Fast Semi-Dense Surface Reconstruction from Stereoscopic Video in Laparoscopic Surgery

Johannes Totz[1], Stephen Thompson[1], Danail Stoyanov[1], Kurinchi Gurusamy[2], Brian R. Davidson[2], David J. Hawkes[1], and Matthew J. Clarkson[1]

[1] Centre for Medical Image Computing
University College London, UK
[2] Royal Free Hospital
London, UK

{j.totz, s.thompson, danail.stoyanov, k.gurusamy, b.davidson, d.hawkes, m.clarkson}@ucl.ac.uk

Abstract Liver resection is the main curative option for liver metastases. While this offers a 5-year survival rate of 50%, only about 20% of all patients are suitable for laparoscopic resection and thus being able to take advantage of minimally invasive surgery. One underlying difficulty is the establishment of a safe resection margin while avoiding critical structures. Intra-operative registration of patient scan data may provide a solution. However, this relies on fast and accurate reconstruction methods to obtain the current shape of the liver. Therefore, this paper presents a method for high-resolution stereoscopic surface reconstruction at interactive rates. To this end, a feature-matching propagation method is adapted to multi-resolution processing to enable parallelisation, remove global synchronisation issues and hence become amenable to a GPU-based implementation. Experiments are conducted on a planar target for reconstruction noise estimation and a visually realistic silicone liver phantom. Results highlight an average reconstruction error of 0.6 mm on the planar target, 2.4–5.7 mm on the phantom and processing times averaging around 370 milliseconds for input images of size 1920 x 540.

1 Introduction

Resection of a segment or lobe of the liver in metastatic or primary liver cancer is the main curative option. This is traditionally done in an open procedure, resulting in a large wound on the patient's abdomen to allow access for the surgeon to palpate and identify important structures within the liver and distinguish normal liver from tumour. A minimally invasive approach instead might reduce trauma, infection risk, post-operative pain and cosmetic issues. However, difficulties in estimating a safe resection margin, proximity to blood vessels and tumour size, etc deny more than 80% of patients this option. In order to increase suitability for the laparoscopic approach, improved surgical guidance and navigation is required.

To this end, robust registration methods are necessary that need as input a physically-based deformable model of the liver [1] and an up-to-date estimate of the organ's surface geometry serving as a deformation target [2,3]. Reconstructing the organ surface in real-time and in sufficient detail is a challenging problem due to view-dependent specular highlights and the relatively uniform appearance of the liver. This also complicates registration because only a small part of it is visible. Existing methods [4,5] use natural features like the falciform ligament and inferior edges along liver segments. As the laparoscope is relatively easy to navigate looking at these features, recovering their position and shape from video should be possible using stereoscopic reconstruction methods.

In building a stereo-matching algorithm, a popular choice is to perform a pyramidal search, reducing the necessary disparity search range. This is because larger features are captured in lower-resolution pyramid levels without increasing the disparity range on that level [6]. A common approach is to filter and subsample the images into Gaussian pyramids first. Then find disparity on low-resolution levels, upscale these to the next level and refine with higher-resolution image data. This, however, easily breaks object boundaries and special care must be taken to consider the effects of down-sampling [7]. Also, while this approach appears to be easily parallelisable, the output degrades quickly. Other recent methods allow real-time reconstruction from either low-resolution [8,9,10] or high-definition video [11].

This paper proposes a stereo-matching strategy based on a coarse-to-fine pyramidal approach, adapted from sequential local match-propagation [12]. Contrary to other approaches that process image pyramid levels in sequence and upscale the results of a lower-resolution level to the next one, the proposed novel approach traverses the pyramid vertically by starting on the pyramid tip and traces out left-right matches to increasing image resolution. This vertical propagation thereby enables correspondence search window sizes to be kept small as a large high-resolution window is equivalent to a small low-resolution one, similar to existing coarse-to-fine approaches. However, vertical propagation also enables bounding of hot-loop data structures in size. This is a prerequisite to efficient GPU-implementation where low-latency on-chip memory is scarce. Multi-threaded operation follows naturally, allowing stereoscopic surface reconstruction at interactive rates from high-resolution video. Recovering the shape of the liver anatomy can then be used to register a deformable liver model, reducing the time required for an initial registration, or updating an existing registration during the procedure.

2 Method

Figure 1 depicts the processing pipeline of the proposed method for an integrated system. After initial transfer of the laparoscopic video frames to GPU memory, left and right channels are prepared for processing, followed by a matching kernel. Highlighted core steps are described in more detail in the following sections.

Disparity filtering and triangulation follow standard procedures and are thus not described further.

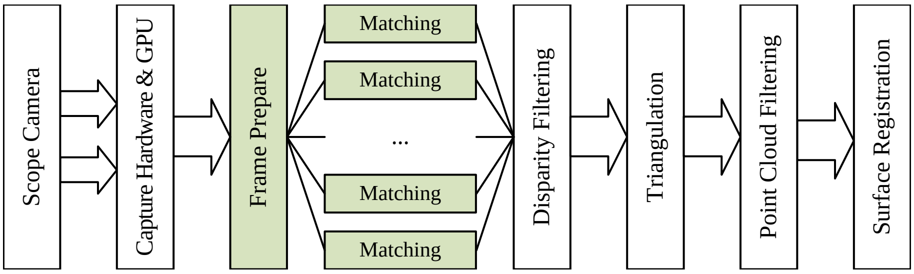

Figure 1. Pipeline of the proposed method in context for this application. The incoming video frames are captured and transferred to GPU memory for stereoscopic matching and other processing. The highlighted core steps are presented in this paper.

2.1 Frame Preparation

Prior to matching, frame preparation is necessary. Input images left I_0 and right I_1 are cropped to a size that is a multiple of 32. This is necessary to ensure a well-formed 2:1 image pyramid with sufficient levels. Cropping is centred so that only a few pixels along the border, which rarely contain usable features, are lost. Afterwards, the cropped RGBA images are converted to greyscale and each resampled with a box filter into an image pyramid, P_0 and P_1 respectively, at successively lower resolutions. For each level l of each pyramid P^l, quantities required for fixed-window-size zero-mean normalised cross correlation (ZNCC) are precomputed. In addition, a bit mask is computed for textureless areas by checking for a non-zero horizontal and vertical pixel gradient, preventing gross mismatches in the correspondence propagation.

2.2 Match Propagation – Single-threaded

While the proposed method is motivated by a multi-threaded GPU-amenable design and implementation, it appears reasonable to describe the matching process for a single thread first.

The overall left-to-right matching strategy takes advantage of an existing match and propagates more matches around this initial “seed” position, avoiding a large amount of false matches that could occur otherwise. Matching starts from the lowest-resolution pyramid level l that is large enough to contain the various pixel windows described below. At this resolution, the disparity for intended stereoscopic cameras is sufficiently close to 1 or 2 pixels, removing the need for explicit feature match initialisation between left and right views for the initial seed. Thus, at the very beginning, an initial seed $k := \{x_0, y_0, x_1, y_1\}$ is set to the image centres. Figure 2 illustrates key elements.

Broadly speaking, each iteration of matching performs:

1. Generation of a list of candidate matches around the current seed.
2. Establishment of global uniqueness per level.
3. Initialisation of a new seed for pyramid level $l + 1$ from the established matches and jump to $l + 1$, starting at (1).
4. On the highest-resolution level, keep matching horizontally.
5. Once the list of candidate matches is exhausted, jump back to previous level $l - 1$ and continue at (1).

More specifically, for step (1): Around each seed k, compute ZNCC [13] for a $c \times c$ pixel sized correlation window C in P_0 and P_1, shifted by the neighbourhood window N of up to $n \times n$ pixels in either dimension (allowing matching to skip across poorly defined areas) in both left and right image simultaneously. In the right image only, the correlation window is shifted by an additional search window S of $s \times s$ pixels (this adapts the computed disparity to changes with perspective). This produces a list of up to $n \times n \times s \times s$ left-right coordinate pairs $q := \{x_0, y_0, x_1, y_1, b\} \in Q_0^l$, each with a corresponding correlation b. If b is smaller than a certain threshold $b_<$ then that entry is dropped.

Entries in Q^l are sorted according to numerical value b, highest first. Each entry is read, and its left-right-coordinates written to the disparity map $d := \{x_0, y_0, x_1, y_1\} \in D^l$ (implemented as a 2-channel image, storing x_1, y_1 at each x_0, y_0) for level l if no other match has been recorded for either x_0, y_0 or x_1, y_1 already. If instead an entry already exists in D^l then that particular q is removed from Q^l. Once Q^l has been processed (leaving its entries intact; these will serve as new seeds later), its top entry is used to initialise a new seed for level $l+1$ by multiplying its coordinates by two (step (3) in the list above). Processing then continues with step (1) again at the next level.

Once the highest-resolution level is reached, match propagation continues horizontally (step (4)). Eventually, the processing in step (1) will not add new entries to Q^l due to poor correlation between left and right pixel patches. At this point, propagation stops at the current level and returns to $l-1$, continuing with Q^{l-1} at step (1) where it left off (step (5) in the list above).

2.3 Multi-threaded Matching

The matching strategy described above can easily be run multi-threaded. Instead of a single starting position, many are chosen with pseudo-random offsets. Each thread processes P_0 and P_1 from its assigned seed, independently of other threads. However, as many threads would start off from effectively the same starting conditions they would also produce exactly the same result yielding no improvement in performance or match coverage. Therefore, divergence is triggered by employing a “permissible thread map” T, a bitmap the size of the input images, labelling each pixel for which thread is allowed to process it. The map T is generated once at start-up time representing a simple block structure of $4 \, \mathrm{x} \, 2$

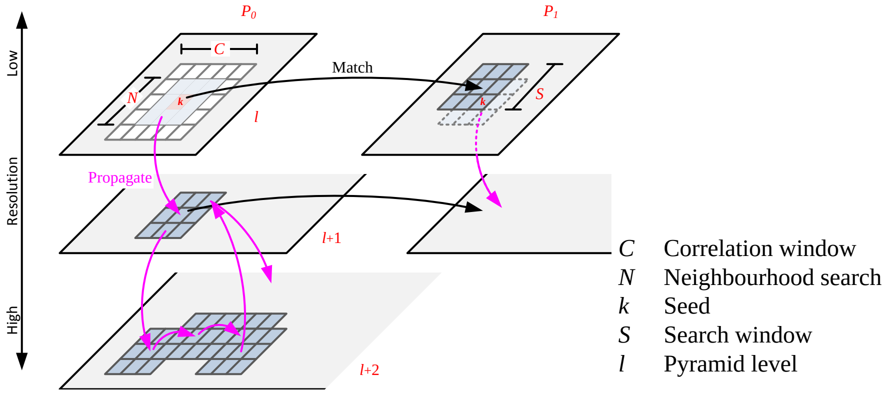

Figure 2. Illustration of how match propagation proceeds vertically across the pyramid. Symbols are further explained in the text of Sec. 2.2.

partitions, yielding 8 different blocks that map onto an 8-bit thread-ID bit pattern. It is then filtered and downsampled into the remaining pyramid levels T^l by OR-ing thread-ID bits from the higher-resolution level, effectively blurring the boundaries between blocks as the pyramid level resolution decreases.

A potential performance bottleneck is the priority queue Q used to store match candidates. While each thread has its own instance per level, it is initially unbounded in size, posing a challenge for an efficient GPU implementation where access patterns to memory are critical. With the proposed method however, it turns out that maximum observed queue sizes are short in practice, only slightly larger than the number of newly arriving candidates in match-step (1). Therefore, constraining the size of Q for each thread allows the fitting of hot data in performance-critical shared memory and registers.

3 Experiments & Results

For all experiments described below, Table 1 lists the parameter values used for the propagation. All experiments are performed on an NVIDIA Quadro K5000 card. Stereo-pairs were recorded with a Viking 3DHD Vision System Dual Channel 30º laparoscope (formerly Viking Systems, Inc., USA). It provides two SDI outputs at 1080i at 59.9 Hz. The bottom field was discarded from both channels as interlacing interferes heavily with matching. Intrinsic and extrinsic camera calibration were determined using functions implemented in OpenCV. Video frames were then undistorted. No further preprocessing was performed.

3.1 Plane Experiment

In stereo-matching, small errors can be amplified easily by the stereo-rig geometry. This manifests in large spread in the z-coordinate. To assess this effect in combination with the above mentioned laparoscope, a flat piece of paper was

Table 1. Propagation parameters used for experimental results. They were determined empirically.

Parameter	Symbol(s)	Value	Units
Search window size	S: $s \times s$	$s = 3$	pixels
Correlation window size	C: $c \times c$	$c = 5$	pixels
Neighbourhood window size	N: $n \times n$	$n = 3$	pixels
Correlation threshold	$b_<$	$b_< = 0.6$	—

printed with a noise pattern and filmed at an angle of approximately 30 degrees by pointing the laparoscope straight down. The distance from lens to surface was in the range of 4–7 cm. The resulting stereo-pair was then processed by sequential matching [12] (with a correlation window 19 x 19 pixels) and the proposed method (with parameters in Table 1), yielding two disparity maps. Figure 3 illustrates these. The disparity maps were triangulated into a point cloud using previously obtained camera parameters, and a plane was fitted through each. These planes serve as a silver standard regarding reconstruction noise: computing an RMS distance of reconstructed points to estimated plane yields 0.42 mm for sequential and 0.67 mm for the proposed method.

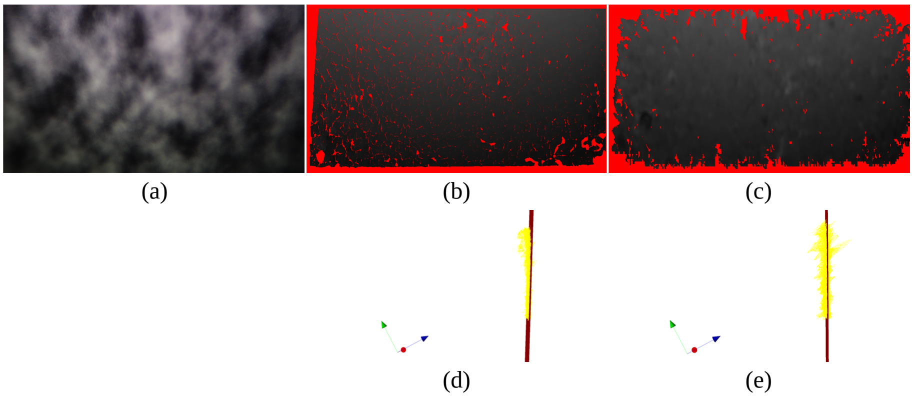

Figure 3. Textured plane imaged at a 30 degree angle for estimating reconstruction noise. (a) shows the left channel of the stereo pair used to reconstruct the disparity maps for (b) the sequential method and (c) the proposed method. Brighter colour corresponds to higher disparity. The corresponding point clouds and fitted planes are shown in (d) and (e), respectively. The axis icon signifies the camera location.

3.2 Phantom Experiment

To evaluate the proposed method in a more realistic scenario, a flexible visually realistic human liver phantom (Healthcuts, London, UK) was custom-made. It consists of a deformable main organ body made of silicone and a rigid carbon fibre base with nine rigid “prongs” holding the body in place, allowing it to be taken off and put back on repeatably (Fig. 4a-b). The phantom was CT-scanned at 0.98 x 0.98 x 0.6 mm voxel resolution, an ISO-intensity surface extracted using Marching Cubes, and edited to remove irrelevant geometry.

The endoscope was positioned at a surface distance of 4–7 cm, making sure at least three prong tips were visible. The tips were marked in the left and right images, triangulated to 3D and aligned to the CT-scan with a least-squares optimisation. This registration was used as the gold standard location of the phantom relative to the camera lens. The corresponding fiducial registration error (FRE) for this alignment is reported in Table 2. The silicone phantom was then replaced onto the prongs and imaged. The left and right image were undistorted, and processed by the sequential method and the proposed one, yielding a point cloud. For each point in the output point cloud, the closest distance to the phantom surface was computed and aggregated into a root-mean-squared error (RMSE) for each method. These steps were repeated for three individual data sets, taken from different angles of the same phantom. Table 2 shows that the proposed method produces slightly higher errors compared to the sequential method, however at a fraction of the run time. Figure 4c-d show unfiltered reconstructed point clouds, overlaying the two methods for comparison. The red point cloud is the sequential method, and the yellow cloud the proposed method. As can be seen, the latter is slightly more noisy. Most of these mismatches are caused by view-dependent specular highlights, which the sequential method can match around more easily as its propagation queue has a global view on all possible match candidates.

All runtime measurements in Table 2 & 3 were conducted on a PC running Windows 7, 16 GB RAM, NVIDIA Quadro K5000 with 4 GB RAM and Intel Xeon E5-2609 at 2.4 GHz dual socket, four cores each.

Table 2. Reconstruction error on the liver phantom, using RMSE between reconstructed points and CT phantom surface as the metric. Input stereo pairs have a resolution of 1920 x 540 pixels. The data set number refers to Fig. 4.

Data set	Fiducial Registration Error	Proposed GPU	Sequential CPU [12]
1	1.3 mm	2.4 mm, 330 ms	1.8 mm, 2855 ms
2	2.1 mm	5.3 mm, 409 ms	4.5 mm, 3333 ms
3	2.6 mm	5.7 mm, 397 ms	2.5 mm, 2875 ms

Existing literature [11,8] compares reconstruction results on the Hamlyn Heart phantom data set [12]. The proposed method reconstructs the surface with an RMSE of 3.2 mm and an average error of 2.1 mm. In comparison, the sequential method, as implemented, reconstructs an RMSE of 3.0 mm and an average error of 2.1 mm (compared to 3.9 and 2.4 mm respectively, as reported previously [8]).

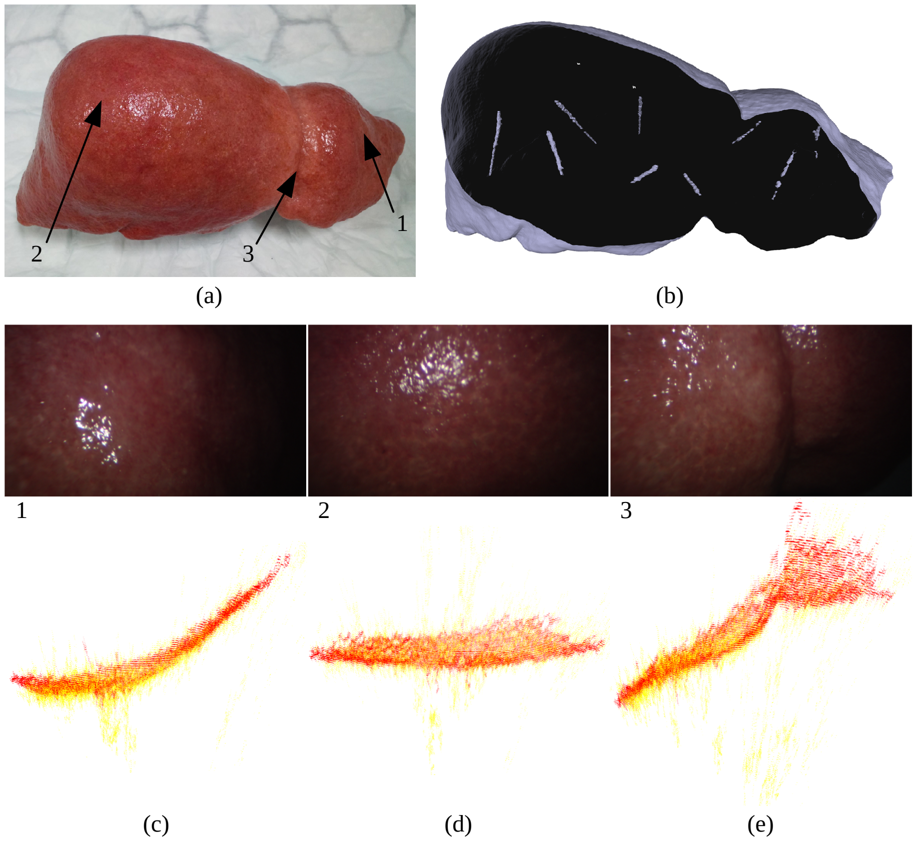

Figure 4. Silicone phantom of a human liver, manufactured to be visually realistic, with carbon fibre prongs holding the deformable main body in position. (a) shows the main body, mounted on its base; (b) shows the mesh derived from a CT-scan including the prongs inside it. (c)-(d) show laparoscope images and corresponding unfiltered reconstructions for three different view points, overlaying both sequential and parallel method, displayed top-down.

3.3 Runtime Evaluation

The proposed method has been integrated with the NVIDIA Digital Video Pipeline, allowing direct transfer of SDI-supplied high resolution video to GPU memory. Once stereoscopic video frames have arrived in texture memory as RGBA arrays, the frame preparation process is started, followed by the matching kernel described above. Table 3 shows average processing times on a NVIDIA Quadro K5000 at different input resolutions. Timing resolution is in the order of one millisecond. The frame preparation step is dependent only on image resolution, image content has no impact on timing, hence variation is effectively zero given the timer resolution. The actual matching step however does depend on image content as the presence of gradients determine propagation. The time required to copy the result back to host memory is specifically excluded because it is expected that a streamlined registration system will perform triangulation, point cloud filtering, etc on the GPU too. Table 3 compares the runtime of the sequential method on the aforementioned Xeon CPU, highlighting a performance increase of 3–9 fold.

Table 3. Runtime of the proposed algorithm, averaged over a number of different sequences, compared to the sequential method. All reported times are in milliseconds, with μ being the mean and σ the standard deviation.

Image size		Prepare	Match		Total	Seq. CPU		Speed up
input	cropped to		μ	σ	time	μ	σ	
360 x 288	352 x 288	1.1	73.2	11.3	74.3	253.6	8.4	3.4
1920 x 540	1920 x 512	4.9	373.9	45.6	378.8	2879.9	225.3	7.6
1920 x 1080	1920 x 1056	9.5	481.2	79.0	490.7	4447.9	260.4	9.1

4 Discussion & Conclusions

The proposed method is able to perform stereo-matching at interactive frame rates with an accuracy suitable for laparoscopic applications. Contrary to many existing methods, the proposed one does not rely on stereo-rectified images; it performs a 2D search instead of a 1D search along the epipolar line. While this increases processing cost significantly, it increases the number of successfully matched pixels as each seed is free to propagate along image structures in any direction. However, stepping the neighbourhood window N simultaneously for both left and right ensures that matches will not criss-cross (observing a local 2D ordering constraint). Also, ZNCC is a very expensive cost function. However, it was chosen for its robustness against radiometric changes between different views. It was found to be reliable [13] on the Middlebury data set, however, not the top performer. Contrary to a controlled lab environment, minimally invasive surgery exhibits severe radiometric issues due to uncontrollable auto-gain in the camera, non-uniform lighting and inter-tissue reflections. As the algorithm is effectively a variant of winner-takes-all in the match propagation phase and first-come-first-served with respect to multi-threading, its output depends on timing and scheduling details. While this sounds bad from a computational point of view, it has no impact in practice and relaxing a strict no-race-condition requirement allows for significant improvements to execution speed. A particular problem not addressed yet is related to object segmentation: the camera views the abdominal cavity, possibly with many unrelated structures in view. This will be addressed in future work.

Acknowledgements This publication presents independent research commissioned by the Health Innovation Challenge Fund (HICF-T4-317), a parallel funding partnership between the Wellcome Trust and the Department of Health. The views expressed in this publication are those of the authors and not necessarily those of the Wellcome Trust or the Department of Health. The authors also thank NVIDIA for the kindly donated Quadro K5000 and SDI capture card.

References

1. Peterlik, I., Duriez, C., Cotin, S.: Modeling and real-time simulation of a vascularized liver tissue. In Ayache, N., Delingette, H., Golland, P., Mori, K., eds.: Medical Image Computing and Computer-Assisted Intervention – MICCAI 2012. Volume 7510 of LNCS., Springer Berlin/Heidelberg (2012) 50–57
2. Pratt, P., Stoyanov, D., Visentini-Scarzanella, M., Yang, G.Z.: Dynamic guidance for robotic surgery using image-constrained biomechanical models. In Jiang, T., Navab, N., Pluim, J., Viergever, M., eds.: Medical Image Computing and Computer-Assisted Intervention – MICCAI 2010. Volume 6361 of LNCS., Springer Berlin/Heidelberg (2010) 77–85
3. Haouchine, N., Dequidt, J., Berger, M.O., Cotin, S.: Deformation-based augmented reality for hepatic surgery. In: Medicine Meets Virtual Reality, MMVR 20. (2013)
4. Kingham, T.P., Jayaraman, S., Clements, L.W., Scherer, M.A., Stefansic, J.D., Jarnagin, W.R.: Evolution of image-guided liver surgery: Transition from open to laparoscopic procedures. Journal of Gastrointestinal Surgery **17**(7) (July 2013) 1274–1282
5. Clements, L.W., Chapman, W.C., Dawant, B.M.., Jr, R.L.G., Miga, M.I.: Robust surface registration using salient anatomical features for image-guided liver surgery: Algorithm and validation. Medical Physics **35**(6) (June 2008) 2528–2540
6. Meerbergen, G.V., Vergauwen, M., Pollefeys, M., Gool, L.V.: A hierarchical symmetric stereo algorithm using dynamic programming. International Journal of Computer Vision **47**(1–3) (April 2002) 275–285
7. Sizintsev, M., Wildes, R.P.: Coarse-to-fine stereo vision with accurate 3D boundaries. Image and Vision Computing **28**(3) (March 2010) 352–366
8. Chang, P.L., Stoyanov, D., Davison, A.J., Edwards, P.".: Real-time dense stereo reconstruction using convex optimisation with a cost-volume for image-guided robotic surgery. In Mori, K., Sakuma, I., Sato, Y., Barillot, C., Navab, N., eds.: Medical Image Computing and Computer-Assisted Intervention – MICCAI 2013. Volume 8149 of LNCS., Springer Berlin/Heidelberg (2013) 42–49
9. Mei, X., Sun, X., Zhou, M., Jiao, S., Wang, H., Zhang, X.: On building an accurate stereo matching system on graphics hardware. In: 2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops). (2011) 467–474
10. Richardt, C., Orr, D., Davies, I., Criminisi, A., Dodgson, N.A.: Real-time spatiotemporal stereo matching using the dual-cross-bilateral grid. In Daniilidis, K., Maragos, P., Paragios, N., eds.: Computer Vision – ECCV 2010. Volume 6313 of LNCS., Springer Berlin/Heidelberg (2010) 510–523
11. Roehl, S., Bodenstedt, S., Suwelack, S., Kenngott, H., Mueller-Stich, B.P., Dillmann, R., Speidel, S.: Dense GPU-enhanced surface reconstruction from stereo endoscopic images for intraoperative registration. Medical Physics **39**(3) (March 2012) 1632–1645
12. Stoyanov, D., Visentini-Scarzanella, M., Pratt, P., Yang, G.Z.: Real-time stereo reconstruction in robotically assisted minimally invasive surgery. In Jiang, T., Navab, N., Pluim, J., Viergever, M., eds.: Medical Image Computing and Computer-Assisted Intervention – MICCAI 2010. Volume 6361 of LNCS., Springer Berlin/Heidelberg (2010) 275–282

13. Hirschmueller, H., Scharstein, D.: Evaluation of stereo matching costs on images with radiometric differences. IEEE Transactions on Pattern Analysis and Machine Intelligence **31**(9) (2008) 1582–1599