Realtime Wide-baseline Registration of the Uterus in Monocular Laparoscopic Videos

Toby Collins, Daniel Pizarro, Adrien Bartoli, Michel Canis and Nicolas Bourdel
ISIT UMR 6284 CNRS/UdA, Clermont-Ferrand, France

Abstract. We present a way to register the uterus in monocular laparoscopy in realtime using a novel two-phase approach. This differs significantly to SLAM, which is currently the main approach for registration in MIS when scenes are approximately rigid. In the first phase we construct a 3D model of the uterus using dense SfM. This involves a method for semi-automatically masking the uterus from background structures in a set of reference frames, which we call Mask Bootstrapping from Motion (MBM). In the second phase the 3D model is registered to the live laparoscopic video using a novel wide-baseline approach that uses many texturemaps to capture the real changes in appearance of the uterus. Capturing these changes means that registration can be performed reliably without needing temporal priors, which are needed in SLAM. This simplifies registration and leads to far fewer tuning parameters. We show that our approach significantly outperforms SLAM on an *in vivo* dataset comprising three human uteri.

1 Introduction

One of the main current goals of computer assisted intervention in Minimal Invasive Surgery (MIS) is to enrich the surgeon's video data using Augmented Reality (AR). Examples of this include being able to visualise sub-surface structures [16], enlarge the surgical field of view [18] and overlay information from other imaging modalities [14]. AR in MIS involves solving a fundamental open problem, namely registration. Depending on the application this may involve registering optical images to one another, or to register them to another modality. A challenging problem is how to achieve registration accurately, reliably and in realtime. In this paper we focus on the problem of registering laparoscopic images of the uterus. Solving this problem would open up several important clinical applications, including AR-assisted resection of lesions such as uterine fibroids and endometriosis.

The uterus is a flexible organ that can exhibit strong deformation when manipulated with laparoscopic tools [12]. However when observing the uterus during intervention prior to resection it remains quite rigid and does not deform significantly due to respiration. Optical registration in laparoscopy has been studied previously for other organs using the assumption of rigid, or approximately rigid motion. This has been developed with monocular [4, 6, 7] and stereo [13, 18] laparoscopes. These solve the problem using a general paradigm called visual Simultaneous Localisation and Mapping (SLAM). Visual SLAM

relies only on raw optical data, and does not need other hardware such as magnetic [14] or optical [16] tracking devices. SLAM involves building a 3D representation of the environment, known as the *map*, and determining the rigid transform which positions the map in the camera's coordinate frame. The core challenge in SLAM is how to achieve *data association*. SLAM requires data association in two respects. The first is for *map building*. The second is for *localisation*, which is to determine where the map's points are located in a new input image. SLAM offers a fast solution to these problems and has found considerable success in man-made environments. However SLAM in MIS is still proving challenging. This is due to the repeated nature of tissue texture, rapid camera motion and photo-constancy violations caused by blood or mucous.

SLAM may also have difficulty when the scene is not globally rigid. When the scene is made up of independently moving structures SLAM can make errors by merging features from different structures into one map. This can occur if there are sections of video where the image motion is induced only by movement of the camera, at which point SLAM will merge features from the entire scene. This may then lead to localisation errors in later frames. For laparoscopic procedures involving the uterus a typical scene will comprise the uterus, ovaries, peritoneum, small intestine and bladder. In most procedures a cannula is inserted into the uterus through the vagina and is operated externally by an assistant. The assistant's hand movement may cause the uterus to move independently of the surrounding structures. As we will show, one cannot apply off-the-shelf monocular SLAM in these conditions because we have a registration *and* segmentation problem to solve. This amounts to computing binary *masks* which label pixels as either being on the uterus body or not. However achieving this automatically is difficult and has not been studied in the literature.

The focus of this work is to solve registration using a minimal amount of manual segmentation. A naive way to proceed would be to mask the uterus manually in one or more frames and enforce that SLAM uses features found only within the masks. However, there is no guarantee that SLAM will not eventually use features from surrounding organs, thus leading to mapping and localisation errors. By contrast it is infeasible to mask frames manually for every frame.

Proposed Approach and Registration Pipeline. Our solution is to step away from the SLAM paradigm and solve the mapping problem with dense multi-view Structure-from-Motion (SfM) [8]. We use SfM to *explicitly decouple* the map building process from localisation. Using SfM has the advantage that data association is done without requiring input images come from a video. Rather, it works using a collection of unorganised images, and unlike SLAM assumes nothing about temporal continuity. We propose a SfM-based method for registering the uterus in two distinct phases. We illustrate this in Figure 1. Phase 1 involves estimating a dense 3D model of the uterus from a set of *reference frames*. These are recorded whilst the surgeon views the uterus from a range of different viewpoints. This involves a novel process that we call *Mask Bootstrapping from Motion* (MBM). The idea behind MBM is to use a small number of manually-segmented masks to bootstrap computing the masks in all

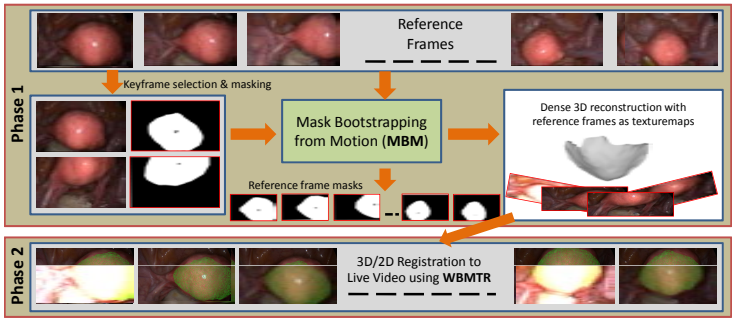

Fig. 1: Proposed approach pipeline divided into two phases. Phase 1 uses a reference video to construct a dense 3D surface model of the uterus. Phase 2 registers the model to new video frames in realtime.

reference frames. First a small number of reference frames are selected, called *keyframes*, which are masked manually. An initial dense 3D uterus model is computed using SfM with only these masked keyframes. The model is then registered to all other reference frames, and their masks are predicted using the model's projected silhouette. We then can use *all* reference frames and masks to compute a more accurate 3D model. Importantly, the masks do not need to be particularly accurate, because modern SfM algorithms are inherently robust. Rather the mask's job is to prevent confusion during SfM by background structures transforming according to different motion models.

Phase 2 involves using the 3D model from Phase 1 to register the uterus in realtime. In contrast to SLAM, we present a way to achieve this that does not rely on a prediction using the registration in previous frames. Rather each frame can be registered independently. This is achievable due to the rich appearance data provided by the model's many reference frames. We call this Wide-Baseline Multi-Texturemap Registration (WBMTR).

Materials. Data has been acquired with a standard Karl Stortz 10mm zero-degree HD laparoscope, capturing videos at 25fps at 1920×1080 pixels. The laparoscope was calibrated using standard methods immediately before intervention using OpenCV's calibration library. Algorithms have been implemented in a combination of C++ and CUDA, and run on a standard Intel i7 desktop PC with an NVidia GTX 660 CUDA-enabled graphics card.

2 Phase 1: Dense 3D Reconstruction using MBM

2.1 Creating the Exploratory Video and Frame Pre-processing

The exploratory video begins at the point during intervention after abdominal inflation, instrument and camera insertion and once the uterus has been localised

by the surgeon. The goal of this video is two-fold. The first is to provide sufficient data so that the uterus body can be reconstructed with SfM. The second is to provide sufficiently different views of the uterus in order to capture how its appearance changes as it is viewed from different viewpoints. This second point is crucial for achieving reliable registration in Phase 2. To achieve these goals we capture the exploratory video in somewhat controlled conditions with a simple protocol. By contrast in Phase 2 the surgeon can view the uterus as they wish.

The protocol is as follows. The uterus is centred in the video so that the uterus fundus is fully visible to the camera (Figure 1, top keyframe). At this point video capture begins. The uterus is then tilted by manipulating the cannula to reveal the posterior side of its body (Figure 1, bottom keyframe). It is then moved in a rotary fashion to reveal lateral and anterior views. Once completed video capture stops. We denote the length in seconds of the exploratory video with T. In practice $T \simeq 30$ seconds. From the capture we select a subset of 60 reference frames. We do this automatically by partitioning the video into 60 even time intervals: $\{t_1, t_2, ...t_{60}\}$. At each time t_k we create a local window comprising the frames at $t_k \pm \frac{T}{60\times 2}$. From this window we select the sharpest frame. We do this by computing the response of a 5×5 smooth Laplacian filter and measuring a robust maximum (specifically at the 90^{th} percentile). The frame with the highest robust maximum in the k^{th} interval is chosen to be the k^{th} reference frame. From the reference frames we select a subset of 8 uniformly-spaced *keyframes*. For each keyframe we create a mask by manually outlining the uterus body with an interactive polygon. This process is quick because the masks do *not* need to be particularly accurate, and takes approximately 1-2 minutes to perform.

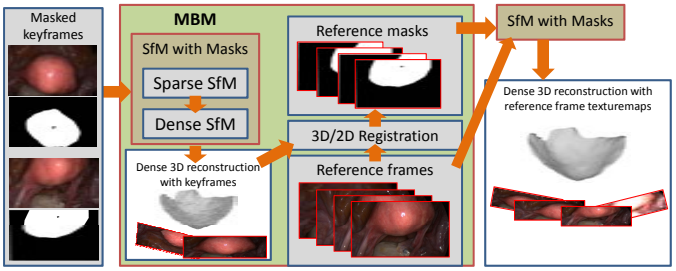

Fig. 2: Mask Bootstrapping from Motion (MBM) applied to the uterus

2.2 Mask Bootstrapping from Motion (MBM)

In Fig. 2 we have expanded out the MBM component in Fig. 1. MBM takes as inputs the set of keyframes and their respective masks. The first step of MBM is to perform dense SfM using the masked keyframes. Modern dense SfM works in two stages. The first stage is to perform sparse SfM using local features extracted from the images. The well established method for this is to estimate the

camera poses from feature correspondences, and then refine them with bundle adjustment [8]. The second stage involves reconstructing a dense surface using multi-view stereo [17]. The masks come into play in both stages. In the first stage features are only used that lie within the masks. In the second stage only pixel information within the masks is used to constrain dense reconstruction. There exist several mature libraries for performing dense SfM. We have found good success for both the sparse and dense stages using Agisoft's Photoscan [1]. For reconstructing the uterus we did not need to change Photoscan's default parameters for mesh smoothness and resolution. With a set of 8 keyframes Sparse SfM takes about 15 seconds on our hardware and Dense SfM takes about 1 minute, returning a 3D model in the order of 20,000 vertices.

The next stage of MBM is to take this 3D model and perform registration using WBMTR for the remaining reference frames. We postpone details of WBMTR to §3, as it is the same algorithm used for live registration. For each reference frame WBMTR either gives us the model's 3D pose, or it returns a failure to register. For all frames with 3D pose estimates, we render the model with OpenGL and compute the model's silhouette. We then morphologically dilate the silhouette to grow its area to allow the next run of SfM to be able to discover more of the uterus surface. Empirically we have found a dilation of about 15% area to be effective. There is a compromise here, as we do not want significant background regions being included in the masks. We then pass the reference images and their masks back to dense SfM, which returns a second 3D surface model, and the 3D poses of the model with respect to the reference frames. Sometimes it may fail to estimate pose. The reasons for this are the same as the reason why WBMTR may fail, chiefly if there is excessive motion blur. We call the set of reference images for which pose *was* estimated the *texturemap images*. We use this term because these images allow us to texturemap the model. However unlike traditional texturemapping where the images are combined to form a *single* aggregated texturemap, we keep *all* texturemap images. By doing so we capture the real changes of appearance of the uterus as it is viewed from different viewpoints. This is important because state-of-the-art feature detectors and descriptors can still have difficulty in handling viewpoint changes due to the complex interaction between tissue reflectance, illumination angle and surface orientation. When we use many texturemap images, we are reducing the requirement for features to be invariant to these changes.

3 Phase 2: Wide-Baseline Multi-Texturemap Registration

In this section we describe WBMTR for registering the 3D model in realtime. WBMTR is a feature-based method. That is, registration is achieved by determining feature correspondences between the 3D model's texturemaps and a given input image. Unlike SLAM, WBMTR requires no initial pose estimate.

3.1 Preparing the Model for Registration

For each texturemap image, we render the 3D model with OpenGL and store the corresponding 3D position of all pixels that lie within the model's silhouette. Using this we can immediately determine the 3D positions of any 2D image features located within the model's silhouette. Note that without computing a dense 3D model this is not possible in general. For each texturemap image we extract a large set of image features. Specifically we use GPU-SURF features [2] because they can be computed very quickly and, as shown in the evaluation section, work well for the uterus. Similar or better accuracy would be expected with SIFT [11], however these are far slower to compute. We use OpenCV's GPU-SURF implementation with default settings, giving descriptors of length $d = 128$ bytes. For a typical 1920×1080 images of the uterus, between 70-500 features are usually found, taking less than 10ms with our hardware. We use the average of the green and blue channels to compute features, rather than the standard approach of using average intensity. The reason is that green and blue light penetrates human tissue superficially and do not exhibit as much sub-surface scattering as with red light. The difference is very prominent with the uterus [3]. To mitigate tracking specularities we detect saturated pixels as those with intensity greater than 250, and any feature that lies within 5 pixels to a saturated pixel is discarded. We concatenate the features from all texturemap images into a single list, represented by $\mathcal{F} = \{(\mathbf{x}_m, I_m, \mathbf{d}_m)\}$, where $\mathbf{x}_m$ denotes the m^{th} feature's 3D position in the model coordinate frame, I_m denotes the index of the texturemap from which it was detected and $\mathbf{d}_m$ denotes its descriptor.

3.2 Registration

For a given input image we compute its GPU-SURF features using the average of its green and blue channels. We denote this with the set $\mathcal{G} = \{(\mathbf{y}_i, \tilde{\mathbf{d}}_i)\}$. $\mathbf{y}_i$ denotes the i^{th} feature's image position and $\tilde{\mathbf{d}}_i$ denotes its descriptor. WBMTR follows a RANSAC-based hypothesis and test framework [5]. Specifically this splits registration into three components. The first involves computing a set of candidate matches between $\mathcal{F}$ and $\mathcal{G}$. The second involves searching for a pose hypothesis that can best explain these matches. The third involves taking the best hypothesis and refining with efficient gradient-based optimisation [10].

Computing candidate matches. Candidate matches are found between $\mathcal{F}$ and $\mathcal{G}$ as those pairs with *(i)* strong descriptor agreement and *(ii)* have a low likelihood of being false. *(ii)* can be achieved with Lowe's Ratio Test (LRT) [11]. For each member of $\mathcal{F}$ we compute the member in $\mathcal{G}$ with the nearest descriptor. If this descriptor distance is less than τ times the distance to the second nearest descriptor in $\mathcal{G}$, it is deemed a candidate match. The LRT is very standard in feature-based pose estimation and we use a default value of $\tau = 0.8$. A novelty of using *multiple* texturemaps is that we can also exploit match *coherence*. What we mean by coherence is that correct matches are likely to be those which come from similar texturemap images. Enforcing coherence can reduce false matches

because it prevents matches occurring from wildly different texturemaps. We enforce coherence with a winner-takes-all strategy. We first find the index I^* of the texturemap with the most amount of candidate matches after applying LRT. This indicates the texturemap image which is 'closest' to the input image. Because SURF is invariant to scale changes and image rotation, close means a texturemap image which views the uterus from a similar viewpoint, up to a change in depth and a rotation of the laparoscope about its optical axis. We then recompute the candidate matches with LRT, but using *only* features from I^*. Performing these processes is very quick. This is because $\mathcal{F}$ is completely pre-computed, and evaluating descriptor distances can be distributed trivially on the GPU.

Computing 3D pose. Given the set of candidate matches, we perform RANSAC to find the most compatible rigid 3D pose. This involves sampling many match subsets of size 4, and for each sample creating a pose hypothesis using PnP [10]. Each hypothesis is tested for support by measuring how many of the other matches are predicted well by the hypothesis. Sampling and hypotheses testing is very parallelisable, and we use OpenCV's existing implementation for this. There are two free parameters which govern performance. The first is the deviation τ_r (in pixels) below which a match is considered to support a hypothesis. The second is the minimum number of matches n_c which must support a hypothesis. We have found good default values to be $\tau_r = 12$ pixels and $n_c = 15$, and terminate RANSAC if more than 500 hypotheses have been sampled. If no pose has been found with more than n_c supported matches, then we say the uterus' pose cannot be estimated for that image.

4 Experimental Results

In this section we evaluate WBMTR using real *in vivo* data from three different human uteri captured before hysterectomy. We name these U_1, U_2 and U_3. The video data for each uterus is divided into two sections. The first is the exploratory section. The second is a *free-hand* section, where the surgical team observed the uterus but were free to move the laparoscope and cannula as they wished. The free section lasted approximately one minute and started immediately after the exploratory section.

Marker-based ground truth evaluation. Before starting the exploratory section, artificial markers were introduced on the uterus to give us accurate pose estimates that could be used for Ground-Truth (GT) evaluation. The surgeon marked the uterus with a coagulation instrument at 12-15 locations spread over the uterus body. This gave a set of small regions approximately 3mm in diameter which could be tracked. We show snapshots of these markers in Figure 3, middle-left column. We performed marker tracking using correlation-based tracking. The markers were tracked using a small patch surrounding each marker, and fitted using a 2D affine transform that was optimised with gradient descent. We manually verified the tracks, and manually initialised if the tracks became

lost. We then ran bundle adjustment [8] to compute the markers' positions in 3D, and the 3D poses of the uterus in each frame. If fewer than four markers were visible in a frame we said GT pose could not be estimated for that frame. Care was taken to avoid WBMTR exploiting the additional texture introduced by the markers. This was done by masking out the markers in each frame, thus preventing SURF from finding features on the markers.

Method comparison. We compared our method against the most recent SLAM system applied to laparoscopic images [7], which is based on EKF. The public code accompanying [7] uses FAST features [15]. We have found FAST to perform very poorly with the uterus because it comprises few corner-like features, and [7] could perform better using SURF features. We use this as the baseline method which we refer to as SLAM+SURF. We also tested the performance of PTAM [9]. However PTAM also uses FAST, and to work requires a good initialisation. This is done by tracking points in the first ten or so frames and performing SfM. For each uterus very few PTAM tracks could be found, despite the motion being smooth, and were insufficient to successfully initialise the maps.

We summarise the results of WBMTR against SLAM+SURF in Figure 4. The three rows correspond to results for the three uteri. We plot error with respect to position (in mm) in the first column, and error with respect to rotation (in degrees) in the second column. The vertical black line corresponds to the point in time when the exploratory section stopped, and the free-hand section started. WBMTR and SLAM+SURF give translation up to a global scale factor. This is a property of *all* visual SLAM and SfM methods. To give translation estimates in mm, it must be rescaled by a scale factor given by GT. For both methods, this was done by computing the least-squares scale factor which minimised the translation error with respect to GT. We can see from Figure 4 that WBMTR significantly outperformed SLAM+SURF, with respect to rotation and translation, and across both the exploratory and free-hand sections. As time increases the translation error of SLAM+SURF steadily increases, indicating that it suffers significant pose estimation drift. By contrast WBMTR suffers no such drift, and the translation error is usually below 2mm. There are some error spikes in WBMTR, particularly in the free-hand sections. This occurs when the uterus is only partially visible to the camera. In these cases only features on a fraction of the surface can be estimated, and hence we have fewer features with which to constrain pose. There are some gaps in the graphs for which error could not be computed. These occur when fewer than four markers were visible in a frame. In the third column of Figure 4 we show the 3D trajectories of the camera estimated by WBMTR and SLAM+SURF. GT is shown as blue dots. Here the performance improvement of WBMTR over SLAM+SURF is very clear. In the third and fourth columns of Figure 3 we show snapshots of the registered 3D model overlaid in two frames. One can see WBMTR handles cases when the surface is partially visible and occluded by tools. Note that the boundary of the reconstructed 3D model should not necessarily align to the occluding contour of the uterus in the image. This is because the 3D models are only partially reconstructed by SfM. The boundary

does not correspond to anything physical, but rather the region on the uterus for which SfM could reconstruct shape.

Fig. 3: Column 1: the dense 3D models built in Phase 1. Column 2: the coagulation markers. Columns 3&4: the registered models using WBMTR.

5 Conclusion and Future Work

We have presented a reliable and fast way to register the uterus in monocular laparoscopy using a novel two-phase approach. The approach differs to SLAM by decoupling 3D mapping and segmentation (done in Phase 1) from live registration (done in Phase 2). Phase 2 is achieved in realtime at approximately 26fps using standard hardware, and does not depend on successful registration in previous frames. It is thus simpler than EKF-SLAM and PTAM because it does not require switching between tracking and re-localisation. We have shown that our approach significantly outperforms EKF-SLAM for this problem. In the future we aim to enlarge our evaluation dataset and to explore the new opportunities that our method opens up for AR-assisted resection planning in uterine laparosurgery.

References

1. Agisoft: Photoscan, http://www.agisoft.ru/products/photoscan
2. Bay, H., Ess, A., Tuytelaars, T., Van Gool, L.: Speeded-up robust features (SURF). Comput. Vis. Image Underst. 110(3) (Jun 2008)
3. Collins, T., Bartoli, A.: Towards live monocular 3D laparoscopy using shading and specularity information. In: IPCAI. pp. 11–21 (2012)
4. Davison, A.J., Reid, I.D., Molton, N.D., Stasse, O.: MonoSLAM: Real-time single camera SLAM. PAMI 29(6), 1052–1067 (2007)
5. Fischler, M.A., Bolles, R.C.: Random sample consensus. Commun. ACM (1981)
6. Grasa, O.G., Civera, J., Guemes, A., Muoz, V., Montiel, J.M.M.: EKF monocular SLAM 3D modeling, measuring and augmented reality from endoscope image sequences. In: AEMI-ARCAI (2009)
7. Grasa, O.G., Civera, J., Montiel, J.M.M.: EKF monocular SLAM with relocalization for laparoscopic sequences. In: ICRA (2011)

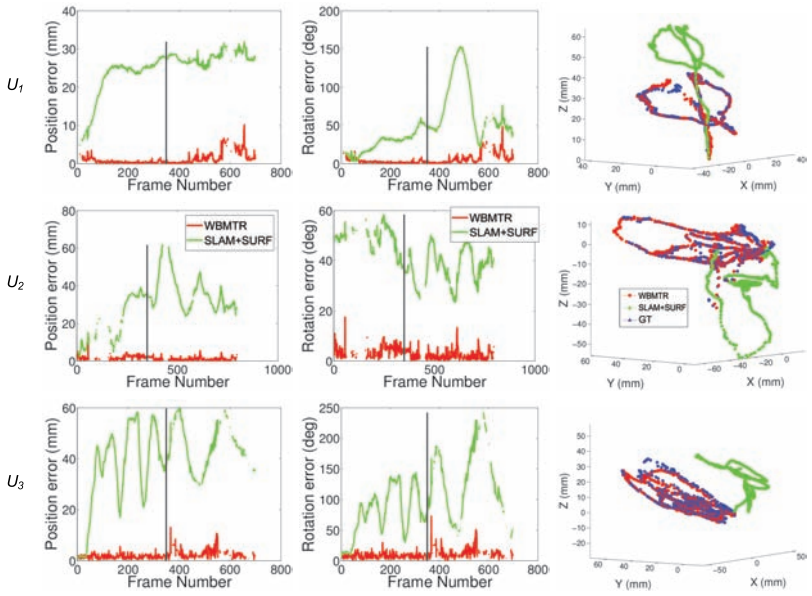

Fig. 4: In vivo evaluation of pose estimation accuracy for three uterus datasets.

8. Hartley, R.I., Zisserman, A.: Multiple View Geometry in Computer Vision. Cambridge University Press, ISBN: 0521540518 (2004)
9. Klein, G., Murray, D.: Parallel tracking and mapping for small AR workspaces. In: ISMAR. Nara, Japan (November 2007)
10. Lepetit, V., Moreno-Noguer, F., Fua, P.: EPnP: An accurate O(n) solution to the PnP problem. IJCV (2009)
11. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. IJCV (2004)
12. Malti, A., Bartoli, A., Collins, T.: Template-based conformal shape-from-motion-and-shading for laparoscopy. In: IPCAI (2012)
13. Mountney, P., Stoyanov, D., Davison, A.J., Yang, G.Z.: Simultaneous stereoscope localization and soft-tissue mapping for MIS. In: MICCAI (2006)
14. Nakamoto, M., Nakada, K., Sato, Y., Konishi, K., Hashizume, M., Tamura, S.: Intraoperative magnetic tracker calibration using a magneto-optic hybrid tracker for 3-d ultrasound-based navigation in laparoscopic surgery. TMI (2008)
15. Rosten, E., Drummond, T.: Machine learning for high-speed corner detection. In: In European Conference on Computer Vision. pp. 430–443 (2006)
16. Simpfendörfer, T., Baumhauer, M., Müller, M., Gutt, C., Meinzer, H., Rassweiler, J., Guven, S., Teber, D.: Augmented reality visualization during laparoscopic radical prostatectomy. Endourology (2011)
17. Strecha, C., von Hansen, W., Gool, L.J.V., Fua, P., Thoennessen, U.: On benchmarking camera calibration and multi-view stereo for high resolution imagery. In: CVPR (2008)
18. Totz, J., Mountney, P., Stoyanov, D., Yang, G.Z.: Dense surface reconstruction for enhanced navigation in MIS. In: MICCAI (2011)