[6] D.D. Morris and T. Kanade, "A Unified Factorization Algorithm for Points, Line Segments and Planes with Uncertainty," *Proc. Int'l Conf. Computer Vision,* pp. 696–702, 1998.

[7] M. Irani and P. Anandan, "Factorization with Uncertainty," *European Conf. Computer Vision,* June 2000.

[8] Z. Sun, V. Ramesh, and A. Murat Tekalp, "Error Characterization of the Factorization Method," *Computer Vision and Image Understanding,* vol. 82, pp. 110–137, May 2001.

[9] T.J. Broida and R. Chellappa, "Estimating the Kinematics and Structure of a Rigid Object from a Sequence of Monocular Images," *IEEE Trans. Pattern Analysis Machine Intelligence,* vol. 13, pp. 497–513, June 1991.

[10] P.A. Beardsley, A. Zisserman, and D.W. Murray, "Sequential Updating of Projective and Affine Structure from Motion," *Int'l J. Computer Vision,* vol. 23, pp. 235–259, June 1997.

[11] A. Azarbayejani and A. Pentland, "Recursive Estimation of Motion Structure and Focal Length," *IEEE Trans. Pattern Analysis Machine Intelligence,* vol. 17, pp. 562–575, June 1995.

[12] J. Weng, N. Ahuja, and T.S. Huang, "Optimal Motion and Structure Estimation," *IEEE Trans. Pattern Analysis Machine Intelligence,* vol. 15, pp. 864–884, Sept. 1993.

[13] R. Szeliski and S. Kang, "Recovering 3D Shape and Motion from Image Streams Using Nonlinear Least Squares," *J. Visual Comm. and Image Representation,* vol. 5, pp. 10–28, Mar. 1994.

[14] B. Triggs, P. McLauchlan, R. Hartley, and A. Fitzgibbon, "Bundle Adjustment—A Modern Synthesis," *Vision Algorithms: Theory and Practice,* vol. 1883, pp. 298–372, 2000.

[15] T.S. Huang and A.N. Netravali, "Motion and Structure from Feature Correspondences: A Review," *Proc. IEEE,* vol. 82, pp. 252–268, Feb. 1994.

[16] P.E. Debevec, C.J. Taylor, and J. Malik, "Modeling and Rendering Architecture from Photographs: A Hybrid Geometry- and Image-Based Approach," *Proc. ACM SIGGRAPH Conf.,* pp. 11–20, Aug. 1996.

[17] D. Liebowitz, A. Criminisi, and A. Zisserman, "Creating Architectural Models from Images," *Proc. EuroGraphics Conf.,* vol. 18, pp. 39–50, Sept. 1999.

[18] R. Szeliski and P. Torr, "Geometrically Constrained Structure from Motion: Points on Planes," *Proc. European Workshop 3D Structure from Multiple Images of Large Scale Environments,* pp. 171–186, June 1998.

[19] H. Stark and Y. Yang, *Vector Space Projections: A Numerical Approach to Signal and Image Processing, Neural Nets, and Optics.* John Wiley & Sons, 1998.

[20] D. Youla and H. Webb, "Image Restoration by the Method of Convex Projections: Part 1—Theory," *IEEE Trans. Medical Imaging,* vol. 1, pp. 81–94, Oct. 1982.

[21] M.I. Sezan and H. Stark, "Image Restoration by the Method of Convex Projections: Part 2—Application and Numerical Results," *IEEE Trans. Medical Imaging,* vol. 1, pp. 95–101, Oct. 1982.

Reconstruction of Planar Surfaces Behind Occlusions in Range Images

Fabio Dell'Acqua, *Member*, *IEEE*, and Robert Fisher

Abstract—Analysis and reconstruction of range images usually focuses on complex objects completely contained in the field of view; little attention has been devoted so far to the reconstruction of simply shaped wide areas like parts of a wall hidden behind furniture pieces in an indoor range image. The work presented in this paper is aimed at such reconstruction. First of all, the range image is partitioned based on depth discontinuities and fold edges. Next, the planes best fitting each of the regions constituting the partition of the image are determined. A third step locates potentially contiguous surfaces, while a final step reconstructs the hidden regions. This paper presents results for reconstruction of the shape of planar surfaces behind arbitrary occluding surfaces. The system proved to be effective and the reconstructed surfaces appear to be reasonable. Some examples of results are presented from the Bornholm church range images.

Index Terms—Image processing, occlusion, range data analysis, range image partition, range data reconstruction.

1 INTRODUCTION

RANGE images are used in a wide range of applications. So far, they have been extensively used in object recognition [10], [15], [8], [9], reverse engineering [6], and other applications, nearly all focused on small and rather complex objects. While extending the use of range images to a whole environment rather than well-delimited objects (an important example of these applications is the CAMERA EU project [5]), new issues arose. Occlusion is a major cause of information loss: With even moderately complicated objects, it is virtually impossible or impractical to obtain complete range scans. Still, an exhaustive description of the observed object or environment is needed for some applications, like construction of a 3D model [5]. An alternative way of filling in the gaps, at least partially, without performing extra scans, is to infer the layout of objects in the occluded area by exploiting information from the surroundings. This procedure is termed reconstruction. Reconstruction of simple-shaped wide regions occluded by objects located closer to the sensor is an important point for this procedure. In this paper, we propose a method for reconstructing plane surfaces behind furniture pieces or other closer objects. This is a new problem on which little previous work [7] has been done. In this paper, Section 2 presents the data we used for our experiments, Section 3 presents the theory, Section 4 presents the experiments and discussion, while Section 5 draws some conclusions.

- *F. Dell'Acqua is with Laboratorio di Comunicazioni Elettriche, Dipartimento di Elettronica, Università di Pavia, Via Ferrata, 1, I-27100 Pavia, Italy. E-mail: fabio.dellacqua@ele.unipv.it.*
- *R. Fisher is the Division of Informatics, the University of Edinburgh, 5 Forrest Hill, Edinburgh EH1 2QL, Scotland, UK. E-mail: rbf@dai.ed.ac.uk.*

Manuscript received 221Sept. 2000; revised 11 May 2001; accepted 30 May 2001.
Recommended for acceptance by Y.-F. Wang.
For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number 112897.

0162-8828/02/$17.00 © 2002 IEEE

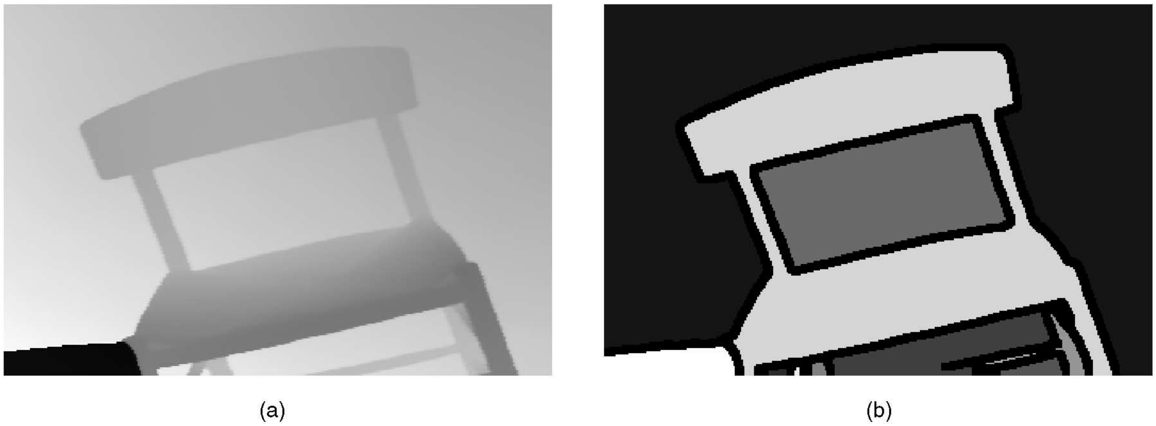

Fig. 1. (a) A range image and (b) corresponding extracted regions after labeling.

2 The Data

Range information can be obtained by a variety of methods [1]. The data we used was sensed by a K2T 3D laser scanner on 360 degrees azimuth, 63 degrees elevation, both with a 0.045 degrees step, thus resulting in $8,000 \times 1,400$ pixel images. For each pixel, both range and IR intensity values are registered. Range precision is around one millimeter; range values are converted into x,y,z values in a Cartesian system centered on the sensor.

The huge dimensions of the raw range data files (hundreds of MB) pushed us to work on small subimages, i.e., rectangular subsets of pixels.

In Fig. 1a, a subimage extracted from a range image is shown. Note that a planar surface is clearly visible lying behind the chair and a part of the surface is hidden by the chair itself. The aim of our work is to reconstruct such missing data.

3 Theory of Occluded Surface Recovery

The key to reconstruction is to identify contiguous surface regions potentially connected behind closer occluding surfaces (identified by depth discontinuities and closer distances). Hypothetical surfaces can then be created to connect or fill in the contiguous surfaces behind the occluding surfaces.

3.1 Planar Image Region Segmentation

Preprocessing with a median filter [13] reduced the 4.62 percent outlying pixels. The next step is to partition the image into different surface regions that may belong to the same surface, although separated by an occluding object. Separation between adjacent regions can be marked [11] either by a depth discontinuity (e.g., a boundary of an occluding object) or by a fold edge, which is by definition an area where the local surface normal changes suddenly (e.g., where two walls meet). Other cases have not been considered [19]. To reduce the complexity of the general problem of segmenting a range image [2], [14], [18], this was split into two stages. The first part finds depth discontinuities; then, a further partition based on fold edges is performed. These two procedures are described in the following.

3.1.1 Depth Discontinuity Detection

Depth discontinuities in closely sampled range images are unexpectedly smooth, probably due to the aperture of the laser beam. In our data, the shortest across-discontinuity path usually averages between three and eight pixels. Thus, a simple threshold on gradient is unsuitable for partitioning the image and we chose to use a more refined Canny edge extractor. First, we build three 2D images out of the x, y, z coordinates for each pixel. Then, we apply the Canny edge extractor to each of those images and combine the results by means of a three-input "OR" operation applied to each pixel. As a discontinuity cannot be parallel to all three of the axes, wherever a gap in the coordinates is present, it will be detected in at least one of the images. The final "OR" operation fuses all the partial results together. Call σ the standard deviation of noise on point coordinates, which turned out to be around 7mm. Then, lower and higher Canny thresholds were set to about $\frac{1}{2}\sigma$ (3 mm) and 3σ (20 mm), respectively. This is to make false activation extremely unlikely ($\mathrm{noise} > 3\sigma$). The next step is a double dilation on this edge image, aimed at two purposes. The first is to include in the "edge region" (pixels set to 1, ignored in the computation of region features) all the pixels around the edge, likely to be corrupted by the "edge smoothing effect" described above. The second purpose is to help bridge possible, though infrequent, small gaps in the image partition. At the end of this stage, the binary image partitions the original image into regions. Each region is extracted and labeled. An example result is shown in Fig. 1b.

3.1.2 Fold Region Selection

Some of the regions extracted using the procedure described in Section 3.1.1 may contain fold edges. In this section, we discuss how to detect them. The best fitting plane is computed for each region (the method used is described in Section 3.2) and the mean square distance σ_d of the points from the best fitting plane is considered. The more the shape of the region departs from a plane, the higher the resulting σ_d. A threshold of 15mm has experimentally been found suitable to discriminate regions possibly containing fold edges (higher σ_d) from the nearly-flat ones (lower σ_d). Only the former undergo the fold edge extraction procedure described in the following.

3.1.3 Fold Edge Detection

Fold edges in range images are difficult to detect due to the high level of noise which prevents reliable determination of a local surface normal vector. A number of methods have been proposed for estimating surface normals, but none outperforms the others, and the choice of the method is not trivial [12]. We considered the methods presented in the overview contained in [12] and combined the features of the various methods in order to address the problems we encountered, in particular the high noise level. To get an acceptable estimate of the surface normal, it is necessary to renounce high resolution and operate on wide-area averages. The x,y,z image with outliers removed is first low-pass filtered to remove more noise and then partitioned into a grid of 5×5 pixel patches. On every patch, the plane best fitting its points is determined. Then, two patches are considered matching when the

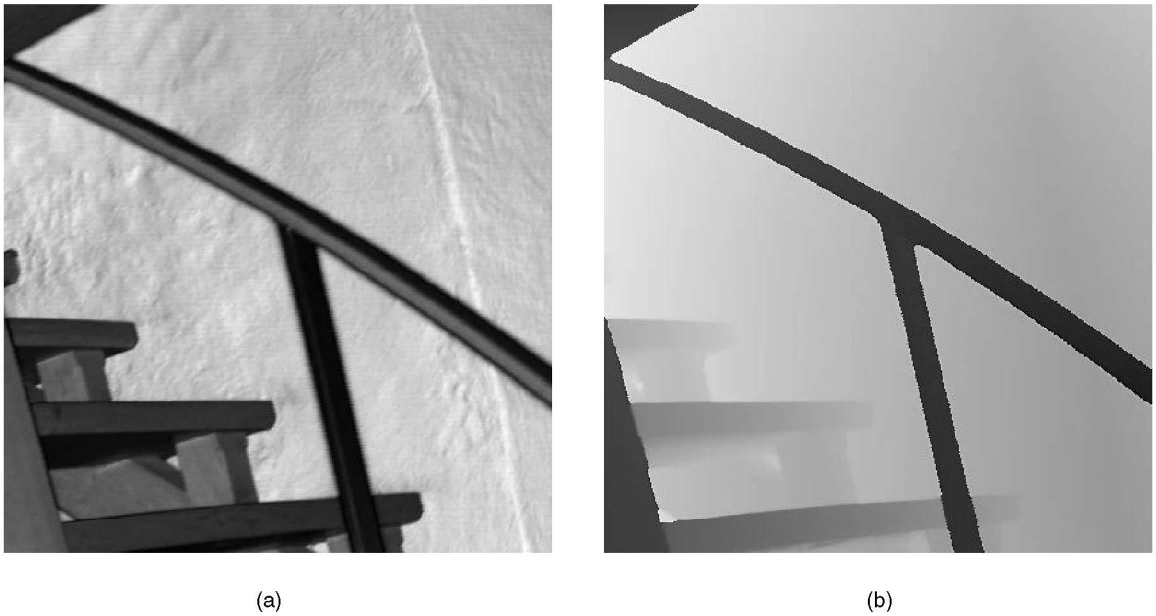

Fig. 2. (a) An intensity image and (b) its corresponding range image.

cosine of the angle between the two normals is higher than 0.9 (26 degrees) and the difference between their distances from the origin is lower than

$$0.2\mathrm{m}(20\sigma, \sigma = \text{standard deviation of noise values}).$$

Both values were determined by experiment. For every patch, the number of matching patches are counted. After that, the patch with the highest number of compatible patches makes the first partition; the procedure is then repeated using only the remaining patches, again picking the largest set of matching patches, and so on; the procedure stops when no new set bigger than 20 patches can be formed; smaller sets generally produce unreliable best fitting planes. Any remaining patches are marked as nonusable. Isolated patches and smaller nonconnected components are then suppressed to regularize results. Eventually, results are combined with the previous depth-discontinuity partition to give final image partition. To illustrate the effectiveness of this technique, consider the sample subimage in Fig. 2. It contains a large fold edge on the wall corner at the right, as well as several smaller ones on the steps. Note the roughness of the wall surface which, together with the high level of high-frequency noise, makes extraction of local normal vectors unreliable.

Fig. 3 shows the image partition. The image on the left displays the partition based on simple depth discontinuity detection. The image on the right shows image partition after fold edge detection with the two sides of the wall separated, as well as some of the many planar surfaces on the steps.

3.2 Region Description

Once the image has been partitioned, for every region, the best fitting plane is determined by means of a least squares approach. The least squares method provides a three-dimensional unit normal vector $\vec{n}$ (directed toward the observer) and a scalar representing the distance of the plane from the axes origin. These two parameters are recorded and associated with the analyzed plane.

3.3 Region Reconnection

The search for possibly contiguous, occluded regions is next. Two regions are deemed possibly contiguous if the scalar product between their normals is higher than 0.92 (corresponding to normals lying less than 23 degrees apart) and the difference between their displacements with respect to the origin is less than 0.25 m (around 35 times σ of noise). Within an image, all the possible region pairs are compared and organized into groups. Groups are such that each region within a group matches with each other region within the same group, according to the definition given above. The constraints on normal orientation and plane distance from the origin are necessary for two regions to be reconnected. However, they are not sufficient, as two separated areas lying approximately on the same plane can be separated by an object that is further than themselves; in this case, no occlusion of the plane is taking place in the area in between and reconstruction makes no sense. An example may be a niche separating two parts of the same wall.

Thus, we need to test the connecting regions between to decide whether occlusion has occurred. We reconstruct potentially occluded areas by a pairwise merging of visible regions across occluded regions and we test the hypothesised occlusion by investigating the area between the boundaries of the two regions. The process starts with the two lists of the boundary pixels from each region in the pair under test. Then, each possible pair of boundary pixels, one from each list, is considered. The Bresenham algorithm [4] is used to generate pixels on the path from the first region boundary to the second region boundary. By interpolating the depths of the endpoint pixels, a depth is associated with each pixel on the path. Each interpolated depth is compared with the actual sensed depth. If the interpolated depth is closer than the observed depth, it is evidence against the possibility of occlusion and the event is recorded. To take into account the high noise level, we had to allow a margin of 4σ on the observed point distance from the plane best fitting the surface. More precisely, the pixel is considered occluded if:

$$d_n < d_1 + \frac{(d_2 - d_1)}{N} \cdot n + 4\sigma, \qquad (1)$$

where d_1 and d_2 are the depths of the line endpoints; N is the number of pixels separating the endpoints, σ is the standard deviation of actual point distances from the best fitting plane, and d_n is the actual range data for the pixel currently being considered,

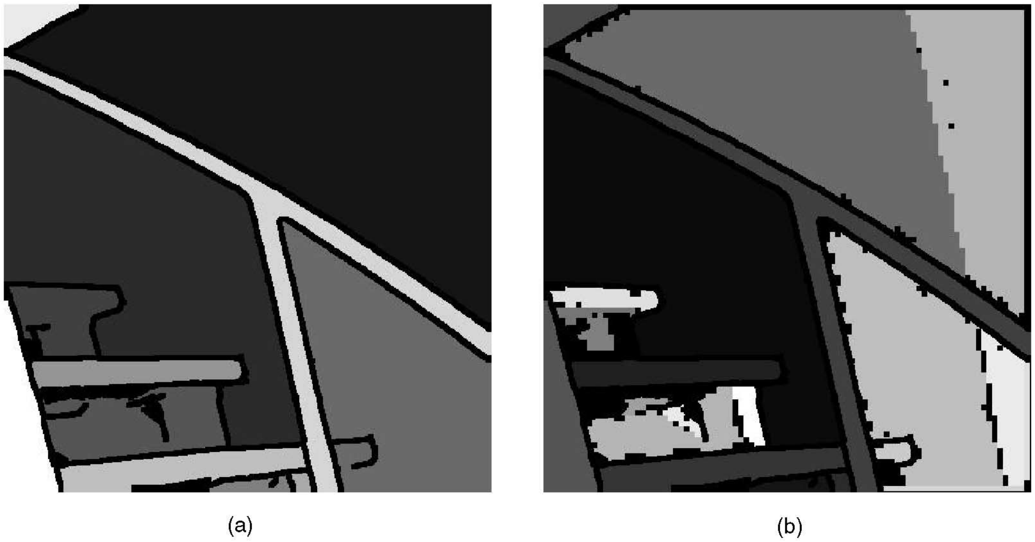

Fig. 3. The image in Fig. 2 after some processing. (a) Partitioned using only depth discontinuities. (b) Additionally partitioned using also fold edges.

which is the *n*th of N. A line is classified as occluded if none of its points have evidence against occlusion. This means that the pixel set covered by the line has a high probability of belonging to an object generating an occlusion. This may be better understood by looking at the scheme in Fig. 4.

The dashed line on the left of the picture represents an occluded surface: From the observer, the sensed surface (solid line on top) is closer than the line interpolating the surrounding regions. The dashed line on the right is nonoccluded, as interpolation between surrounding regions is closer than the observed surface (solid line on the bottom).

As the number of boundary pixels is generally on the order of hundreds or thousands, the number of possible endpoint pairs can be huge and a selection step is needed to keep processing time within reasonable limits. A first elimination criterion relies on the fact that it is useless to test lines crossing the considered regions themselves. More ways to reduce the computational effort are to consider only the shorter lines from one region to another and to limit the number of lines starting from each point.

After the connecting lines have been tested, the number of occluded lines (lines reporting points closer than the local threshold) is tested for consistency with the hypothesis of occlusion. In theory, a single occluded line might connect two surfaces, but this would require a very thin connection and is thus unlikely. Thus, two regions are considered to be separated by an occluding object if the number of occluded lines exceeds 10.

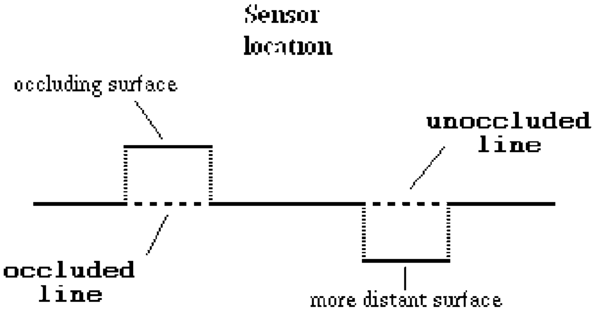

Fig. 4. Occluded and unoccluded surface patches. The solid line represents the observed surface, while the dashed lines represent the test paths.

3.4 Reconstruction

Reconstruction takes place between regions belonging to the same group and satisfying the depth constraint. Groups may contain more than two regions but, for the sake of simplicity, an explanation is first given for a single pair of regions and then extended to the general case. Given two matching regions, reconstruction only takes place on pixels belonging to lines satisfying the occlusion path constraint described in Section 3.3. This is to ensure that reconstruction will not:

1. involve areas that do not satisfy the maximum depth constraint and
2. extend to areas too far from the nearest known data.

As any reconstruction requires hypothesising nonobservable data, these assumptions lead to a conservative reconstruction having a high likelihood of being correct. While testing lines, a map is generated out of all the pixels belonging to accepted lines; these pixels undergo reconstruction. An example of such map, still using the example image in Fig. 1, is shown in Fig. 5a. The white region contains occluded lines.

Where all the tests have given a positive result, reconstruction is performed. Every pixel in the reconstructed area is assigned a new depth by interpolating the depths at the end points of the shortest line passing through the pixel and joining two boundary pixels in matching regions. As a result, the occluding object seems to have been removed from the scene, as visible in the corrected range image in Fig. 5b.

However, further analysis using a cosine shaded image like in Fig. 6a can give a better idea of the situation. The reconstructed area is indeed still distinguishable from the true wall, even though it is not so evident. This is due to wall texture, which is missing in the reconstructed area; simple interpolation cannot reproduce it satisfactorily, especially for large areas.

We make the reconstructed area more homogeneous with the original regions by estimating the noise level in them and adding a similar amount of noise to the reconstructed area. The improvement is visible in Fig. 6b.

Still, if only a rough reconstruction of the area behind occlusion is required, the algorithm does a good job of filling gaps. This can be noted in Fig. 7, where 3D representations of the scene are displayed, before and after reconstruction.

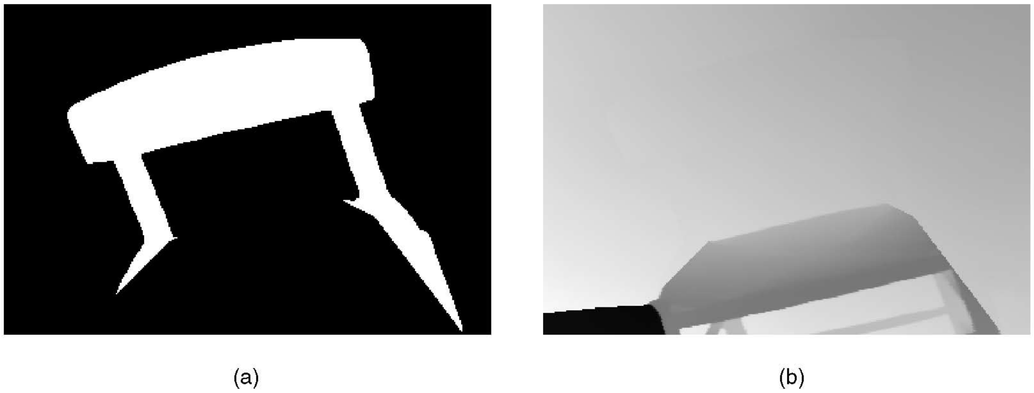

Fig. 5. (a) Map of occluded lines, in white in a black backgrond. (b) Range image after occluding points have been replaced by estimated occluded points (compare with Fig. 1a).

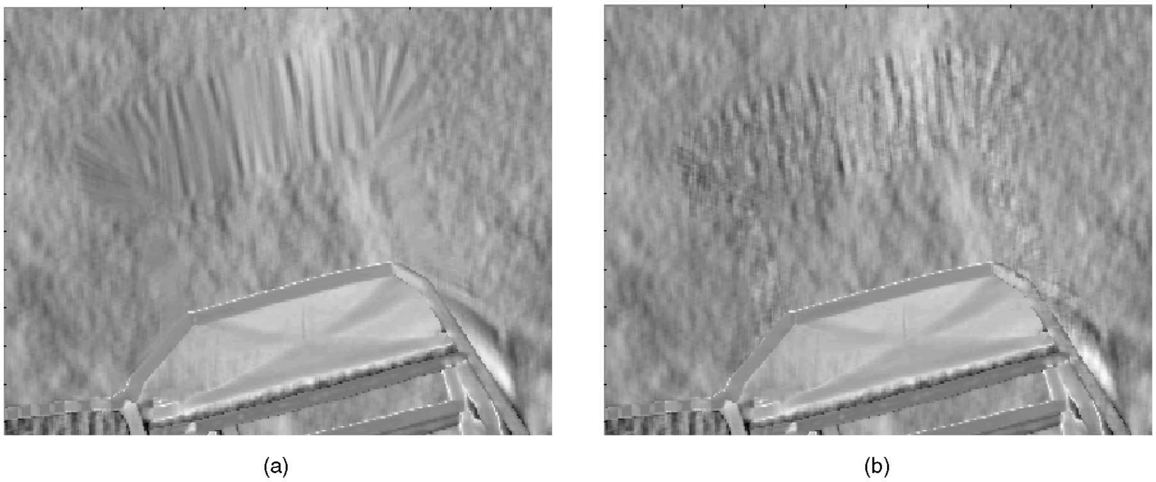

Fig. 6. (a) The cosine shade image for the reconstructed version of the subimage in Fig. 1. (b) Same as above, with noise added.

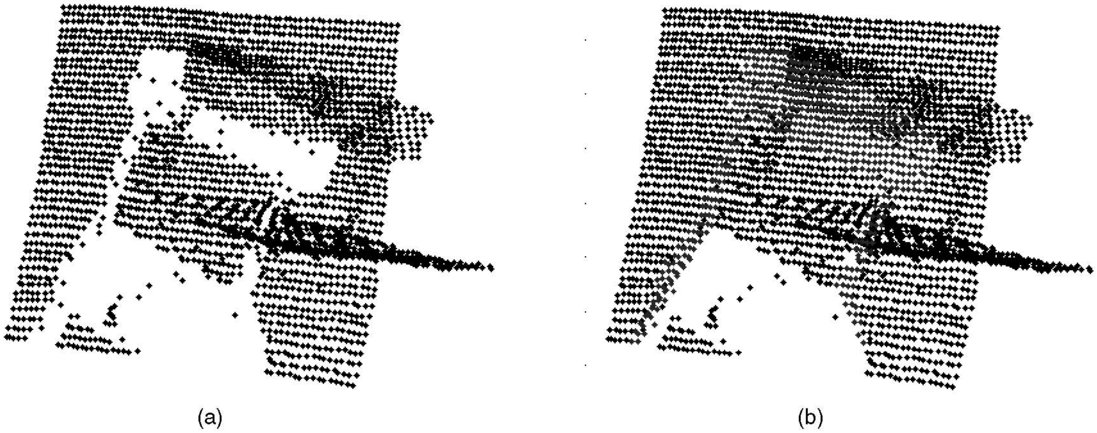

Fig. 7. (a) Original x,y,z image and (b) the same image with the reconstructed pixels added.

4 Some Results

The algorithms described in Section 3 were implemented in Matlab and run without compilation on a Sun Ultra 10 machine. The process takes about half an hour per image on average, though duration is dependent on image complexity. The largest contributions to total processing time comes from fold edge extraction and planar approximation. A substantial improvement is expected from recoding the process and compiling the scripts, possibly resulting in cutting the time per image to less than one minute.

We used the Bornholm Church range data for more experiments. From the nine original indoor range images, a set of 19 small subimages were extracted. The sizes of subimages ranged from 231×151 to 751×501 pixels. The depth discontinuity method (Section 3.1.1) first partitioned the subimages into 90 regions. Then, the fold edge method (Section 3.1.3) further partitioned the 90 regions into a total of 160 regions, an average of 9.26 per subimage. The least partitioned subimage contained only two regions, while the most partitioned one had 22 regions.

TABLE 1
Confusion Matrix for Region Pair Matching, Before Occlusion Test

	Matching pairs	Non-matching pairs
Declared potentially matching	41	120
Declared non-matching	5	780

TABLE 2
Confusion Matrix for Region Pair Matching, Final Results

	Matching pairs	Non-matching pairs
Declared matching	41	37
Declared non-matching	5	863

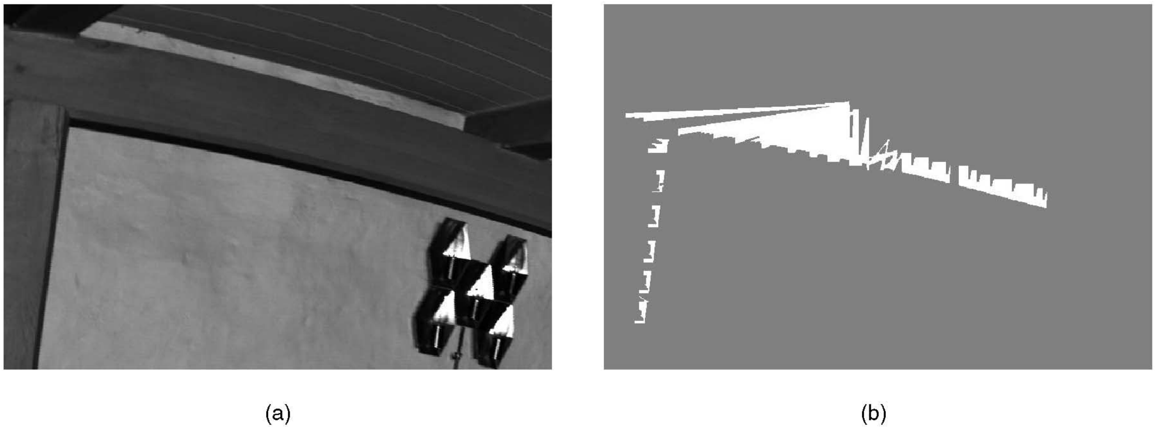

Fig. 8. (a) An example subimage and (b) the corresponding set of reconstructed points, in white on black background.

In two subimages, the fold edge method increased the number of regions from 1 to 10. Such subimages depict niches with no depth discontinuities where segmentation can only rely on fold edges. In three subimages, the fold edge method caused no further partition as they actually contained no fold edges.

Upon completion of subimage partition, a human observer marked matching regions (“ground truth”), exploiting the clear visual information contained in corresponding intensity images. Within the set of 946 possible region pair combinations, only 46 pairs are actually matching.

Note the imbalance between false matching (12.7 percent) and missed matching (0.53 percent) rates. We chose to keep coplanarity requirements loose to minimize missed matching, relying instead on the occlusion test (Section 3.3) to reduce the final rate of false matching.

The five missed matching cases were inspected (see Table 1). The regions involved are small (< 500 pixels) and have an elongated shape. Thus, border effects corrupt results to a wider extent than with regular data; moreover, the slope of best fitting plane is more easily biased by noise.

Based on reciprocal matching, the 97 regions involved in the 161 matching pairs were collected into 31 groups containing an average of 3.12 regions each. Of the 161 pairs, only 78 passed the occlusion test and were thus deemed suitable for reconstruction. Table 2 shows the final classifications.

Inspection of the images showed that the 41 correctly declared matching regions (52.6 percent) were actually suitable for reconstruction, while the remaining 37 were not. This latter figure seems quite high, but please note:

1. Included are the cases of “useless” reconstruction, e.g., where partition is due to noise. Correct plus useless reconstruction score 46 (59.5 percent) together.
2. In any case, reconstruction only takes place where occlusion is possible and, in most cases, this is sufficient to reduce to a minimum the effect of a wrong choice. The number of invalid pixels generally ranges from tens to a few hundreds.

A nullified incorrect matching (see point 2 above) is visible in the following example. In Fig. 8a, the intensity image for one of the subimages is shown.

The fold edge partition created two regions on the faces nearest to the observer of the two dark beams parallel to the wall, due to noise in between. One region includes the vertical beam, while the other region includes the rightmost part of the horizontal beam. Coplanarity of the faces made this pair pass the first test. The second test (minimum number of occluded connecting lines) also passed because of a group of lines running from the top of the vertical beam to the nearest part of the horizontal beam, on the flat surface at the crossing of the two beams. The two regions were thus deemed suitable for reconnection. Additional tests could remove some of these cases.

However, thanks to local occlusion test, reconstruction took place to a negligible extent, mostly flattening small bumps on the beams. The wall was untouched, as seen in the reconstruction map in Fig. 8b where the white area is the set of reconstructed points. Indeed, replacing real data points with points from the best fitting plane meant little change or no change at all. Original and reconstructed 3D point sets are nearly undistinguishable.

Still, some remarks are to be made on the cause of false matching. A brief investigation revealed that the main reasons for such errors were:

1. random matching of small regions whose parameters are heavily biased by the noise and
2. forced partition of concave curved regions into planar patches. The partition of a curved region into planar patches creates a range of slopes and depths available for matching, while concavity provides surrounding regions satisfying the occlusion constraint on test paths. A typical example is the back of a chair against the wall, which accounts for the majority of the false matching cases reported. The result of such a false matching is a surface that connects a portion of the chair back to the wall.

Where applicable, results of reconstruction are hard to judge quantitatively because no truth is available for comparison. From a qualitative point of view, results seem reasonable, as they appear correct just like in the example shown in Fig. 7.

5 Conclusions and Future Work

A method for analyzing range images, locating occlusions of large homogeneous areas, and reconstructing these areas behind occluding objects has been presented. Some results obtained from the research have also been shown. From the work done so far, some remarks can be made. Partition by depth discontinuities and reconstruction did not cause any particular problems. The most difficult tasks are to partition the image based on fold edges and to make a correct choice of the regions to reconstruct. In both cases, a major disturbance factor is the high level of noise encountered in range images; fold edge identification requires strong smoothing of the data and various devices to get around the consequent information loss. Noise is also a cause of error in region matching, as its influence on the parameters computed for small regions is overwhelming.

Decisions about region matching and the choice of areas to be reconstructed are currently based on low-level information, e.g., the location of a point with respect to the plane best fitting a region and the large number of odd cases encountered in range images would be better dealt with using a higher-level approach, such as recognition of observed objects. Progress in this direction may come from availability and use of knowledge on the range image content, though this is not straightforward. Other future work on this topic may be as follows:

1. A wavelet analysis of the visible area surrounding the reconstructed occlusion area may allow a more sensible reconstruction, taking into account structures like bumps that would otherwise be smoothed by depth interpolation.
2. The reconstruction could be made to work on more general shapes in addition to planes. Many walls in old buildings are cylindrical and even more complicated shapes can be found. Of course, matching parts of the shapes could cause higher complexity levels and probably some knowledge about the environment should also be integrated. Preliminary results on this extension are presented in [17].
3. One could also develop a reliability gauge for the reconstructed pixel, as not all the pixels can be reconstructed with the same confidence.

Finally, one might argue that the solution to the occlusion problem is to simply acquire additional images. Even if it is possible to return the the identical scene and to place the sensor in a location for scanning the missing area, recent results by Sanchiz [16] show that even simple scenes can require hundreds of images to obtain complete, high quality range data. Thus, it may be preferable to reconstruct small regions of missing data instead of attempting to observe them.

Acknowledgments

The authors wish to acknowledge the SMART2 EC TMR network for funding this research.

References

[1] P. Besl, "Active, Optical Imaging Sensors," *Machine Vision and Applications,* pp. 127-152, 1988.

[2] P.J. Besl and R.C. Jain, "Segmentation Through Variable-Order Surface Fitting," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 10, no. 2, pp. 167-192, Mar. 1988.

[3] B. Bhanu and L.A. Nuttall, "Clustering Based Recognition of Occluded Objects," *Proc. Int'l Conf. Pattern Recognition (ICPR '86),* pp. 732-734, 1986.

[4] J. Bresenham, "Incremental Line Compaction," *The Computer J.,* vol. 25, no. 1, pp. 116-120, 1982.

[5] CAMERA EC TMR network home page [Online]. Available: http://www.dai.ed.ac.uk/daidb/people/homes/rbf/CAMERA/camera.htm.

[6] D.W. Eggert, A.W. Fitzgibbon, and R.B. Fisher, "Simultaneous Registration of Multiple Range Views for Use in Reverse Engineering of CAD Models," *Computer Vision and Image Understanding,* vol. 69, no. 3, pp. 253-272, Mar. 1998.

[7] R.B. Fisher, *From Surfaces to Objects.* J. Wiley & Sons, 1989.

[8] R.B. Fisher, A.W. Fitzgibbon, M. Waite, M. Orr, and E. Trucco, "Recognition of Complex 3-D Objects from Range Data," *Proc. CIAP 93,* pp. 509-606, 1993.

[9] Y.K. Ham and R.H. Park, "3D Object Recognition in Range Images Using Hidden Markov Models and Neural Networks," *Pattern Recognition,* vol. 32, no. 5, pp. 729-742, May 1999.

[10] R.L. Hoffman, "Object Recognition from Range Images," PhD thesis, Dept. of Computer Science, Michigan State Univ., 1986.

[11] R. Hoffmann and A.K. Jain, "Segmentation and Classification of Range Images," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 9, no. 5, pp. 608-620, Sept. 1987.

[12] A. Hoover, G. Jean-Baptiste, X. Jiang, P.J. Flynn, H. Bunke, D. Goldgof, K. Bowyer, D. Eggert, A. Fitzgibbon, and R. Fisher, "An Experimental Comparison of Range Segmentation Algorithms," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol 18, no. 7, pp. 673-689, July 1996.

[13] S.L. Hurt and A. Rosenfeld, "Noise Reduction in Three-Dimensional Digital Images," *Pattern Recognition,* vol. 17, no. 4, pp. 407-421, 1984.

[14] X.Y. Jiang, H. Bunke, and U. Meier, "High-Level Feature Based Range Image Segmentation," *Image and Vision Computing,* vol. 18, no. 10, pp. 817-822, July 2000.

[15] P.G. Mulgahonkar, C.K. Cowan, and J. DeCurtins, "Understanding Object Configurations Using Range Images," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 14, no. 2, pp. 303-307, Feb. 1992.

[16] J.M. Sanchiz and R.B. Fisher, "Environment Recovery by Range Scanning with a Next-Best-View Algorithm," *Robotica,* to be published.

[17] F. Stulp, F. Dell'Acqua, and R. B. Fisher, "Reconstruction of Surfaces Behind Occlusions in Range Images," *Proc. Third Int'l Conf. 3-D Digital Imaging and Modeling (3DIM01),* June 2001.

[18] E. Trucco and R.B. Fisher, "Experiments in Curvature-Based Segmentation of Range Data," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 17, no. 2, pp. 177-182, Feb. 1995.

[19] M.A. Wani and B.G. Batchelor, "Edge-Region-Based Segmentation of Range Images," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 16, no. 3, pp. 314-319, Mar. 1994.