Notes in Computer Science, pp. 543–550.

[26] P. Mountney, B. Lo, S. Thiemjarus, D. Stoyanov, and G. Zhong-Yang, "A probabilistic framework for tracking deformable soft tissue in minimally invasive surgery," in *Medical Image Computing and Computer-Assisted Intervention - MICCAI 2007*. Berlin, Germany: Springer, 2007, vol. 4792, Lecture Notes in Computer Science, pp. 34–41.

[27] P. Mountney, D. Stoyanov, A. Davison, and G.-Z. Yang, "Simultaneous stereoscope localization and soft-tissue mapping for minimal invasive surgery," in *Medical Image Computing and Computer-Assisted Intervention—MICCAI 2006*, R. Larsen, M. Nielsen, and J. Sporring, Eds. Berlin, Germany: Springer, 2006, vol. 4190, Lecture Notes in Computer Science, pp. 347–354.

[28] P. Mountney and G.-Z. Yang, "Soft tissue tracking for minimally invasive surgery: Learning local deformation online," in *Medical Image Computing and Computer-Assisted Intervention MICCAI 2008*. Berlin, Germany: Springer, 2008, vol. 5242, Lecture Notes in Computer Science, pp. 364–372.

[29] P. Mountney and G.-Z. Yang, "Dynamic view expansion for minimally invasive surgery using simultaneous localization and mapping," in *IEEE Eng. Med. Biol. Soc.*, 2009, pp. 1184–1187.

[30] Y. Nakamura, K. Kishi, and H. Kawakami, "Heartbeat synchronization for robotic cardiac surgery," in *IEEE Int. Conf. Robot. Automat.*, 2001, vol. 2, pp. 2014–2019.

[31] A. Noce, J. Triboulet, and P. Poignet, "Efficient tracking of the heart using texture," in *IEEE Eng. Med. Biol. Soc.*, 2007, pp. 4480–4483.

[32] N. Oda, J. Hasegawa, T. Nonami, M. Yamaguchi, and N. Ohyama, "Estimation of the surface topography from monocular endoscopic images," *Opt. Commun.*, vol. 109, no. 3–4, pp. 215–221, 1994.

[33] T. Ortmaier, M. Groger, D. Boehm, V. Falk, and G. Hirzinger, "Motion estimation in beating heart surgery," *IEEE Trans. Biomed. Eng.*, vol. 52, no. 10, pp. 1729–1740, Oct. 2005.

[34] C. Q. Forster and C. Tozzi, "Towards 3-D reconstruction of endoscope images using shape from shading," *Comput. Graphics Image Process.*, pp. 90–96, 2000.

[35] R. Richa, A. B. , and P. Poignet, "Robust 3-D visual tracking for robotic-assisted cardiac interventions," in *Medical Image Computing and Computer-Assisted Intervention MICCAI 2010*. Berlin: Springer, 2010, vol. 6361, Lecture Notes in Computer Science, pp. 267–274.

[36] R. Richa, A. P. B. , and P. Poignet, "Motion prediction for tracking the beating heart," in *IEEE Eng. Med. Biol. Soc.*, Aug. 2008, pp. 3261–3264.

[37] R. Richa, A. P. B. , and P. Poignet, "Towards robust 3-D visual tracking for motion compensation in beating heart surgery," *Med. Image Anal.*, vol. 15, no. 3, pp. 302–315, 2011.

[38] R. Richa, P. Poignet, and C. Liu, "Deformable motion tracking of the heart surface," in *IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Sep. 2008, pp. 3997–4003.

[39] R. Richa, P. Poignet, and C. Liu, "Efficient 3-D tracking for motion compensation in beating heart surgery," in *Medical Image Computing and Computer-Assisted Intervention—MICCAI 2008*. Berlin, Germany: Springer, 2008, vol. 5242, Lecture Notes in Computer Science, pp. 684–691.

[40] R. Richa, P. Poignet, and C. Liu, "Three-dimensional motion tracking for beating heart surgery using a thin-plate spline deformable model," *Int. J. Robot. Res.*, vol. 29, pp. 218–230, Feb. 2010.

[41] E. Rosten and T. Drummond, "Machine learning for high-speed corner detection," in *Computer Vision ECCV 2006*. Berlin, Germany: Springer, 2006, vol. 3951, Lecture Notes in Computer Science, pp. 430–443.

[42] M. Sauvee, A. Noce, P. Poignet, J. Triboulet, and E. Dombre, "Three-dimensional heart motion estimation using endoscopic monocular vision system: From artificial landmarks to texture analysis," *Biomed. Signal Process. Control*, vol. 2, no. 3, pp. 199–207, 2007.

[43] J. Shi and C. Tomasi, "Good features to track," in *Comput. Vis. Pattern Recognit.*, Jun. 1994, pp. 593–600.

[44] S. Sinha, J.-M. Frahm, M. Pollefeys, and Y. Genc, "Feature tracking and matching in video using programmable graphics hardware," *Mach. Vis. Appl.*, vol. 22, pp. 207–217, 2011.

[45] S. N. Sinha, J.-M. Frahm, M. Pollefeys, and Y. Genc, Gpu-based video feature tracking and matching Tech. Rep., 2006.

[46] D. Stoyanov, A. Darzi, and G. Z. Yang, "Dense 3-D depth recovery for soft tissue deformation during robotic assisted laparoscopic surgery," in *Medical Image Computing and Computer-Assisted Intervention—MICCAI 2004*. Berlin, Germany: Springer, 2004, vol. 3217, pp. 41–48.

[47] D. Stoyanov, G. Mylonas, F. Deligianni, A. Darzi, and G. Yang, "Soft-tissue motion tracking and structure estimation for robotic assisted MIS procedures," in *Medical Image Computing and Computer-Assisted Intervention—MICCAI 2005*. Berlin, Germany: Springer, 2005, vol. 3750, Lecture Notes in Computer Science, pp. 139–146.

[48] D. Teber, S. Guven, T. Simpfendorfer, M. Baumhauer, E. O. Guven, F. Yencilek, A. S. Gozen, and J. Rassweiler, "Augmented reality: A new tool to improve surgical accuracy during laparoscopic partial nephrectomy? Preliminary in vitro and in vivo results," *Eur. Urol.*, vol. 56, no. 2, pp. 332–338, 2009.

[49] R. U. Thoranaghatte, G. Zheng, F. Langlotz, and L. P. Nolte, "Endoscope based hybrid-navigation system for minimally invasive ventral-spine surgeries," *Comput. Aided Surg.*, pp. 351–356, 2005.

[50] T. Tuytelaars and K. Mikolajczyk, "Local invariant feature detectors: A survey," *Foundat. Trends Comput. Graphics Vis.*, vol. 3, pp. 177–-280, Jul. 2008.

[51] O. Ukimura and I. S. Gill*, Augmented Reality for Computer-Assisted Image-Guided Minimally Invasive Urology*. Berlin, Germany: Springer-Verlag, 2009, pp. 179–184.

[52] M. Visentini-Scarzanella, G. Mylonas, D. Stoyanov, and G.-Z. Yang, "i-Brush: A gaze-contingent virtual paintbrush for dense 3-D reconstruction in robotic assisted surgery," in *Medical Image Computing and Computer-Assisted Intervention—MICCAI 2009*. Berlin, Germany: Springer, 2009, vol. 5761, Lecture Notes in Computer Science, pp. 353–360.

[53] H. Wang, D. Mirota, M. Ishii, and G. Hager, "Robust motion estimation and structure recovery from endoscopic image sequences with an adaptive scale kernel consensus estimator," in *IEEE Conf. Comput. Vis. Pattern Recognit.*, 2008, pp. 1–7.

[54] C. Wengert, P. Cattin, J. Duff, C. Baur, and G. Szkely, "Markerless endoscopic registration and referencing," in *Medical Image Computing and Computer-Assisted Intervention MICCAI 2006*. Berlin, Germany: Springer, 2006, vol. 4190, Lecture Notes in Computer Science, pp. 816–823.

[55] Willow Garage, STAR detector [Online]. Available: http://pr.willowgarage.com/wiki/star_detector

[56] C. Wu, SiftGPU: A GPU Implementation of Scale Invariant Feature Transform (SIFT) 2007 [Online]. Available: http://cs.unc.edu/ccwu/siftgpu

[57] A. Yilmaz, O. Javed, and M. Shah, "Object tracking: A survey," *ACM Comput. Sur.*, vol. 38, Dec. 2006.

[58] M. Yip, T. Adebar, R. Rohling, S. Salcudean, and C. Nguan, "3-D ultrasound to stereoscopic camera registration through an air-tissue boundary," in *Medical Image Computing and Computer-Assisted Intervention—MICCAI 2010*. Berlin, Germany: Springer, 2010, vol. 6362, Lecture Notes in Computer Science, pp. 626–634.

[59] M. Yip, D. Lowe, S. Salcudean, R. Rohling, and C. Nguan, "Real-time methods for long-term tissue feature tracking in endoscopic scenes," in *Information Processing in Computer-Assisted Interventions—IPCAI 2012*. Berlin, Germany: Springer, 2012, vol. 7330, pp. 33–43.