From Stereo Image Sequences to Smooth and Robust Surface Models using Temporal Information and Bilateral Postprocessing

S. Röhl[+], S. Speidel[+], D. Gonzalez-Aguirre[+], S. Suwelack[+], H. Kenngott[*], T. Asfour[+], B. P. Müller-Stich[*], R. Dillmann[+]

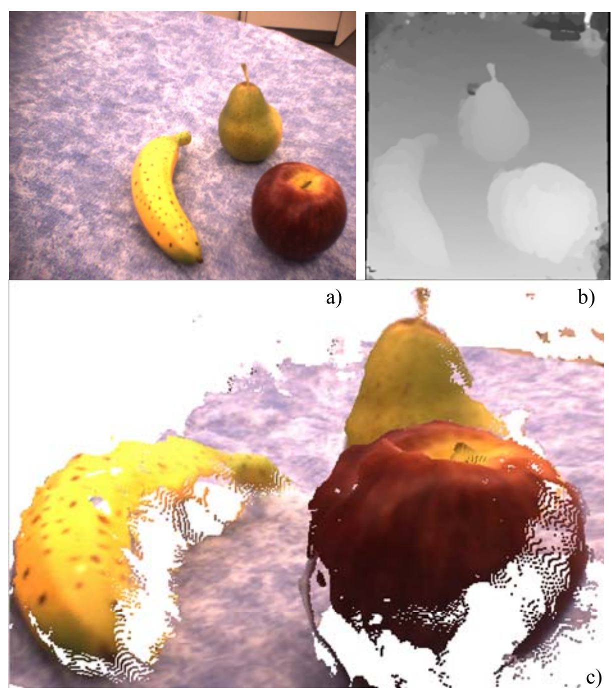
Abstract—**Reconstruction of surface models from camera images has many applications in robotics such as surface registration or object recognition. In this paper, we describe a workflow in which we extract depth information from stereo image sequences to generate a surface model. We present our solutions to correspondence analysis, disparity correction and refinement, as well as 3D reconstruction, point cloud smoothing and meshing. One important feature of the correspondence analysis that we evaluate in detail is the use of temporal information. Another emphasis is on correcting and smoothing the disparity images as well as the reconstructed point cloud without losing too much detail. We, hence, introduce our application of the Bilateral filter on disparity images and our usage of least squares smoothing. The components of the workflow were evaluated using three image sources: Endoscopic images from the daVinci® telemanipulator; images from a stereo camera integrated in the ARMAR III humanoid robot; synthetic data. Depending on the image resolution and the application, the workflow reconstructs surface models in real-time. We show that by using temporal information we obtain more accurate and robust correspondences. Additionally, the Bilateral filter was especially useful in refining the correspondences extracted from endoscopic images as well as the synthetic data sets, whereas the least squares method showed good results in smoothing the point cloud of ARMAR III images. Overall, the presented approach achieves good results for different camera settings and image types, especially with respect to the real-time requirement.**

I. Introduction

When working in an unknown or unstructured environment, robotic systems have to possess some kind of perception of their surrounding area and the objects within. A device often used in robotics is a stereo camera. It is low-cost, easy to integrate and similar to human perception. In contrast to modern 3D cameras such as Kinect, stereo cameras are more flexible and easier to miniaturize. They have a larger range and can inspect objects close to the cameras. Tasks such as object recognition, motion analysis or scene reconstruction can be solved by extracting useful information from stereo images. In particular, the reconstruction of the surface of observed objects or the whole area is an important task. We consider two robotic applications where such a surface model can be used.

The first application for this work is surface registration in the context of minimally invasive surgery using the daVinci® telemanipulator. Here, the surgeon controls the robotic arms via a console while observing the intervention on a stereo display [1]. To further facilitate the intervention, navigation information from preoperative imaging can be visualized via Augmented Reality. In order to adapt the navigation data to the changing intraoperative environment, a representation of the deformation has to be acquired that can be used for registration. Here, we use the daVinci® stereo endoscope to intraoperatively extract depth information and build a surface model of the surgical site. Previous work in surface reconstruction from endoscopic images are e.g. from Stoyanov et al. [2], Devernay et al. [3] and Vagvolgyi et al. [4].

The second application is object categorization based on surface models created from images of a stereo camera system [5] (Fig. 1). This is an important task for service robots, such as the humanoid robot ARMAR III [6]. ARMAR III performs service-tasks in a household scenario. Therefore, it has to recognize the objects it interacts with. The 3D shape of an object can generate features such as



a) b) c)

Fig. 1. a) Image from ARMAR III; b) disparity image; c) reconstructed point cloud

Manuscript accepted Oct 5, 2011. The present research was conducted within the setting of the "Research training group 1126: Intelligent Surgery – Development of new computer-based methods for the future workplace in surgery" founded by the German Research Foundation and partly sponsored by the European Social Fund.

The + authors are with the Karlsruhe Institute of Technology, Adenauerring 2, Karlsruhe-Germany (e-mail {roehl, stefanie.speidel, suwelack, asfour, dillmann}@kit.edu, gonzalez@ira.uka.de).

The * authors are with the General, Visceral and Transplantation Surgery, Heidelberg University Hospital (e-mail {Hannes.Kenngott, Beat.Mueller }@med.uni-heidelberg.de).

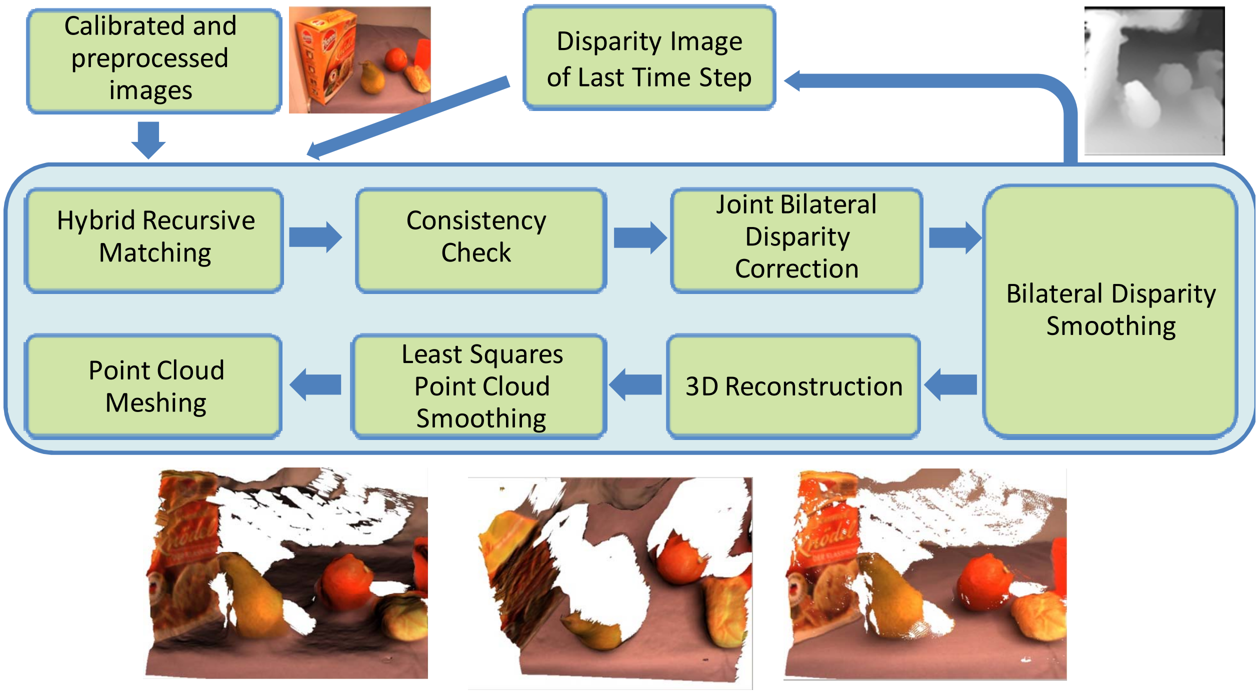


Fig. 2: Overview of the surface reconstruction workflow

shape distributions [7] or spin images [8] that discriminate it from other objects. These features can then be compared to features from previously trained objects in order to correctly categorize the objects found in the images.

In this paper, we present an approach in which we extract depth information from stereo image sequences to generate a surface model. Therefore, we present solutions to correspondence analysis, correction and refinement, as well as 3D reconstruction, normal extraction, point cloud smoothing and meshing (Fig. 2).

The approach is applied in completely different scenarios. In minimally invasive surgery, the camera is stationary for most of the time while the observed scene is deforming. Additionally, the baseline between the two cameras is very small which impacts the accuracy of the reconstructed 3D points. In humanoid robotics, the robot moves around while the objects are often stationary and non-deforming. Both applications suffer from occlusions.

As both applications are set in an interactive environment that changes continuously, the aim is to reconstruct the surfaces in real-time. To achieve high frame rates, we parallelized the methods where it was possible and implemented them on the GPU.

One important feature that we evaluate in detail is the use of temporal information for correspondence analysis. This is motivated by having image sequences with often little differences between adjacent images. This also implicates a method that is fast enough to benefit from the temporal information. Similar strategies are pursued by Richard et al. [9], Davis et al. [10] or Leung et al. [11].

Another emphasis was on denoising of the 3D points without losing too much detail. The impact on the quality of post processing of the disparity images using the Bilateral filter is profoundly evaluated. Here, we use two variants of the filter. A joint-bilateral filter is used for correspondence correction especially on object edges and occluded areas, whereas a normal Bilateral filter is used to smooth the disparity images. We also present the application of least squares normal calculation to the reconstructed point clouds for further smoothing.

The approach is applied to several image sequences from the two scenarios as well as synthetic images to prove its reliability. It reconstructs an accurate and robust smooth and detailed surface model from stereo images in real-time.

II. METHODS

Fig. 2 shows an overview of the approach. The algorithms are implemented in C/C++ using Qt, VTK from Kitware and IVT, a vision library [12]. Part of the approach is implemented on the GPU using CUDA.

A. *Image acquisition and preprocessing*

For our experiments, we used images from two stereo endoscopes, one from FA WOLF and the other from the da Vinci® system (Fig. 3), as well as from two stereo cameras for wide and narrow angle that are integrated in the head of ARMAR III (Fig. 1). We also used synthetic stereo sequences presented by Richardt et al[1] [9].

The kind of preprocessing depends on the image source. While the synthetic images are not preprocessed, the camera images are rectified and undistorted in order to facilitate the correspondence analysis and improve the 3D reconstruction.

B. *Correspondence analysis*

The goal of correspondence analysis in stereo images is to find points in both images that are the projection of the same

[1] http://www.cl.cam.ac.uk/research/rainbow/projects/dcbgrid/

world point into the image planes. The 3D reconstruction uses the detected correspondences and the geometry of the cameras to calculate the three-dimensional structure of the observed scene.

In our approach, we use the *Hybrid Recursive Matching* method [13]. This algorithm was originally developed for video conferencing systems, a setting where the baseline between the cameras is often very large. It uses information from the spatial and temporal neighbourhood to recursively generate a dense disparity image in real-time. As input it needs the two images of the stereo camera, the disparity image of the last time step and the already calculated disparities of the current time step.

In a two-stage process, the block recursion and the pixel recursion step, it calculates a new disparity value for the current pixel by choosing between four different candidates in the other image. For each candidate, the similarity to the current pixel is calculated using block matching. The candidate with the most similar neighbourhood wins. While the block recursion step ensures a smooth disparity distribution especially in textureless regions, the pixel recursion step introduces new values in regions of discontinuity. The recursive nature of the algorithm makes sure that disparities are propagated from textured regions into textureless regions or regions with occlusions. For more details about the algorithm, we refer to the original paper.

The adaptations of this algorithm in order to work with our applications can be seen in [14]. To make it more independent of local and global intensity variations in both images, another similarity measure was used. We also extended the basic algorithm to subpixel precision by calculating new subpixel disparities in the block and pixel recursion step. In this way, the 3D points can be reconstructed more precisely.

The correspondence analysis step still produces mismatches in textureless regions or near occluded areas which have to be corrected. The detection is done via a double consistency check. For each pixel in one image, its disparity and the disparity of the corresponding pixel in the other image are compared. If they exceed a predefined threshold, these disparities are rejected. An overview of the correction methods can be seen in [14].

C. *Bilateral postprocessing*

Without smoothing, the resulting disparity image is still noisy which would lead to a noisy point cloud, especially when subpixel disparities are important. Additionally, the correction methods from [14] still produce erroneous disparities near object edges, especially in partly-occluded areas. Traditional smoothing filters don't preserve edges which reduces the level of detail of the surface. To prevent that, we use the Bilateral filter. This filter calculates a new value for one pixel depending on the weighted average of neighboring pixel values. The idea is to weight the influence of a neighboring pixel not only by its spatial distance but also by its pixel value similarity in order to reduce the influence of pixels separated by an edge. A good introduction on Bilateral filtering is found in [15]. In contrast to other work, e.g. [9] or [16], where it is used for correspondence analysis, we use is correcting false disparities and smoothing the disparity images.

A popular version of the filter uses two Gaussian filters G_{σ_s} and G_{σ_r}. G_{σ_s} is applied to the spatial neighborhood and decreases the influence of distant pixels. G_{σ_r} decreases the influence of pixels with different values. Given a pixel $p(x, y)$, its disparity value $I(p)$ and its surrounding neighborhood N, a new value $I_{new}(p)$ is obtained by

$$I_{new}(p) = \frac{1}{W_p} \sum_{q \in N} G_{\sigma_s}(\|p - q\|)\, G_{\sigma_r}(|I_p - I_q|)\, I_q$$

with the normalization factor

$$W_p = \sum_{q \in N} G_{\sigma_s}(\|p - q\|)\, G_{\sigma_r}(|I_p - I_q|).$$

The Bilateral filter is controlled by two parameters. The spatial range parameter σ_s sets the size of the neighborhood that is used for smoothing. The intensity range parameter σ_r controls the impact of G_{σ_r}.

It is straight-forward to use this filter on disparity images, since the pixel values encode the distance of a 3D point to the camera reconstructed from the corresponding points. In this way, the disparity image is implicitly separated into regions that have the same or a smoothly changing distance to the camera. The filter only smoothes in these regions, areas with discontinuities are preserved.

Another application is the correction of false disparities near object edges or partly-occluded areas. In this case, we can't use the normal filter because the disparity value $I(p)$ of mismatch pixel p is invalid. So, we replace $G_{\sigma_r}(|I_p - I_q|)$ by $G'_{\sigma'_r}(|I'_p - I'_q|)$ where I'_p is the RGB value of the corresponding colour image and σ'_r the RGB range parameter. This is called joint-Bilateral filtering [16]. The idea is to use only disparities from neighbouring pixels that are no mismatches and where the pixel colours are similar to

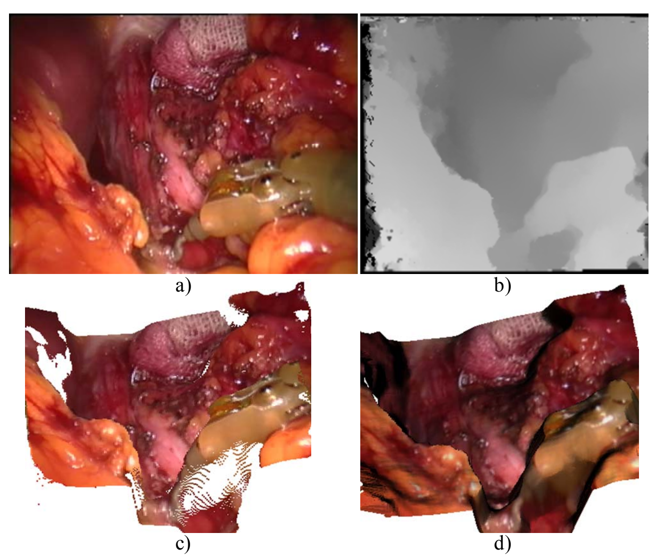


Fig. 3. a) Image of an intervention with the daVinci® system; b) disparity image; c) reconstructed point cloud; d) surface mesh

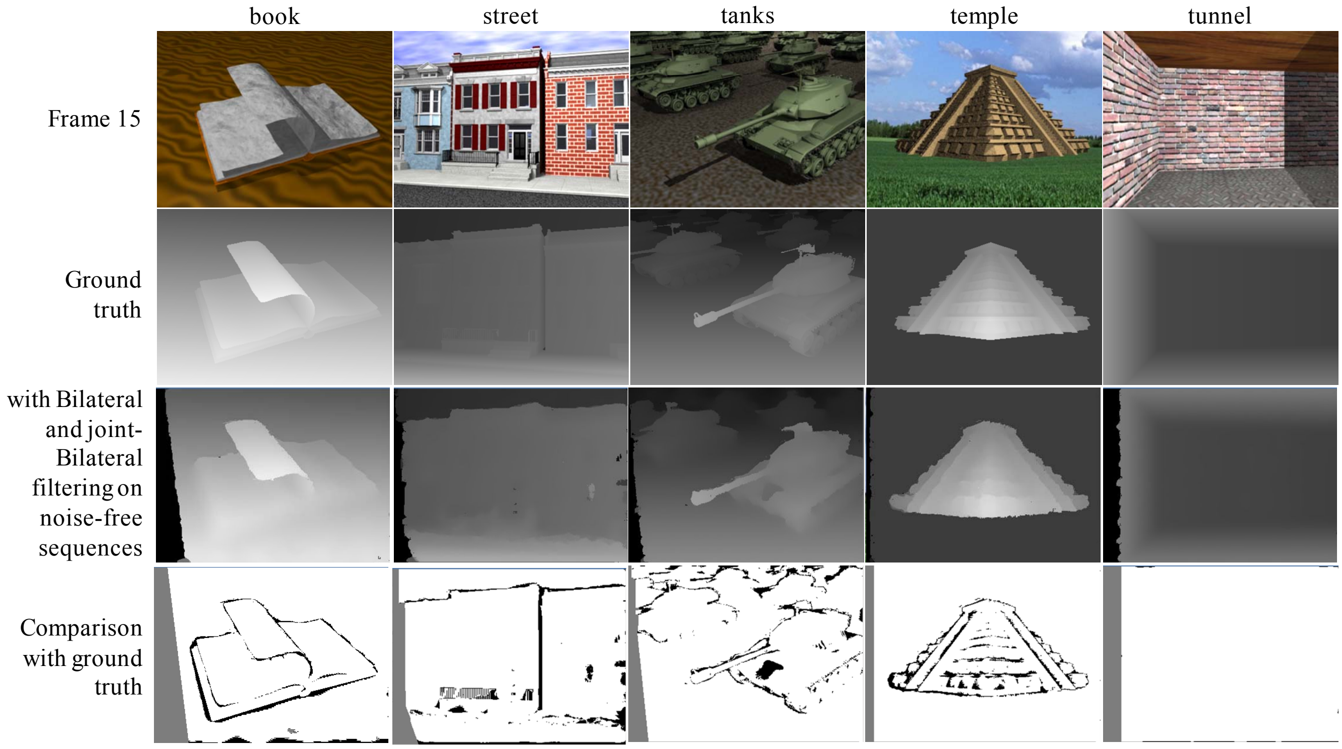


Fig. 4: Selected frames of the synthetic sequences and comparison between ground truth disparities and the version with Bilateral filter

the colours of p. In this way, we can propagate correct disparities into partly-occluded areas near object edges without blurring the disparities at the edges.

A CPU and a GPU version of the Bilateral and joint-Bilateral Filter is available. The naïve CPU implementation is too slow, so we use an approximation that is presented by Paris et al [17]. The GPU version which exploits the inherent parallelism of the algorithm can calculate the exact values within milliseconds.

D. 3D reconstruction and point cloud postprocessing

The corrected and smoothed disparity image is used to calculate the 3D coordinates of the image points. This is realized by a standard triangulation method on the GPU. Here, the same mismatch detection as in the consistency step is used and mismatches are not reconstructed. Additionally, 3D points that violate the ordering constraint from the disparity image are prevented from reconstruction.

We want to further reduce the noise in the point cloud by applying the method presented by Hoppe et al [18]. Here, for each point p a tangent plane H is calculated from its neighborhood N using the least-squares method. N is implicitly known from the neighbors of a point in the disparity image. Its normal n_H serves as approximation of the real normal n_p of p. The only important parameter is the size of N.

Given k points $q_i \in$ N centered on p, the 3x3 covariance matrix M of N is created by

$$M = \frac{1}{k}\sum_{i=1}^{k} q_i q_i^T - \bar{q}\bar{q}^T, \qquad \bar{q} = \frac{1}{k}\sum_{i=1}^{k} q_i.$$

The normal n_H is the eigenvector corresponding to the smallest eigenvalue of M. Since M is 3x3, symmetric, and positive semi-definite, its eigenvalues can be directly computed and are always positive. To further smooth the point cloud, the calculated normal of each point is used to project it into the tangent plane.

The least squares normal calculation is easy to parallelize and, therefore, implemented on the GPU.

It is also possible to quickly generate a surface mesh from the point cloud. For meshing we exploit the implicit ordering of the pixels in the image. The neighbours of a pixel are directly known without explicitly calculating them. When connecting points to triangles, we only have to ensure that the 3D points don't violate the ordering constraint, meaning the neighbours in the image and the reconstructed point cloud are different. Finally, the resulting point cloud or mesh is rendered using standard methods from VTK.

III. Results

The surface reconstruction has to be evaluated in terms of accuracy, robustness and speed for endoscopic and robot images as well as synthetic data. Main emphasis is the postprocessing using Bilateral filtering considering quality of the disparity images and surfaces.

A. Evaluation with Synthetic Image Sequences

We tested the accuracy of the algorithm and its robustness to noise using the synthetic stereo sequences from [9] (Fig. 4). As in their paper, the images were altered adding zero-centred Gaussian noise to all colour channels. The most important parameters can be seen in Table II.

Table I. Synthetic image sequences with added noise $\boldsymbol{\sigma}$: Comparison between four versions (with Bilateral smoothing, without Bilateral smoothing, with Bilateral smoothing and without temporal information, with Bilateral smoothing and joint Bilateral disparity correction) and the temporal DCB grid algorithm from [9]. Measured is the percentage of bad pixels with disparity difference to ground truth > 1

Algorithm	σ	book		street		tanks		temple		tunnel	
		mean	stdev	mean	stdev	mean	stdev	mean	stdev	mean	stdev
standard Bil. smoothing	0	5.1	1.1	6.2	**0.7**	6.7	1.3	8.8	1.2	1.3	1.3
No Bilateral smoothing		8.2	1.5	7.9	0.9	7.4	1.5	10.6	1.4	1.5	1.3
No temporal info		5.8	1.3	7.7	1.3	6.9	1.4	10.7	1.9	1.8	1.2
Bil. and joint Bil. filter.		**4.9**	1.0	**5.6**	**0.7**	**6.0**	1.1	**7.5**	1.0	**1.0**	0.9
With Bilateral smoothing	10	21.0	4.3	13.2	1.7	16.1	3.8	22.2	3.0	6.6	4.4
No Bilateral smoothing		27.5	4.9	16.7	1.9	20.2	4.1	27.9	3.2	10.4	4.2
No temporal info		25.8	4.5	19.6	2.4	16.6	4.0	32.2	3.8	8.5	5.3
Bil. and joint Bil. filter.		**20.2**	4.1	**12.5**	**1.7**	**14.9**	3.6	**18.2**	2.8	**6.1**	4.0
With Bilateral smoothing	20	38.5	7.1	22.6	2.5	38.0	10.7	40.7	4.4	17.8	7.3
No Bilateral smoothing		44.8	7.7	24.5	2.7	40.8	9.5	44.2	4.7	22.3	6.3
No temporal info		51.2	8.5	29.3	3.4	42.5	9.5	50.0	5.3	18.3	7.9
Bil. and joint Bil. filter.		**37.6**	7.0	**19.9**	2.4	35.4	10.5	38.5	4.4	**17.3**	7.5
Temp. DCB Grid		44.0	2.0	25.9	2.0	**31.4**	6.1	**31.7**	1.8	36.4	7.9

For three noise levels with different variance ($\sigma_1 = 0; \sigma_2 = 10; \sigma_3 = 20$) we compared four different versions of our method with their best performing method (Table I and fig. 5) to evaluate the impact of temporal information, Bilateral smoothing and joint-Bilateral disparity correction.

Overall, the method shows good results in noise-free and noisy sequences. Especially the tunnel sequence works well. Here, the recursive structure of the algorithm is beneficial for its robustness. Problems arise with the tanks and temple sequence especially on noisy images. Here, the temporal information can sometimes worsen the result. It introduces false disparities from the last time step in the actual calculation, which due to the noise, are harder to discriminate from the correct disparity candidate. This is especially problematic in homogeneous regions. The application of the Bilateral filter always significantly improves the result. The joint Bilateral disparity correction is especially useful in the street, tanks and temple sequence that have a lot of object edges. In comparison, our method seems to be more robust to noise in three of the five sequences than the best algorithm from [9], but it is hard to directly compare the methods only based on the results given in the paper. The tunnel sequence is significantly better which implies the conclusion that our method is especially robust when the disparity gradually changes.

Each image sequence has a resolution of 400x300. We evaluated the running time on an Intel Xeon X5450 CPU

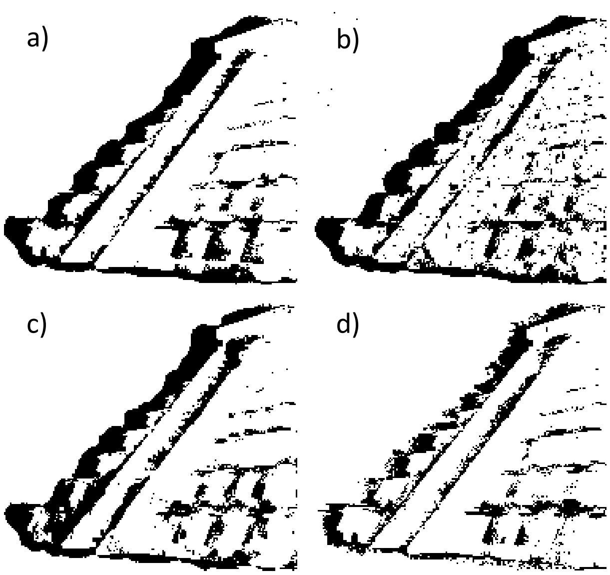
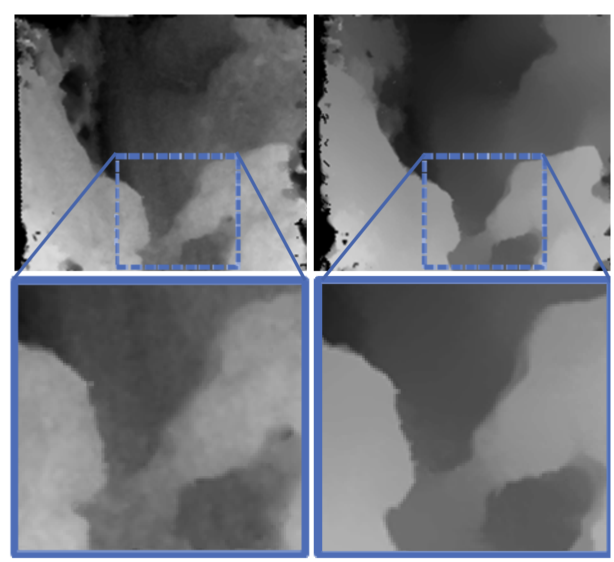


Fig. 5. Section of temple frame 15, comparison with ground truth a) version with Bilateral smoothing; b) without Bilateral smoothing; c) without temporal information; d) with additional joint Bilateral filtering

Table II. Parameterization of the evaluation with synthetic image sequences

Parameter	value
Hybrid Recursive Matching block size	9x9
Correspondence correction threshold	0.5
Bilateral filter neighbourhood size	17x17
Bilateral filter spatial parameter σ_s	8
Bilateral filter range parameter σ_r	1.5
Joint Bilateral filter range parameter σ'_r	0.5



a) b)

Fig. 6. a) Disparity image and enlarged segment from Fig. 2 without Bilateral filtering b) with Bilateral filtering

with 16 GB RAM and an Nvidia GTX 280 graphic card. On this machine, the surface reconstruction with Bilateral filtering and correction achieved about 11-13 fps for noise-free sequences and about 7-9 fps for noisy images ($\sigma = 20$). The reason for this performance loss is mainly the more expensive correspondence correction due to more mismatches.

B. *Stereo Endoscopic Images*

When using images from *da Vinci®*, the Bilateral filter has to be configured depending on the image resolution. Using a resolution of 320x240, the filter range parameter σ_r is set to 0.75. In this way, only neighbor pixels with disparities in a subpixel distance have a decisive impact on the smoothing. If the resolution is doubled, the disparity interval is doubled. So, we also doubled σ_r to have a smoothing effect comparable to images with half resolution. The least square neighbourhood size is set to 7x7, a tradeoff between noise and detail reduction.

The correspondence analysis produces smooth disparity images. Most mismatches arise at the image margin, an area that is less important for surface registration. Artefacts mainly arise when the images are filled with smoke caused by tissue ablation or suffer from interlacing effects due to fast movements of objects. The algorithm has also problems in regions with very fine structures and near discontinuities where one or both regions show almost no texture. Here, disparities from one region tend to flow out into the other region. This behavior is caused by the recursive structure of the algorithm. But in most parts, the propagation of disparity values from high-texture into low-texture areas is beneficial. However, mismatches are corrected in the next time steps as the algorithm optimizes its disparity image over time. Although the baseline between the cameras is small, the points in the point cloud are smoothly distributed and no staircase effect can be seen. Further results concerning robustness and accuracy can be found in [14].

The speed was evaluated using the daVinci® sequence from Fig. 1 which has a resolution of 320x240. We tested it on the same machine we used for the synthetic sequences. In this case, the combined CPU-GPU surface reconstruction without Bilateral smoothing and disparity correction (without visualization of the result) achieved about 40 fps. With the application of the filters, it achieved about 33 fps. Most of the time is consumed by the correspondence analysis.

We also qualitatively evaluated the influence of the Bilateral filter and the least squares calculation on the smoothness of the point cloud. Without Bilateral filtering, the subpixel disparity image resulting from the Hybrid Recursive Matching is quite noisy. After its application, the disparity image looks smoother whereas edges are preserved (Fig. 6). When also smoothing the point cloud using the least square normals, the resulting clouds are almost noise-free with structures still recognizable (Fig. 7).

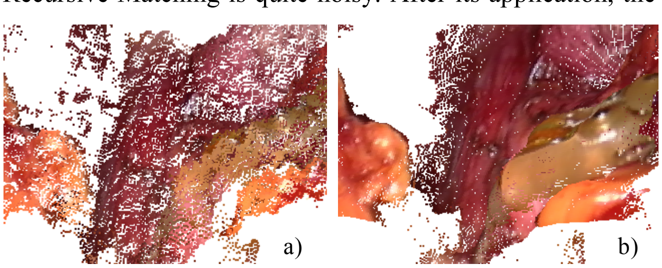


Fig. 7. a) Noisy point cloud from Fig. 2; b) point cloud after Bilateral filtering and least squares smoothing

C. *Stereo Images from ARMAR III*

When reconstructing the surface from ARMAR III images, we are confronted with different problems. The baseline of the cameras is wider which makes subpixel calculation less important. We also have more separated objects that only cover a small section of the image. This leads to more discontinuities and occlusions. Additionally, the camera is moved while there is little change in the scene when detecting objects.

Here, the same parameterization is used as in the daVinci sequences. As we use a higher image resolution of 640x480, the intensity range parameter σ_r of the Bilateral filter is resized to 1.5. Here, the least squares smoothing has a larger impact on noise reduction.

The objects have to be detected only in near real-time. So, we can process each image pair several times in order to get better results. The first temporal disparity candidate is only partially useful when the camera was strongly moved, but after several iterations, the disparity image converges.

In regions occluded in the other image, the disparity calculation still produces erroneous values. While this deteriorates the quality of the disparity image, it has less impact on the reconstructed point cloud. This is due to filtering of disparity mismatches during 3D reconstruction which makes the method more robust to occlusions.

Qualitative results of the surface mesh calculation and smoothing for an enlarged region of Fig. 3 can be seen in Fig. 8. Here, the calculation was done twice for each image pair. These images show that small details like the pear stem is partially reconstructed while the general shape is preserved.

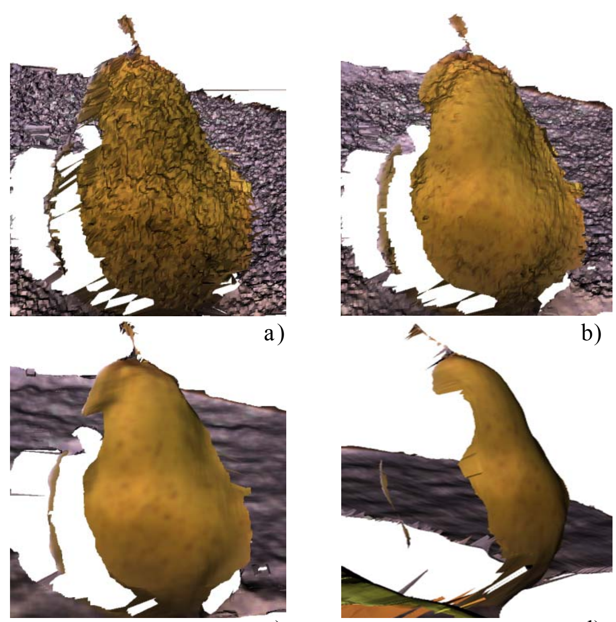


Fig. 8. Enlarged surface meshes from pear of Fig. 3. a) Mesh without Smoothing; b) with Bilateral filtering; c) with additional least squares smoothing; d) other perspective of c)

IV. Conclusion

In this paper, we presented an approach to create a surface model from stereo camera image sequences. Although it was first developed for endoscopic images and is optimized for this use (real-time calculation, temporal information, subpixel disparity calculation and smoothing), it was successfully adapted to a humanoid robot.

For low-resolution images, the approach works in real-time. When using higher resolutions, the limiting factors are the correspondence analysis and the consistency check which both run on the CPU.

In our evaluation, we proved the usefulness of temporal information for disparity calculation on image sequences. It makes the disparity image more robust to image noise. To improve the result, the correspondence analysis and refinement can be applied several times to the same images.

We also showed that the Bilateral filter can be used to considerably reduce the noise in the disparity image. The disparity calculation on the synthetic data performed considerably better. The reconstructed point clouds were much smoother without losing too many details compared to the point clouds reconstructed without filtering. An additional application of a joint Bilateral filter to correct disparity mismatches can effectively be used to improve the quality of the disparity images especially in partly occluded regions. In conclusion, Bilateral filtering is an efficient way to improve disparity images, especially when subpixel accuracy is needed. Although it can be combined with an arbitrary correspondence analysis method, it amplifies the usefulness of temporal information in particular, as can be seen in the evaluation using the synthetic data.

The least squares smoothing was especially useful for the images of the robot. Here, the extracted normals are not only used for smoothing, but can serve as additional features to discriminate objects [5].

Overall, the method produces accurate results on low-resolution images, especially when concerning its short execution time. These results are good enough to be used in both our applications. The method is especially robust to noise or low-textured regions.

Future work includes the full transfer of the approach on the GPU. Here, the recursive nature of the correspondence analysis is the biggest challenge. Another emphasis lies on further improvement of the algorithm near discontinuities and occlusion where disparities tend to flow out in neighboring structureless regions. While this effect can be reduced using the algorithm several times, it is still not satisfactory. We also have to be able to separate objects from the background. In medical surface registration, the points belonging to instruments have to be removed from the point cloud whereas in object recognition, only points belonging to these objects may be used. We also have to implement efficient solutions to merge surface models from different time steps.

Additionally, we need synthetic data where we can extract virtual images from a 3D model, reconstruct and compare it to the original. This can serve as a great supplement to the data already used to evaluate the approach.

V. References

[1] G. Guthart and J. Salisbury, "The intuitive tm telesurgery system: Overview and application," in *IEEE International Conference on Robotics and Automation*, vol. 1, pp. 618–621, San Francisco, CA. USA, 2000.

[2] D. Stoyanov, A. Darzi, and G. Yang, "A practical approach towards accurate dense 3d depth recovery for robotic laparoscopic surgery," *Computer Aided Surgery*, vol. 4, pp. 199–208, Juli 2005.

[3] F. Devernay, F. Mourgues, and E. Coste-Maniere, "Towards endoscopic augmented reality for robotically assisted minimally invasive cardiac surgery," in *Proceedings of Medical Imaging and Augmented Reality*, 2001.

[4] B. Vagvolgyi, L. Su, R. Taylor, and G. Hager, "Video to ct registration for image overlay on solid organs," *Proc. Augmented Reality in Medical Imaging and Augmented Reality in Computer-Aided Surgery (AMIARCS)*, pp. 78–86, 2008.

[5] D. Gonzalez-Aguirre, J. Hoch, S. Roehl, T. Asfour, E. Bayro-Corrochano, and R. Dillmann, "Towards shape-based visual object categorization for humanoid robots," in *IEEE International Conference on Robotics and Automation - ICRA*, 2011.

[6] T. Asfour, K. Regenstein, P. Azad, J. Schroder, A. Bierbaum, N. Vahrenkamp, and R. Dillmann, "Armar-iii: An integrated humanoid platform for sensory-motor control," in *2006 6th IEEE-RAS International Conference on Humanoid Robots*, pp. 169–175, 2006.

[7] R. Osada, T. Funkhouser, B. Chazelle, and D. Dobkin, "Shape distributions," *ACM Transactions on Graphics (TOG)*, vol. 21, no. 4, pp. 807–832, 2002.

[8] A. Johnson, "Spin-images: a representation for 3-d surface matching," tech. rep., Carnegie Mellon University, 1997.

[9] C. Richardt, D. Orr, I. Davies, A. Criminisi, N. Dodgson, and A. Criminisi, "Real-time spatiotemporal stereo matching using the dual-cross-bilateral grid," *Computer Vision–ECCV 2010*, pp. 510–523, 2010.

[10] J. Davis, D. Nehab, R. Ramamoorthi, and S. Rusinkiewicz, "Spacetime stereo: A unifying framework for depth from triangulation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 296–302, 2005.

[11] C. Leung, B. Appleton, B. Lovell, and C. Sun, "An energy minimisation approach to stereo-temporal dense reconstruction," in *Proc. Int. Conf. on Pattern Recognition*, Citeseer, 2004.

[12] P. Azad, T. Gockel, and R. Dillmann, *Computer Vision: Principles and Practice*. Elektor-Verlag, 2007.

[13] N. Atzpadin, P. Kauff, and O. Schreer, "Stereo analysis by hybrid recursive matching for real-time immersive video conferencing," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 14, March 2004.

[14] S. Röhl, S. Bodenstedt, S. Suwelack, H. Kenngott, B. Müller-Stich, R. Dillmann, and S. Speidel, "Real-time surface reconstruction from stereo endoscopic images for intraoperative registration," *SPIE Medical Imaging*, 2011.

[15] S. Paris, P. Kornprobst, J. Tumblin, and F. Durand, "A gentle introduction to bilateral filtering and its applications," in *ACM SIGGRAPH 2007 courses*, p. 1, ACM, 2007.

[16] S. Mattoccia, S. Giardino, and A. Gambini, "Accurate and efficient cost aggregation strategy for stereo correspondence based on approximated joint bilateral filtering," *Computer Vision–ACCV 2009*, pp. 371–380, 2010.

[17] S. Paris and F. Durand, "A fast approximation of the bilateral filter using a signal processing approach," *International journal of computer vision*, vol. 81, no. 1, pp. 24–52, 2009.

[18] H. Hoppe, T. DeRose, T. Duchamp, J. McDonald, and W. Stuetzle, "Surface reconstruction from unorganized points," *Computer graphics-Ney York-Association for computing machinery*, vol. 26, pp. 71–71, 1992.