International Society for Photogrammetry and Remote Sensing
Société Internationale de Photogrammétrie et de Télédétection
Internationale Gesellschaft für Photogrammetrie und Fernerkundung

THE INTERNATIONAL ARCHIVES OF THE PHOTOGRAMMETRY, REMOTE SENSING AND SPATIAL INFORMATION SCIENCES
ARCHIVES INTERNATIONALES DE PHOTOGRAMMÉTRIE, DE TÉLÉDÉTECTION ET DE SCIENCES DE L'INFORMATION SPATIALE
INTERNATIONALES ARCHIV FÜR PHOTOGRAMMETRIE, FERNERKUNDUNG UND RAUMBEZOGENE INFORMATIONSWISSENSCHAFTEN

VOLUME
VOLUME
BAND XXXVI

PART
TOME
TEIL 3 / W36

ISPRS Workshop

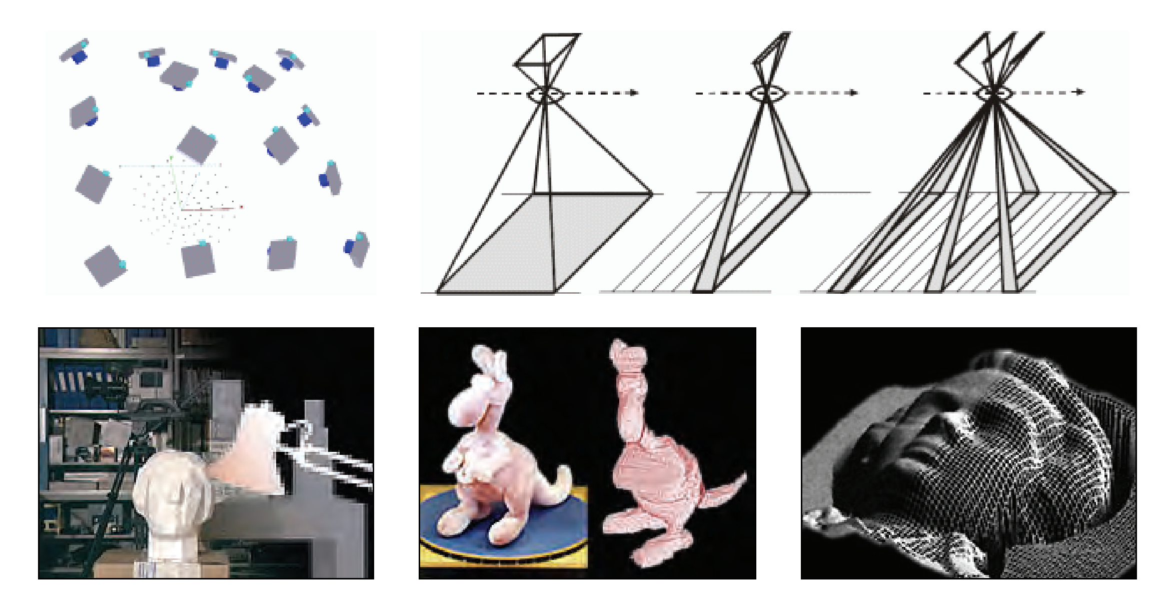
in conjuction with ICCV 2005

BenCOS

Towards Benchmarking Automated Calibration, Orientation and Surface Reconstruction from Images

Beijing, China

October 15, 2005



Editors

O. Hellwich, I. Niini, C. Ressl, V. Rodehorst, D. Scharstein, P. Sturm

Organisers

ISPRS WG III/1 - Automatic Calibration and Orientation of Optical Cameras
ISPRS WG III/2 - Surface Reconstruction

ISSN 1682-1777

Published by
ISPRS Working Groups
- WG III/1 – Automatic Calibration and Orientation of Optical Cameras
- WG III/2 – Surface Reconstruction

ISPRS Headquarters 2004-2008
c/o ORHAN ALTAN, ISPRS Secretary General
Division of Photogrammetry, Faculty of Civil Engineering
Istanbul Technical University
Ayazaga, 34469 Istanbul, Turkey
Phone: +90 212 285 38 10
FAX: +90 212 285 65 87
Email: oaltan@itu.edu.tr
ISPRS WEB Homepage: http://www.isprs.org

Available from
GITC bv
P.O.Box 112
8530 AC Lemmer
The Netherlands
Tel: +31 (0) 514 56 18 54
Fax: +31 (0) 514 56 38 98
E-mail: mailbox@gitc.nl
Website: www.gitc.nl

TABLE OF CONTENTS

Workshop Organising Committee

Olaf Hellwich, Berlin University of Technology, Germany
Ilkka Niini, Oy Mapvision Ltd., Finland
Camillo Ressl, Vienna University of Technology, Austria
Daniel Scharstein, Middlebury College, USA
Peter Sturm, INRIA, France

Program Committee

Niclas Börlin, Sweden
Yuri Boykov, Canada
Andrew Fitzgibbon, UK
Wolfgang Förstner, Germany
Armin Grün, Switzerland
Henrik Haggrén, Finland
Richard Hartley, Australia
Janne Heikkilä, Finland
Christian Heipke, Germany
Karsten Jacobsen, Germany
Karl Kraus, Austria
Kyros Kutulakos, Canada
Yi Ma, USA
Hans-Gerd Maas, Germany
Helmut Mayer, Germany
David Nistér, USA
Marc Pollefeys, USA
Richard Szeliski, USA
Bernhard Wrobel, Germany
Ramin Zabih, USA

INTRODUCTION

The workshop *Towards Benchmarking Automated Calibration, Orientation and Surface Reconstruction from Images* (BenCOS) of the *International Society for Photogrammetry and Remote Sensing* (ISPRS) focuses on automatic methods for surface reconstruction from images, multi-view stereo, camera (self-) calibration, motion estimation and related topics.

One major aim of the new Commission III on *Photogrammetric Computer Vision and Image Analysis* is to bring together researchers from the related fields, and let them benefit from mutual experience. The Working Groups *Automatic Calibration and Orientation of Optical Cameras* and *Surface Reconstruction* are co-chaired by researchers from the Computer Vision and Photogrammetry communities.

Apart from being a forum for discussing new scientific results, the major goal of these Working Groups is to establish true benchmarks for the performance evaluation of proposed methods. We believe this is a highly important aspect of scientific research; it allows an objective comparison of different approaches, catalyzes new developments, and eases the access of potential commercial users to these research areas and communities. The motivation for this workshop is thus threefold:

- communication of new scientific results in the related areas,
- bringing together researchers from different communities, and
- working towards the definition of benchmarks.

The workshop proceedings include the ten presented papers. Reviewing was carried out in a double-blind process by leading international researchers of the Computer Vision and Photogrammetry areas. Each full paper has been reviewed by 3 members of the Program Committee.

We hope that all workshop participants will leave Beijing with the most rewarding memories in the scientific, technical and social aspects, and that those unable to attend will find the proceedings a valuable source of information.

Olaf Hellwich
Ilkka Niini
Camillo Ressl
Volker Rodehorst
Daniel Scharstein
Peter Sturm

SESSION 1

ORIENTATION AND CALIBRATION

AUTOMATIC IMAGE SEQUENCE REGISTRATION BASED ON A LINEAR SOLUTION AND SCALE INVARIANT KEYPOINT MATCHING

Z. Shragai, S. Barnea, S. Filin, G. Zalmanson, Y. Doytsher

Department of Transportation and Geo-Information, Technion – Israel Institute of Technology, Israel

{zivs, barneas, filin, zalmanson, doytsher}@technion.ac.il

Commission III/1

KEY WORDS: Aerial Triangulation, Linear solutions, Scale invariant feature transform, Essential matrix, image correspondence

ABSTRACT:

Automatic registration of image sequences has been a subject of research for many years, both in the photogrammetric and computer vision communities. As part of the automation, linear orientation methods are used to obtain approximations for a subsequent bundle adjustment solution. Linear solutions can be at time "too general" particularly in a sense that they mostly employ uncalibrated cameras, a fact leading to severely unstable results in most photogrammetric problems such as the case for the direct linear transformation (DLT) in a nearly flat terrain. Furthermore, to the best of our knowledge, none of them handle more than two or three images simultaneously without imposing several theoretical constraints that cannot be guaranteed in practical imaging missions. In this paper a sub-optimal linear solution for the exterior orientation parameters of image sequences is developed. The proposed method is demonstrated on an aerial image strip. The paper shows that the method successfully generates reliable and accurate approximations both for the orientation parameters as well as for tie point coordinates. For an automatic extraction of the latter, the Scale Invariant Feature Transform (SIFT) algorithm is applied.

1. INTRODUCTION

It is commonly accepted both in photogrammetry and computer vision communities that bundle adjustment is a "golden standard" method for recovering exterior orientation parameters from image sequences (Hartley et al., 2001). A bundle adjustment process requires, however, good initial values for all the six exterior parameter, as well as approximations for the 3D coordinates of the tie points. To avoid the need for approximations, a great deal of effort has been put on developing general algorithms that provide linear solutions to a variety of orientation problems (see e.g., Hartley et al.,2001; Rother and Carlsson, 2001; Carlsson and Weinshall, 1998). Many of them address a general problem in which the entire set of camera intrinsic (calibration) and extrinsic parameters is unknown. These solutions are stable and perform successfully only in cases where no limitations on either the acquisition geometry or the underlying object space are present. However, for typical photogrammetric problems these solutions have not yet proven useful. For example, the solutions proposed by Hartley et al. (2001) and Rother and Carlsson (2001) require a reference plane across any two images in a sequence. Carlson-Weinshall duality algorithm (1998) requires a specific number of points in a given number of images. Fitizgibbon and Zisserman (1998) offer the use of the trifocal-tensor in a close or open sequence. The trifocal-tensor does not suit, however, the photogrammetric process because of its requirement for tie points to appear in three sequential images. In the standard photogrammetric process, with 60 percent overlap between images, applying this model will relate to only 20 percent of each image. Furthermore, most of the works do not refer to the global exterior orientation parameters and produce only a relative solution. Pollefeys et. al (2002a) offer a solution that is based on sequentially linking and reconstructing image after image, which is then followed by a bundle adjustment.

In this paper a framework for an automated photogrammetric solution is presented. Our objectives are reducing the operator input to a minimum and eliminating the reliance on initial values for the computation of the exterior orientation parameters. The proposed solution requires neither knowing the order of the images nor their overlapping percentage. The only external information required is the ground control points and their corresponding image points. Solutions that follow a similar line can be found in Nistér et al. (2004) where a sequence of video frames is oriented and in Oliensis (1997) where an iterative solution for weak motion (short baselines) image sequences is presented.

As an outline, our solution detects first tie points in image pairs. For this purpose the SIFT strategy (Lowe, 2004; Lowe 1999) is used as described in Section 2. Following the autonomous extraction of the tie point, comes the geometric computation. The proposed geometric framework is founded on the Essential matrix (Hartley and Zisserman, 2003). The Essential matrix between every image pair is calculated and the five relative orientation parameters are extracted. The geometric concept of the pose estimation and the scene reconstruction are given in Section 3. Section 4 presents experimental results and Section 5 concludes the paper.

2. EXTRACTION OF CORRESPONDING POINTS

The Scale Invariant Feature Transform - SIFT (Lowe, 2004; Lowe 1999) is a methodology for finding corresponding points in a set of images. The method designed to be invariant to scale, rotation, and illumination. Lowe (2004) outlines the methodology as consisting of the following four steps:

1. Scale-space extrema detection – using the difference of Gaussian (DoG), potential interest points are detected.
2. Localization – detected candidate points are being probed further. Keypoints are evaluated by fitting an analytical model (mostly in the form of parabola) to determine their location and scale, and are then tested by a set of conditions. Most of them aim guaranteeing the stability of the selected points.
3. Orientation assignment – orientation is assigned to each keypoint based on the image local gradient. To ensure scale and orientation invariance, a transformation (in the form of rotation and scale) is applied on the image keypoint area.
4. Keypoint descriptor – for each detected keypoint a descriptor, which is invariant to scale, rotation and changes in illumination, is generated. The descriptor is based on orientation histograms in the appropriate scale. Each descriptor consists of 128 values.

With the completion of the keypoint detection (in which descriptors are created) the matching process between images begins. Matching is carried out between the descriptors, so the original image content is not considered here. Generally, for a given keypoint, matching can be carried with respect to all the extracted keypoints from all images. A minimum Euclidian distance between descriptors will then lead to finding the correspondence. However, matching in this exhaustive manner can be computationally expensive (i.e., O(N^2) with N the number of keypoints). Common indexing schemes cannot be applied to improve the search here because of the descriptors dimensionality. However, an indexing paradigm, called Best Bin First (BBF), is proposed by Beis and Lowe, (1997). The BBF algorithm reduces the search to a limited number of the most significant descriptors values and then tries locating the closest neighbor with high probability. Compared to the exhaustive matching, this approach improves the performance by up to two orders of magnitude, while difference between the amount of matched points is small. Our proposed solution follows Schaffalitzky and Zisserman (2002) and Brown and Lowe (2003) where all key points from all images are organized in one K-d tree. Once a set of matching points has been generated, another filtering process is applied. This process is based on the RANSAC algorithm (Fischler and Bolles, 1981). The fundamental matrix of the image pairs is calculated and points that do not satisfy the geometric relation are filtered out as outliers. Based on the matching, the order of images within the image sequence is determined. When applying the SIFT method for aerial images the huge image size may lead to the extraction of numerous keypoints. Excess of information is valuable for redundancy; however, it comes with high computational cost. Experiments show, however, that even downscaling the aerial image resolution satisfying amount of keypoints has been provided. In comparative research presented by Mikolajczk and Schmid (2003) the SIFT method has shown superiority over classical methods for interest point detection and matching.

Figure 1 shows the matched keypoints on an extract of two overlapping aerial images. Generally, the algorithm extracted ~4000 keypoints per image, out of them 339 points were matched with less than 5 pixels offset between corresponding points. 146 keypoints have satisfied the geometric model with less than 1 pixel between corresponding points. It is noted that seven points are needed for computing the Fundamental matrix. Experiments on different images with different characteristics (e.g., vegetation, urban scenes) exhibited similar results.

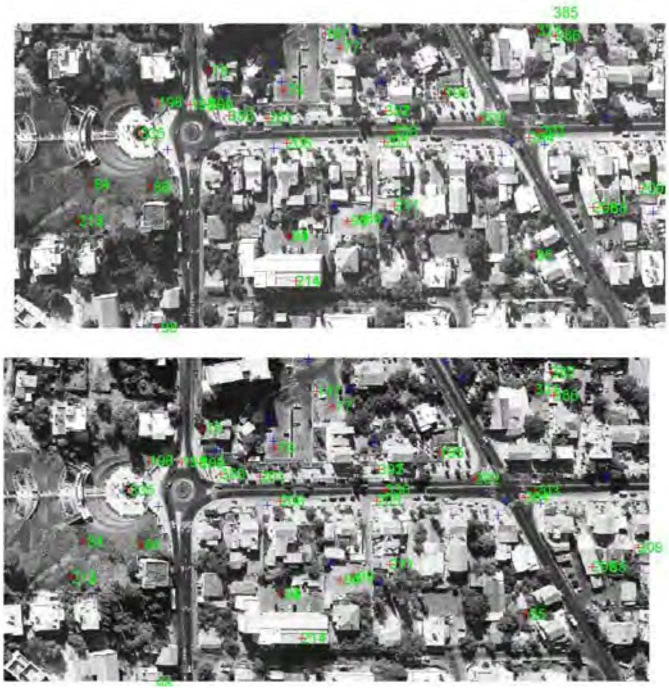


Figure 1. Matched keypoints in an aerial image pair extract

3. THE GEOMETRIC FRAMEWORK

The input for the geometric process is a set of matched points for all overlapping images. In addition, the Ground Control Points (GCPs) and their corresponding image points are provided. The solution considers the intrinsic parameters to be known. The process consists of two main steps: first is finding the relative orientation between all image pairs in the sequence. The second is a simultaneous computation of a transformation that takes into account the relative orientations and optionally the control points. This step is performed linearly as a single optimization process.

3.1 Relative Orientation

The first step is the linear computation of the Essential matrix for each of the overlapping image pairs. The minimum number of required tie points ranges between five (Nistér, 2004; Philip, 1996) to seven (Hartley, 1997).

Extraction of the rotation and translation parameters from the Essential matrix can be carried out as proposed by Hartley and Zisserman (2003). We begin with a singular value decomposing of the Essential matrix: $E=UDV^T$ where U and V are chosen such that det(U)>0 and det(V)>0. Assuming that the first camera matrix is P = [I | 0], the second camera matrix can be one of four possible choices:

$$P_1' = [UWV^T \mid u_3], P_2' = [UWV^T \mid -u_3],$$
$$P_3' = [UW^TV^T \mid u_3], P_4' = [UW^TV^T \mid -u_3]$$

with u_3 the third column of *U*, and

$$W = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

A reconstructed point X will be in front of both cameras only in one of the four possible solutions. Thus, testing with a single point to determine if it is in front of both cameras is sufficient for the choice between the four possible solutions of *P*' (Hartley and Zisserman, 2003). To fine-tune the relative orientation parameters, a non-linear geometric optimization can now take place.

An important issue to account for is the degeneracy of the Essential matrix which arises in the following cases (Torr et al., 1999):

1. All points and camera centers laying on quadratic surface (e.g., cone, cylinder).
2. There is no translation between the images.
3. All tie points lie on the same plane in object space.

Cases (1) and (2) are also a degeneracy of the bundle adjustment algorithm. Cases (2) and (3) are more common. For these cases there is a simpler geometrical model – the Homography. From a Homography one can retrieve the relative orientation parameters as proposed by (Tsai et al., 1982). To choose between the Essential matrix and the Homography, Torr et al. (1999) proposes a measure they call Geometric Robust Information Criterion (GRIC) that computes scores to the fitness of the geometrical model for a given dataset. This measure is also used by Pollefeys et al. (2002b). An alternative way to avoid the degeneracy as in case (3) is using the five point algorithm (Philip, 1996; Nistér, 2004). However, then a tenth degree polynomial must be solved.

3.2 Global Registration

Following the computation of the relative orientation parameters, we are provided with two camera matrices for each image - one, which is fixed (when the image is the first in the pair) and the other, which is relative (when the image is the second). The first and the last images have only one camera matrix. The task of concatenating the relative orientation parameters into one global model is divided into two subtasks: concatenating rotations and concatenating translations. The first subtask can be described by a recursion formula:

$$R_{i+1} = R_m^{i\mapsto i+1} R_i \quad \text{Where } R_1 = I_{3\times3} \qquad (1)$$

where $R_m^{i\mapsto i+1}$ is the rotation in the *m*-th model between the images *i* and *i*+1. Concatenating the camera centers (translation) in the sequence (the second subtask) is a more complicated process. Here, similarly to the first subtask, there are two translation vectors for each image in the sequence (apart of the first and last) one is fixed (in the origin) and the other is relative. However, in contrast to the rotations, with the translation concatenation all vectors are defined up to a scale factor only. The scale ambiguity of each vector affects the size of the reconstructed scene from each image pair, as Figure 2 demonstrates. In Figure 2, C_1 and C_2 are the camera centers of the first and the second images. C_3 is the actual position of image 3, so the scale of the translation vector t_{23} is correct – the scenes reconstructed from images 1, 2 and images 2, 3 fit. Contrary to C_3, a camera position in C_3' leads to reconstructed scenes that differ in scale. The recursion formula of the translation concatenation should, therefore, have the form of:

$$t_{i+1} = t_i + s_m t_m^{i\mapsto i+1} \quad \text{Where } t_1 = [0,0,0]^T \qquad (2)$$

s_m and t_m are the scale factor and the translation vector of the m-model between images i and i+1.

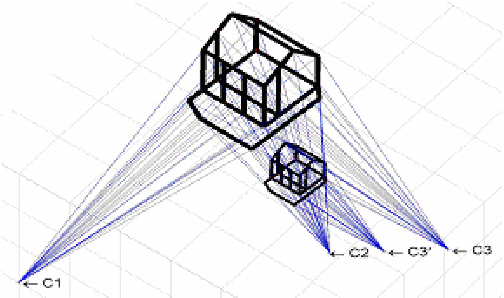


Figure 2. Influence of the translation scale factor on the reconstructed scene.

For solving all the translation scale factors together with the tie point coordinates we now develop a simultaneous and linear solution. The solution is derived from the camera matrix, *P* that fulfills the relation x=*P*X, with X the coordinate vector of a point in object space, and x is the image coordinate vector. Both are given in homogenous coordinates (the last term of X and x is set to 1). *P* may be decomposed into:

$$P = KR[I \mid -t] \qquad (3)$$

with *K* is the camera calibration matrix and *I* a 3x3 identity matrix. By substituting (1) and (2) into (3) a recursion formula for the *P* matrices can be written as

$$P_{i+1} = K \cdot R_m^{i\to i+1} R_i \cdot [I \mid t_i + t_m^{i\to i+1} \cdot s]$$

leading when inserted into the x=*P*X relation to

$$x_{i+1} = K \cdot R_{i+1} \cdot [I \mid t_i + t_m^{i\to i+1} \cdot s] \cdot \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \qquad (4)$$

As *K* and R_i are known $\forall i$, they are of no interest. We, therefore, rewrite Equation (4) as follows

$$\hat{x}_{i+1} = [I \mid \sum_{m=1}^{i} t_m^{i\to i+1}\cdot s_m]\cdot \begin{bmatrix}X\\Y\\Z\\1\end{bmatrix} \qquad (5)$$

with $\hat{x} = (KR)^{-1}x$. Equation (5) provides a linear form for the estimation of the point coordinates, X, Y, and Z, and the scale *s*. Notice that with this model a point is reconstructed from all its instantiations in all images. Each image point contributes two independent equations. There is still one ambiguity left, namely the scale of the first model. This ambiguity is solved by the absolute orientation (into the object space reference frame).

Generally, for each of the components (i.e., tie points and camera matrices) one has to find a similarity transformation, $X_w=H_sX_m$, to the object space reference frame via the GCPs, with H_s of the form:

$$H_s = \begin{bmatrix}R & t\\0^T & \lambda\end{bmatrix} \qquad (6)$$

and λ as the model scale. Linear solutions to this problem have been offered by several authors, e.g., a quaternion based solution (Horn, 1987), orthogonal matrices (Horn et al., 1988) and the Rodriguez matrix (Pozzoli and Mussio, 2003).

An approach that simultaneously integrates the solution for the scale parameters, tie point coordinates and the absolute orientation parameters is now presented. For a control point that appears in an image, it is possible to use equation (7)

$$x = PH_s^{-1}X_W \qquad (7)$$

with *P* as any projection matrix in the model space that acquires the point X_w, and H_s given in Equation (6). In a simultaneous solution, the scale factor λ in H_s can be replaced by the scale factor as given in Equation (2) for the first image pair. H_s becomes now an Euclidian transformation with only six parameters, where λ = 1.

Substituting H_s^{-1} into equation (4) and multiplying both sides by $(KR)^{-1}$ will lead to:

$$\hat{x}_{i+1} = [I \mid \sum_{m=1}^{i} t_m^{i\to i+1}\cdot s_m]\cdot \begin{bmatrix}R^T & \hat{T}\\0^T & 1\end{bmatrix}\cdot \begin{bmatrix}X\\Y\\Z\\1\end{bmatrix}_{World} \qquad (8)$$

with $H_s^{-1}=\begin{bmatrix}R^T & \hat{T}\\0^T & 1\end{bmatrix}$. Equation (8) can be rearranged as:

$$\hat{x}_{i+1} = [R^T \mid \hat{T} + \sum_{m=1}^{i} t_m^{i\to i+1}\cdot s_m]\cdot \begin{bmatrix}X\\Y\\Z\\1\end{bmatrix}_{World} \qquad (9)$$

Equation (9) provides a linear form for the estimation of the scale factors s_m, the global translation $\hat{T}$ and the nine rotation matrix terms. In this representation a 3D affine transformation is solved. This model requires at least four control points. Restricting the solution to a 3D rotation (namely maintaining the orthonormality) can be achieved by using the identity matrix instead of the singular values in the SVD of *R*. Using Equation (5) for tie points and (9) for control points, we are provided with a simultaneous and linear solution. This solution allows having the external effect of control points and the internal constrains of the tie points weighted in simultaneously. Furthermore, control points that appear in only one image can also be taken into account. This solution offers an alternative to the two steps procedure. However, it is noted that it is not optimal in the sense of solving nine parameters explicitly instead of an orthonormal rotation matrix. Experiments with this method yield good results only under specific configurations.

4. EXPERIMENTAL RESULTS

The proposed method is now investigated using synthetic and real data. The sensitivity of the geometric model to additive Gaussian noise is tested first, followed by an application of the process on a strip consisting of four images.

4.1 Synthetic Data

A synthetic configuration that follows typical mapping-mission characteristics was designed with the following parameters, flying altitude, 1700 m, terrain variation ranging between 0-200m, and a focal length of 153 mm. The test set consisted of four images in a sequence with 60 percent overlap. The pitch and roll angles were in the range of ±2^o. For each image pair ~50 tie points were provided. Six ground control points were used. To investigate the sensitivity of the proposed to random errors Gaussian noise with zero mean and standard deviation ranging between 0.0 and 0.3 mm has been added to image coordinates of control and tie points. The maximum standard deviation (0.3 mm) is equivalent to an error of 20 pixels for scanning resolution of 15μ.

Given this input, fundamental matrices were computed and normalized by the known interior camera parameters to form the Essential matrix. Then, a decomposition of the Essential matrix to the rotation and translation components was carried out, followed by up to five (non-linear) iterations to optimize the computed *R* and *t* values. The transformation into a global reference frame was computed using Equations (5) and (6). Rodriguez matrices were used to represent rotations. For each noise level 100 trials were performed. Results were evaluated by three measures: the *std.* of the 3D Euclidean distance between the computed object point coordinates and the actual ones, both for tie and control points (Figure 3), the offsets in the camera positions, again in terms of *std.* of the 3D Euclidean distances (Figure 4) and the angular error of the three camera rotation parameters (Figure 5). Results were compared to bundle adjustment solution, as shown in Figures 3-5. The experiments show that even in the presence of a severe noise reasonable and acceptable solutions can be achieved by the proposed geometric model. Indeed, bundle adjustment solution performs better than the sub-optimal solution, which is of no surprise, but the fact that the results obtained using our method do not fall too far from the optimal solution makes it a good candidate to precede any subsequent optimal solution. Also, the deviations in orientation parameters fairly compare with accuracies obtained with typical GPS/INS systems. Furthermore, under realistic noise level, these results satisfy the requirements of some applications – thus avoiding a subsequent use of bundle adjustment.

4.2 Real Images

An experiment with a strip consisting of four aerial images with flying altitude of 1800 m, and a focal length of 152 mm is now presented. Eight GCPs were available for this image set. The four images are arranged in an L shape form (see Figure 6); their order is not provided as an input. The image coordinates of the GCPs were manually digitized. Tie points were generated using the SIFT procedure. Globally there were ~1000 matched keypoints. About 300 matched points between images with similar orientation (image pairs 1-2 and 3-4), and about 60 matched points for image pair 3-4. Between image triplets about 10 common points were detected.

To evaluate the quality of the two-steps method the orientations were computed first by this procedure only, and then using a bundle adjustment solution. For the bundle adjustment solution the parameters originating from the linear procedure were used as initial approximations. To evaluate the difference between solutions we compare the reconstructed tie point coordinates between the two-steps solution and the bundle adjustment. Results show that the mean distance between the two methods is 0.33 m. However, the accuracy estimate of the points achieved by the bundle adjustment procedure is about ±1 m. This difference is within the uncertainty range of the tie points coordinates. These results are in agreement with those achieved by the synthetic data experiments in Section 4.1 and indicate that the proposed method can be used as an independent solution when achieving high level of accuracy is not a concern and also as an initial values generator for a bundle adjustment solution.

5. SUMMARY AND CONCLUTIONS

Recent years have seen a significant progress made in automation of registration processes. At the same time advances have been made in the field of multi-view geometry. This paper has demonstrated the integration of these two disciplines. No assumptions on the order of the image sequence have been made to execute the proposed linear solution for estimating the camera parameters. Experiments made have demonstrated robustness and stability of the proposed geometric solution even to severe noise levels. Those with real data showed that even with non-standard image configuration a full automation can be achieved.

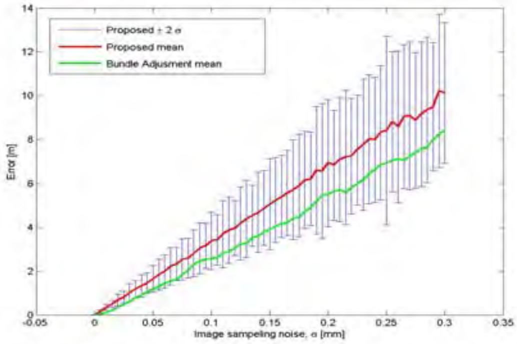


Figure 3. Mean error of the reconstructed points. The X-axis is the noise level in millimeters and the Y-axis represent the ground error (distance) in meters. The error bars represent ± 2σ of the accuracy range as resulted from the trials for each noise level.

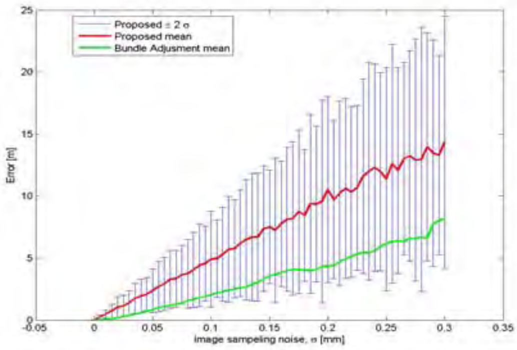


Figure 4. Mean error of the reconstructed image positions parameters. The X-axis is the noise level [mm] and the Y-axis represents the image positions error (distance) [m]. The error bars represent ± 2σ of the accuracy range as resulted from the trials for each noise level.

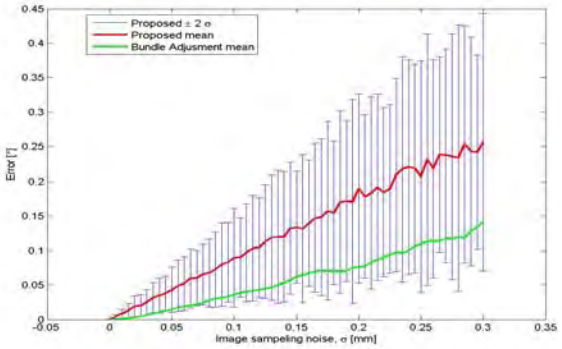


Figure 5. Mean error of the reconstructed camera angles. The X-axis is the noise level in mm and the Y-axis represent the angular error [^{o}]. The error bars represent ± 2σ of the accuracy range as resulted from the trials for each noise level

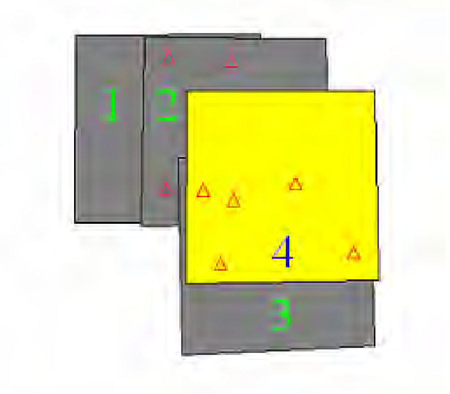


Figure 6. Outline of the aerial image arrangement used for the experiment. Triangles depict control points.

6. REFERENCES

Beis J.S., Lowe D.G., 1997. Shape Indexing Using Approximate Nearest-Neighbour Search in High-Dimensional Spaces. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1000-1006.

Carlsson S., Weinshall D., 1998. Dual Computation of Projective Shape and Camera Positions from Multiple Images, International Journal of Computer Vision 27(3), pp.227–241.

Fischler M.A., Bolles R.C., 1981. Random Sample Consensus: A paradigm for model fitting with application to image analysis and Automated Cartography. *Communication Association and Computing Machine*, 24(6), pp. 381-395.

Fitzgibbon A.W., Zisserman A., 1998. Automatic Camera Recovery for Closed or Open Images Sequences. in Proceedings fifth European Conference on Computer Vision (ICCV), 1998, pp. 311-326.

Hartley R. I., 1997. In defense of the eight-point algorithm. IEEE Transactions on Pattern Analysis and Machine Intelligence, 19, pp. 80-93.

Hartley R., Dano N., Kaucic R., 2001. Plane-based Projective Reconstruction. Proceedings of the Eighth IEEE International Conference on Computer Vision (ICCV), 2001, Vol. 1 pp. 420-427.

Hartley R., Zisserman A., 2003. Multiple View Geometry in Computer Vision. Cambridge University Press, Second Edition.

Horn B., 1987. Closed–form solution of absolute orientation using unit quaternions. *Journal of the Optical Society of America* 4(4) pp.629–642.

Horn B., Hilden H.M., Negahdaripour S., 1988. Closed-form solution of absolute orientation using orthonormal matrices, *Journal of the Optical Society of America* vol. 5, no.7, pp. 1127–1638.

Lowe G. D., 1999. Object Recognition from Local Scale-Invariant Features. Proceedings of the sixth International Conference on Computer Vision (ICCV), Vol. 2, p. 1150

Lowe G. D., 2004. Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision* 60(2) pp. 91–110.

Mikolajczk, K., Schmid C., 2003. A performance evaluation of local descriptors. Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Vol. 2, pp. 257-263.

Nistér, D., 2004. An efficient solution to the five-point relative pose problem. IEEE transaction on pattern analysis and machine intelligence 26(6) pp.756-770.

Nistér D., Naroditsky O., Bergen J., 2004. Visual Odometry, Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), Vol. 1, pp. 652-659.

Philip J., 1996. A non-iterative algorithm for determining all essential matrices corresponding to five point pairs. *Photogrammetric record* 15(88), pp. 589-599.

Pollefeys M., Verbiest F., Van Gool L., 2002a. Surviving dominant planes in uncalibrated structure and motion recovery. Proceedings of the seventh European Conference on Computer Vision (ECCV), Vol. 2, pp. 837-851.

Pollefeys M., Van Gool L., Vergauwen M., Cornelis K., Verbiest F., Tops J., 2002b. Video-to-3D, Proceedings of Photogrammetric Computer Vision (PCV), *International Archive of Photogrammetry and Remote Sensing*, 34(3A), pp. 252-257.

Pozzoli A., Mussio L., 2003. Quick solutions particularly in close range photogrammetry. *International Archives of the Photogrammetry, Remote Sensing*, 34(5/W12) pp. 273-278.

Rother C., Carlsson S., 2001. Linear multi view reconstruction and camera recovery, Proceedings of the Eighth IEEE International Conference on Computer Vision (ICCV), Vol. 1, pp. 42-50.

F. Schaffalitzky and A. Zisserman., 2002. Multi-view matching for unordered image sets, or "How do I organize my holiday snaps?". Proceedings of the 7th European Conference on Computer Vision, pp. 414-431.

Torr P. H. S., Fitzgibbon A. W., Zisserman A., 1999. The Problem of Degeneracy in Structure and Motion Recovery from Uncalibrated Image Sequences, *International Journal of Computer Vision*, Vol. 32 (1), pp. 27-44.

Tsai R. Y., Huang T. S., Zhu W. L. 1982. Estimating three-dimensional motion parameters of a rigid planar patch, ii: Singular value decomposition. IEEE Transactions on *Acoustics, Speech, and Signal Processing*, Vol. 30, pp. 525-534.

ROBUST LEAST-SQUARES ADJUSTMENT BASED ORIENTATION AND AUTO-CALIBRATION OF WIDE-BASELINE IMAGE SEQUENCES

Helmut Mayer
Institute for Photogrammetry and Cartography, Bundeswehr University Munich, D-85577 Neubiberg, Germany
Helmut.Mayer@unibw.de

ABSTRACT

In this paper we propose a strategy for the orientation and auto-calibration of wide-baseline image sequences. Our particular contribution lies in demonstrating, that by means of robust least-squares adjustment in the form of bundle adjustment as well as least-squares matching (LSM), one can obtain highly precise and reliable results. To deal with large image sizes, we make use of image pyramids. We do not need approximate values, neither for orientation nor calibration, because we use direct solutions and robust algorithms, particularly fundamental matrices F, trifocal tensors $\mathcal{T}$, random sample consensus (RANSAC), and auto-calibration based on the image of the dual absolute quadric. We describe our strategy from end to end, and demonstrate its potential by means of examples, showing also one way for evaluation. The latter is based on imaging a cylindrical object (advertisement column), taking the last to be the first image, but without employing the closedness constraint. We finally summarize our findings and point to further directions of research.

1 INTRODUCTION

(Hartley and Zisserman, 2000) has transformed the art of producing a Euclidean model from basically nothing into text-book knowledge. As can be seen from recent examples such as (Nistér, 2004, Pollefeys et al., 2004, Lhuillier and Quan, 2005) a very high level has been reached.

We also head into this direction, making it possible to generate a Euclidean three-dimensional (3D) relative model (no scale, translation, and rotation known, i.e., seven degrees of freedoms undefined) from not much more than the images and the knowledge, that the images are perspective and sufficiently overlapping. Besides the latter, we make two in many practical cases reasonable assumptions, namely, that the camera is not too strongly (below about 15°) rotated around its optical axis between consecutive images and that all images are taken with one set of calibration (interior) parameters. The latter has to be true only approximately. While we cannot deal with zooming, we found empirically, that we can handle focusing.

The strategy, that we propose, particularly focuses on robust least-squares adjustment (Mikhail et al., 2001) in the form of bundle adjustment and least-squares matching (LSM). By means of affine LSM, we obtain highly precise conjugate points. Together with bundle adjustment, which we use for the computation of every fundamental matrix F as well as trifocal tensor $\mathcal{T}$, and after linking triplets via 3D projective transformation, we obtain highly precise and at the same time reliable solutions. This is demonstrated by means of two examples, in one of which a cylindrical object (advertisement column) was imaged with 28 images. Even though the information, that for the last image the first has been taken, has not been used in the adjustment, the cylinder is preserved very well.

Basically, our strategy rests on extracting points which we match highly precisely with LSM (cf. Section 2). Section 3 explains how hypothesis for conjugate points undergo rigorous geometric checks by projective reconstruction via computing F and $\mathcal{T}$, robustified by means of random sample consensus (RANSAC), as well as linking triplets via 3D projective transformation. All, including intermediate results of projective reconstruction are improved via robust bundle adjustment, important issues for which we explain in Section 4. As we deal with images of several Mega pixels, we employ image pyramids including tracking points via LSM through the pyramid (cf. Section 5). The projective reconstruction is upgraded to Euclidean via auto-calibration, described in Section 6. In Section 7 we demonstrate the potential of our strategy, particularly the high geometric precision and reliability achievable by means of LSM and bundle adjustment by means of an experiment specifically designed to evaluate the precision of the 3D reconstruction. Finally, we present a summary and directions for further research.

2 POINT EXTRACTION AND LEAST-SQUARES MATCHING

We start by extracting Förstner (Förstner and Gülch, 1987) points. An even distribution of the conjugate points on the image is enforced if possible by regional non-maximum suppression in the reference image of a particular matching step. No suppression is employed in the other images, because due to noise and occlusions the regionally strongest points in two images do not have to be the conjugate points.

Contrary to most approaches, we do not use the coordinates of the points for the conjugate points directly, but we determine relative coordinates by selecting one image and determining the relative shift of image patches around the points in the other images via LSM. This has the big advantage, that we obtain an estimate of the precision of the match.

To be able to deal with large baseline scenarios, we use as search space the size of the image. This naturally leads to a large number of hypotheses. As LSM is computational expensive, we first sort out unlikely candidates for conjugate points by means of normalized cross correlation. We particularly have found that correlating in red, green, and blue and combining the outcome by means of multiplication is a good choice for making use of color information. We employ a relatively low threshold of 0.7^3 to keep most of the correct points. Experiments with color spaces have not been successful as we found the color information to be mostly noisy, leading to bad correlation in the chrominance, etc., band.

As color information has already been used, we do not make use of it for LSM. For it, we employ affine geometric transformation, because the parameters for a projective transformation cannot be reliably determined for image patches in the range of 11×11 pixels. Additionally to the the six affine geometric parameters, we determine a bias and a drift (contrast) parameter for the brightness. For two images we just match the second to the first. For three and more images we determine an average image in the geometry of the reference image. Matching against it, we avoid the bias by a radiometrically badly selected reference image (e.g., distorted by occlusion).

The result of this step are highly precise image coordinates for the conjugate points including an estimate of the precision. This value is mostly over optimistic (one often obtains standard deviations in the range of one hundredth of a pixel), but they still give a good hint on the relative quality of the solution obtained.

3 ROBUST PROJECTIVE RECONSTRUCTION

The conjugate points of the preceding section are input for projective reconstruction. Basically, the goal is reconstruction of the whole sequence. Because of the inherent noise and due to problems with similar and repeating structures as well as occlusions, the strategy needs to be rather robust, and at the same time efficient.

We have decided to use triplets as the basic building block of our strategy. This is due to the fact, that by means of the intersection of three image rays one can sort out wrong matches, i.e., outliers, highly reliably. Opposed to this, one cannot check the depth for image pairs, as the only constraint is, that a point has to lie on the epipolar line. Even though using triplets as basic building block, combinatorics suggests to actually start with image pairs, restricting the search space via epipolar lines. For the actual estimation of the relations of pairs and triplets we employ F and $\mathcal{T}$ (Hartley and Zisserman, 2003). Triplets are computed sequentially and are linked by means of projecting points of the preceding triplet via the new $\mathcal{T}$ into the new last image resulting into (n+1)-fold points as well as computing the projection matrix of the last image via 3D projective transformation for the first and second but last images. (Projection matrices for F and $\mathcal{T}$ can be obtained with the standard algorithms explained in (Hartley and Zisserman, 2003).) Finally, points not yet seen are added.

Of extreme importance for the feasibility of our strategy is the use of robust means, particularly RANSAC (Fischler and Bolles, 1981), that we use for the computation of F and $\mathcal{T}$. As we are dealing with a relatively large number of outliers in the range of up to 80%, RANSAC becomes especially for the computation of $\mathcal{T}$ extremely slow. This is mostly due to the fact, that for reliably estimating $\mathcal{T}$, it is necessary to compute a point-wise bundle adjustment. We use a modified version of RANSAC speeding up the computation by more than one order of magnitude for high noise levels, where as shown in (Tordoff and Murray, 2002), often much larger numbers of iterations are needed to obtain a correct result than predicted by the standard formula given in (Hartley and Zisserman, 2003).

4 ROBUST BUNDLE ADJUSTMENT

Bundle adjustment is at the core of our strategy. We have found, that only by adjusting virtually all results, we obtain a high precision, but also reliability. The latter stems from the fact, that by enforcing highly precise results for a large number of points, one can guarantee with a very high likelihood, that the solution is not random.

Basically this means, that when estimating F and $\mathcal{T}$, we compute the optimum RANSAC solution for junks of several hundreds of iterations and then we run a projective bundle adjustment on it. This is done a larger number of times (we have found empirically five to be the minimum number), as the bundle adjustment solution is partly much better than the RANSAC solution and its result can vary a lot. But having several instances of bundle solutions, there is nearly always one which is sufficiently precise and representing the correct solution.

We employ projective as well as Euclidean bundle adjustment, both including radial distortion $ds = 1. + k_2 * (r^2 - r_0^2) + k_4 * (r^4 - r_0^4)$ with r the distance of a point to the principal point (or its estimate) and r_0 the distance where ds is 0. $r_0 = 0.5$ is used as recommended in literature and empirically verified. We have found by a larger number of experiments, that it is important to employ radial distortion only after outlier removal. It is not used at all for the determination of F or $\mathcal{T}$, but only after we have tracked down points to the original image resolution (cf. below).

We originally wanted to employ standard least-squares adjustment without Levenberg Marquardt stabilization (Hartley and Zisserman, 2003), to avoid a bias during estimation. Therefore, we are using the SVD-based minimal parameterization proposed in (Bartoli and Sturm, 2001) for the first camera for projective bundle adjustment. Yet, we have found, that only by means of a Levenberg Marquardt stabilization we can deal with the large initial distortions of the solution caused by outliers. Particularly, this means, that we multiply the elements of the diagonal of the normal equations with $1 + stab$, the stabilization parameter $stab$ being adaptively determined by means of varying it with a factor of 10 between 1.e-5 and 1.

We base the robustness of bundle adjustment on standardized residuals $\bar{v}_i = v_i/\sigma_{v_i}$ involving the standard deviations σ_{v_i} of the residuals, i.e., the differences between observed and predicted values. As a first means we employ reweighting with $w_i = \sqrt{2 + \bar{v_i}^2}$ (McGlone et al., 2004). Additionally, having obtained a stable solution concerning reweighting, outliers are characterized by $\bar{v_i}$ exceeding a threshold, which we have set to 4, in accordance with theoretical derivations and empirical findings, eliminating the outliers for the next iteration.

For bundle adjustment, efficient solutions are extremely important. E.g., a 29 image sequence as the one presented below leads to more than thirty thousand unknowns, making straightforward computation impossible. We therefore follow (Mikhail et al., 2001) and reduce the normal equations in two steps: First, we reduce the points. Secondly, we also reduce parameters which are common to all, or at least sets of images, namely the calibration and / or (radial) distortion parameters. This results into a tremendous reduction in computation time and storage requirements, even when computing also σ_{v_i}.

5 HIERARCHICAL PROCESSING VIA PYRAMIDS

As we deal with relatively large images in the range of 5 Mega pixels or above and we assume at the same time, that we do not know the percentage or direction of overlap of the images, only a hierarchical scheme allows for an adequate performance. We particularly compute image pyramids with a reduction factor of 2. For the highest level we found that a size of about 100×100 pixels is sufficient in most cases. On this level we compute F. $\mathcal{T}$ are computed on the second highest and for images with a size of more than 1000×1000 pixels also on the third highest level.

We do not compute $\mathcal{T}$ on the fourth highest or lower levels, firstly due to the complexity of the matching and secondly because already on the second or third highest level we obtain for most sequences hundredth of points, more than enough for a stable and precise solution. To still use the information from the original resolution, we track the points via LSM down to the original resolution once the sequence has been oriented completely on the second or third highest level. This is rather efficient also for images of several Mega pixels. As reference image we use for every point the image, where the point is closest to the center of the image, assuming that there the perspective distortion of the patches around the points is minimum on average. After tracking, a final robust projective bundle adjustment is employed, at this time including radial distortion.

6 AUTO-CALIBRATION

To proceed from projective to Euclidean space, one needs to estimate the position of the plane at infinity π_∞ as well as the calibration matrix

$$\boldsymbol{K} = \begin{bmatrix} c & c \cdot s & x_0 \\ & c \cdot (1+m) & y_0 \\ & & 1 \end{bmatrix}$$

with c the principal distance, m the scale factor between x- and y-axis, needed, e.g., for video cameras with rectangular instead of quadratic pixels, x_0 and y_0 the coordinates of the principal point in x- and y-direction, and finally s the sheer, i.e., the deviation of a 90° angle between the x- and the y-axis. The latter can safely be assumed to be zero for digital cameras.

To compute $\boldsymbol{K}$ and a transform to upgrade our projective to a Euclidean configuration, we use the approach of Pollefeys (Pollefeys et al., 2002, Pollefeys et al., 2004). It is based on the image of the dual absolute quadric

$$\omega^* \approx \mathsf{K}\mathsf{K}^\top \approx \mathsf{P}\Omega^*\mathsf{P}^\top$$

which is related to the calibration matrix multiplied with any scalar $\neq 0$ (K) and the dual absolute quadric Ω^*, projected by the projection matrices P. (Pollefeys et al., 2002, Pollefeys et al., 2004) employ knowledge about meaningful values and their standard deviations for the parameters of K to constrain the computation of Ω^* such as, that the principal distance is one with a standard deviation of nine and all other parameters are zero with standard deviations of 0.1 for the principal point and m and 0.01 for s. The result is a transformation matrix from projective to Euclidean space and one $\boldsymbol{K}$ for every image.

We have experienced, that the resulting Euclidean configuration can be some way off the final result, especially for longer sequences. I.e., for the sequence of 29 images below, the estimated principal distance, known to be constant, varied between 0.3 and 3. To avoid this problem, we have found it to be sufficient to compute the calibration for the first few images and transform the rest of the sequence accordingly. Though this has worked for our experiments, a better way might be to define a number of images n, say three or five, and compute the calibration, which is of very low computational complexity, for all subsequent n images. Finally, the solution should be taken with the smallest summed up standard deviation of all parameters for the average $\boldsymbol{K}$.

As demonstrated, e.g., by the experiments below, robust bundle adjustment including radial distortion is an absolute must after calibration. We start with configurations where the back projection errors can be in the range of several hundred pixels. This stems from the fact, that the calibration procedure produces locally varying $\boldsymbol{K}$ (cf. above). Using Levenberg Marquardt stabilization, it is possible to bring down theses large values to fractions of a pixel. In the beginning the multiplication factor for the elements on the main diagonal can be as high as two, i.e., $stab = 1$.

Because also after projective bundle adjustment there still can be a large number of outliers, also the strategy for bundle adjustment was found to be very important. This is due to the fact, that we accepted sound configurations in projective space, which yet can imply relatively different $\boldsymbol{K}$. Optimizing all parameters of an average $\boldsymbol{K}$ simultaneously can lead to initially very wrong values for x_0, y_0, and s. It was therefore found to be very important to first optimize only c and $c \cdot (1 + m)$, and to optimize the rest of the parameters only when this adjustment has converged. Optimizing c and $c \cdot (1 + m)$ independently makes the whole procedure less stable on one hand, but allows on the other hand to check the quality of the result by comparing both.

7 EXPERIMENTS AND EVALUATION

In this section we report about results for the proposed strategy and propose one means to evaluate results. All images used in the experiments shown here have been acquired with the same camera, namely a Sony P100 5 Mega pixel camera with Zeiss objective using the smallest possible focal length / principal distance to optimize the geometry of the intersections. To guarantee sharp images (and to make the experiments more difficult), the camera was allowed to auto-focus, leading to slightly varying principal distances. We first present the result for one example out of tens, namely the scene yard, for which our strategy works reliably using the same set of parameters. I.e., one acquires the images, runs the program implementing the strategy and obtains the result consisting of 3D points, camera translations and rotations as well as the calibration, all including standard deviations.

Additionally, we report about one experiment we have devised to evaluate the quality of the solution. For it we acquired 28 images of an advertisement column, which is close to a perfect cylinder. The images have been taken walking unconstrained, so there is some flexibility in the orientation. Though, by always trying to be able to see the whole width of the column, there was a strong constraint to actually take the images from positions on a circle.

The scene yard consists of eight images taken in a backyard. The first three images and the last image are given in Figure 1. Figure 2 shows a view on the resulting VRML model. For the sequence we have obtained 426 threefold points, i.e., points which could be matched in three images, 377 fourfold, 228 fivefold, 103 sixfold, and 20 sevenfold points resulting in an uncalibrated back projection error σ_0 of 0.39 pixels and a σ_0 of 0.3 pixels after calibration. Further parameters such as the calibration matrix $\boldsymbol{K}$ can be found in Table 1.

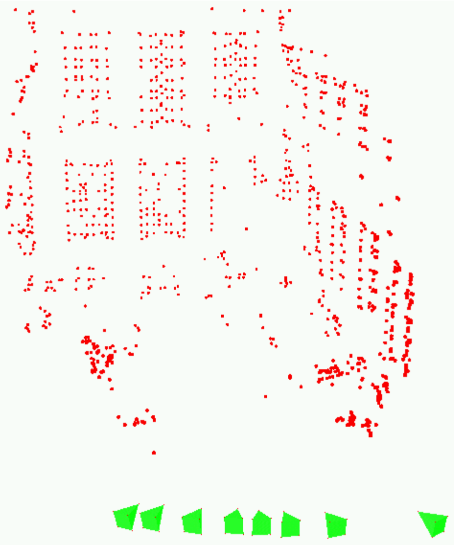


Figure 2: Visualization of points (red) and cameras (green pyramids) of model yard

number images	8
σ_0 projective / Euclidean	0.39 / 0.30 pixel
$\boldsymbol{K}$	$\begin{matrix} 1.247 & -0.001 & -0.004 \\ & 1.251 & 0.0024 \\ & & 1 \end{matrix}$
k_2 / k_4 (radial distortion)	-0.041 / -0.069

Table 1: Results for sequence yard

Of the 28 approximately evenly spaced images of the advertisement column / cylinder, the first three and the fifth are shown in Figure 3. Four other images, showing the variety of texture found on the column, are given in Figure 4.

For the evaluation we have devised three experiments. The first is with the original resolution of 2592×1944 pixels, the second with the resolution reduced by a factor of three, i.e., 864×648 pixels, and for the last experiment we have reduced the resolution by a factor of three and the number of images, wherever there is enough texture, by a factor of two. I.e., we have taken the first, third, and fifth image, etc., as shown in Figure 4.

On the original resolution we obtained 2498 threefold, 3387 fourfold, 2559 fivefold, 1085 sixfold, 309 sevenfold, and 45 eightfold points, as well as one ninefold point resulting in a back projection error of $\sigma_0 = 0.1$ pixels on the third highest pyramid level and of $\sigma_0 = 0.29$ pixels after tracking down to the original resolution. Auto-calibration resulted into estimated $c = 1.04$ and $c \cdot (1 + m) = 1.05$. The resulting configuration is given in Figure 5 left. The back projection error has been in the range of 500 pixels before bundle-adjustment. Bundle adjustment reduced it to 0.19 pixels. The final result is very close to a perfect cylinder as proven by Figure 5 right.

Table 2 shows a comparison of the results. They are rather similar for the original and the reduced resolution sequence. This suggests, that probably because of the relatively small pixel size of the employed mid-end Sony P 100 consumer camera, the original resolution does not convey much more information than the reduced resolution. Similar findings have been made for other sequences. On the other hand, the results for the sequence with the reduced number of images are rather different. This probably stems from the fact, that the overlap between the images is small and the view angles on the surface are partly rather large. For large areas of weak or no texture, such as in image thirteen (cf. Figure 4), we even had to use the original configuration. One can see this, e.g., as a hole in the upper right of the cylinder in Figure 5, right. The comparison of Tables 2 and 1 shows, that even though the time between acquiring the cylinder and the yard sequence was about one year, all the parameters including the distortion are rather similar, if enough images were used for the cylinder sequence. (Please remember, that the same camera has been used.)

For the evaluation of the different versions of the cylinder sequence, we have taken the first image to be the last image of the sequence as well. Instead of using this information in the bundle adjustment, we employ it for evaluation by

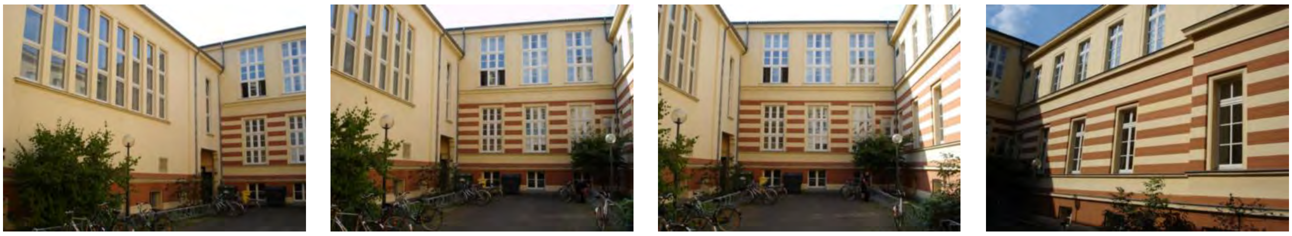


Figure 1: First three images and image eight, i.e., last image, of sequence yard

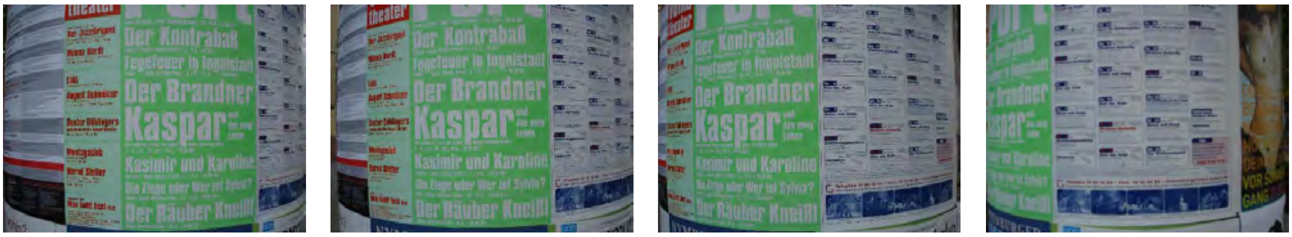


Figure 3: First three images and fifth image of the original sequence cylinder with 28 images

DU SOLLST NICHT BEGEHREN DEINES NÄCHSTEN WEIB, NOCH ALLES WAS DEIN NÄCHSTER HAT.
VOR SONNENAUFGANG 29.05.05
GRILL
NACHTBIERGARTEN
3,80 €
BACKSTAGE
www.unterfahrt.de

Figure 4: Images eight, thirteen, eighteen and twenty three of the original sequence cylinder

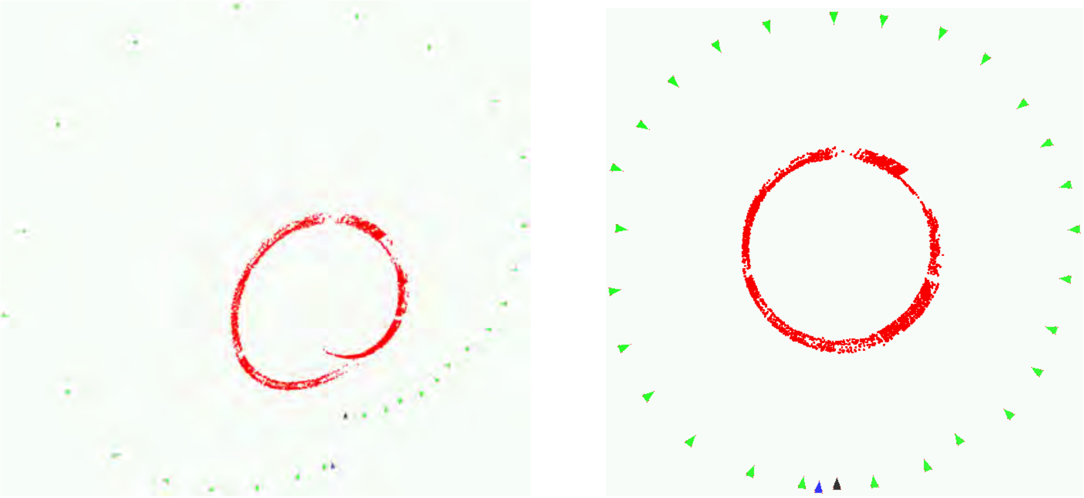


Figure 5: Result for the original sequence cylinder before (left) and after (right) robust Euclidean bundle adjustment. The first and the last camera are marked as black and blue and the rest of the cameras as green pyramids. Points are shown in red.

comparing the parameters of the first and the last camera, which ideally should be the same. Table 3 gives two different types of descriptions, namely the translation in x-, y-, and z-direction of the first = last camera in relation to the radius of the circle constructed by all cameras, as well as the difference in rotation (this is the rotation angle of an axis-angle representation), the latter also in terms of a single image. One can see, that the difference is rather small for the original as well as for the sequence with reduced resolution. Only for the sequence with the reduced number of images there is a significant reduction of the quality.

8 SUMMARY AND CONCLUSIONS

We have shown, that via least-squares adjustment based techniques, particularly least-squares matching and bundle adjustment, highly precise and at the same time reliable results can be obtained. This has been demonstrated by means of a cylindrical object, for which it was shown, that the ring of cameras closes very well and for which at least visually also the shape is preserved extremely well. By means of enlarging the distance between the cameras, we have shown difficulties of the strategy when using a weaker

	original	resolution reduced by 3	reduced number images
number images	29	29	22
σ_0 projective / Euclidean	0.29 / 0.19 pixel	0.12 / 0.08 pixel	0.24 / 0.13 pixel
K	$\begin{matrix} 1.239 & 0.0002 & 0.002 \\ & 1.241 & 0.0001 \\ & & 1 \end{matrix}$	$\begin{matrix} 1.242 & 0.0001 & 0.003 \\ & 1.241 & -0.0003 \\ & & 1 \end{matrix}$	$\begin{matrix} 1.168 & -0.0006 & -0.0015 \\ & 1.179 & -0.0062 \\ & & 1 \end{matrix}$
k_2 / k_4 (radial distortion)	-0.040 / -0.060	-0.043 / -0.053	-0.041 / -0.069

Table 2: Results for sequence cylinder

	original	resolution reduced by 3	reduced number images
dx / dy / dz in % of radius circle images	3.5 / -0.36 / 0.74	3.8 / -0.81 / 0.8	7.1 / -1. / 1.1
$d\phi$ global / $d\phi$ per image	$5°$ / $0.18°$	$5.8°$ / $0.21°$	$8.7°$ / $0.41°$

Table 3: Differences in translation and rotation of the parameters of the first = last image of sequence circle. dx, dy, and dz are given in relation to the approximate radius of the circle constructed by the camera positions.

geometry.

A first issue for further research is a more quantitative evaluation of the shape of the given object. This could be done in our case by fitting a cylinder to the object and determining the distances from this cylinder. Though the object is not an ideal cylinder, it should be rather close to it.

Calibration is a further issue. Here the approach of (Nistér, 2004) based on the cheirality constraint seems to be extremely promising. We also still need to deal with planar parts of the sequence. For this we want to follow (Pollefeys et al., 2002), though we note that we have found the issue of model selection (homography versus **F** or $\mathcal{T}$) rather tricky.

Finally, an issue that we see as particularly important to achieve the goal of being able to orient also traditional photogrammetric close range image setups is matching which is more invariant with respect to strong geometric distortion. For it we find especially (Georgescu and Meer, 2004) and (Lowe, 2004) very interesting.

REFERENCES

Bartoli, A. and Sturm, P., 2001. Three New Algorithms for Projective Bundle Adjustment with Minimum Parameters. Rapport de Recherche 4236, INRIA, Sophia Antipolis, France.

Fischler, M. and Bolles, R., 1981. Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. Communications of the ACM 24(6), pp. 381–395.

Förstner, W. and Gülch, E., 1987. A Fast Operator for Detection and Precise Location of Distinct Points, Corners and Centres of Circular Features. In: ISPRS Intercommission Conference on Fast Processing of Photogrammetric Data, Interlaken, Switzerland, pp. 281–305.

Georgescu, B. and Meer, P., 2004. Point Matching Under Large Image Deformations and Illumination Changes. IEEE Transactions on Pattern Analysis and Machine Intelligence 26(6), pp. 674–688.

Hartley, R. and Zisserman, A., 2000. Multiple View Geometry in Computer Vision. Cambridge University Press, Cambridge, UK.

Hartley, R. and Zisserman, A., 2003. Multiple View Geometry in Computer Vision – Second Edition. Cambridge University Press, Cambridge, UK.

Lhuillier, M. and Quan, L., 2005. A Qasi-Dense Approach to Surface Reconstruction from Uncalibrated Images. IEEE Transactions on Pattern Analysis and Machine Intelligence 27(3), pp. 418–433.

Lowe, D., 2004. Distintive Image Features from Scale-Invariant Keypoints. International Journal of Computer Vision 60(2), pp. 91–110.

McGlone, J., Bethel, J. and Mikhail, E. (eds), 2004. Manual of Photogrammetry. American Society of Photogrammetry and Remote Sensing, Bethesda, USA.

Mikhail, E., Bethel, J. and McGlone, J., 2001. Introduction to Modern Photogrammetry. John Wiley & Sons, Inc, New York, USA.

Nistér, D., 2004. Untwisting a Projective Reconstruction. International Journal of Computer Vision 60(2), pp. 165–183.

Pollefeys, M., Van Gool, L., Vergauwen, M., Verbiest, F., Cornelis, K. and Tops, J., 2004. Visual Modeling with a Hand-Held Camera. International Journal of Computer Vision 59(3), pp. 207–232.

Pollefeys, M., Verbiest, F. and Van Gool, L., 2002. Surviving Dominant Planes in Uncalibrated Structure and Motion Recovery. In: Seventh European Conference on Computer Vision, Vol. II, pp. 837–851.

Tordoff, B. and Murray, D., 2002. Guided Sampling and Consensus for Motion Estimation. In: Seventh European Conference on Computer Vision, Vol. I, pp. 82–96.

GLOBAL UNCERTAINTY IN EPIPOLAR GEOMETRY VIA FULLY AND PARTIALLY DATA-DRIVEN SAMPLING

C. Engels, D. Nistér

Center for Visualization and Virtual Environments, Dept. of Computer Science,
University of Kentucky, Lexington, KY 40507, USA
(engels@vis, dnister@cs).uky.edu

KEY WORDS: Relative orientation, Stucture from Motion, Epipolar Geometry

ABSTRACT:

In this paper we explore the relative efficiency of various data-driven sampling techniques for estimating the epipolar geometry and its global uncertainty. We explore standard fully data-driven methods, specifically the five-point, seven-point, and eight-point methods. We also explore what we refer to as partially data-driven methods, where in the sampling we choose some of the parameters deterministically. The goal of these sampling methods is to approximate full search within a computionally feasible time frame. As a compromise between fully representing posterior likelihood over the space of fundamental matrices and producing a single estimate, we represent the uncertainty over the space of translation directions. In contrast to finding a single estimate, representing the posterior likelihood is always a well-posed problem, albeit an often computionally challenging one. Furthermore, this representation yields an estimate of the global uncertainty, which may be used for comparison between differing methods.

1 . INTRODUCTION

Estimation of the relative orientation between two images is an extensively researched subject in computer vision. Many methods have been proposed and the state of the art is now quite elaborate and mature. In our view, the main requirements on an estimation method are that it

- Is accurate (both locally and globally)
- Is robust
- Is computationally efficient
- Can exploit all constraints, exact and approximate
- Gives a truthful uncertainty estimate (local and global)

It is widely accepted that accuracy is best achieved with iterative refinement, called bundle adjustment [24], according to a cost function that is derived from a realistic model of the problem. However, bundle adjustment is dependent on an initial starting point and only achieves what we refer to as local accuracy, which is the ability to precisely pinpoint a local minimum of the cost function. Perhaps even more important and challenging in computer vision is to, insofar as possible, achieve global accuracy, which is the ability to reliably locate the global minimum of the cost function.

Robustness is achieved by using an appropriate data model that includes data distortions and outliers. Computational efficiency is always desirable, although the requirements are more stringent in some applications than others. It is likewise desirable to use all available constraints, such as camera calibration information.

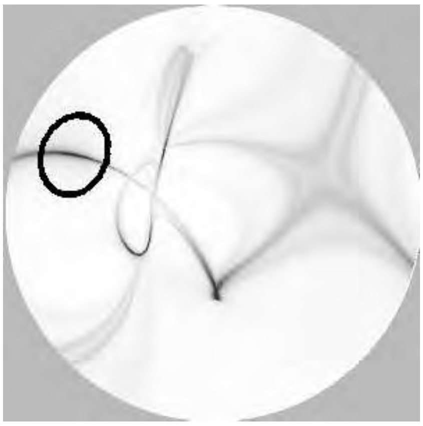


Figure 1: We derive an uncertainty representation for epipolar geometry parameterized by the epipole in the first image. The figure shows an example of the uncertainty representation when the number of point correspondences is too low, leading to intricate patterns of probability mass. The global maximum is circled, but notice the multiple peaks captured by the representation.

Gauging the uncertainty is important, since without a notion of how likely it is that the estimate at hand is in error, it is very hard to take any useful action based upon it. It is best-practice to gauge local uncertainty around an estimate by analyzing the local shape of the cost function around the minimum. However, such an uncertainty measure only makes sense if the global minimum was truly found. Moreover, it assumes that the cost function is unimodal and nicely behaved. This is seldom the case. Due to outliers, noise, the nonlinear nature of the problem, planar scenes and small translation, the cost function may lack a clear global minimum or have several throughs of complicated shape.

Therefore, to assess global uncertainty, an estimation method should ideally provide a representation of the posterior probability distribution over all the regions of parameter space where the probability is significant.

For strong data, producing a single estimate is possible. However, there will always be situations with ambiguous data, in which obtaining a single estimate is essentially an ill-posed problem. On the other hand, provided we have selected an appropriate data model, representing the posterior distribution is always a well-posed problem. Representing the posterior may be computationally difficult, but it is well-posed for any input data.

Our approach draws upon background material in probabilistic Bayesian frameworks and multiple view geometry. Due to space limitations, we by necessity have to assume that the reader has some familiarity with these concepts. The interested reader is referred to [4, 5, 20] for the former and [6, 9, 15] for the latter.

2 . APPROACH

Ideally, we would like to evaluate the likelihood $p(d|w)$ for all possible world states w to derive our representation for the posterior distribution. However, it is impractical to perform full search over a high-dimensional space (in this case five or more dimensions). Such a complete representation would also be unmanageable for a module that needs to use the results for further computation or decision making.

To reach an efficient representation of the likelihood, we will rely on the following observation: If the epipole in the first image is known, the remaining parameters of the fundamental matrix (simply rotation in an uncalibrated setting) are uniquely determined unless all the points from the point correspondences and the epipole lie on a common conic in the second image.

Thus it is natural to represent the likelihood with an explicit representation indexed by the translation direction (epipole in the first image).

The usefulness of treating the translation and rotation differently has been understood by many authors and exploited in different ways, see for example [10, 3, 18, 1]. It is also closely related to the highly popular plane-plus-parallax approach [11, 14, 21, 23, 13], where one relies on the existence of a dominant homography and solves for that in order to guide the search for the translation direction.

3 . DATA DRIVEN SAMPLING

As argued above, we can not search the likelihood over the whole parameter space. Several authors have noted that it can be much more efficient to search the parameter space with data-driven hypothesis generators [2, 25]. We will use hypothesis generation in a similar manner as in RANSAC [7], where minimal samples of correspondences are randomly chosen from the whole set of correspondences. A minimal sample contains the smallest number of data points that will determine the geometric relation up to a finite number of solutions. The samples are made minimal to minimize the risk of including devastating outliers. In this case, a minimal sample contains seven correspondences for the fundamental matrix and five for the essential matrix. We refer to this as fully data-driven sampling, since the correspondences ideally should determine the fundamental matrix. We will also use partially data-driven sampling, where for a given translation direction, we take samples containing the smallest number of correspondences that will determine the remaining parameters of the fundamental matrix up to a finite number of solutions. The samples contain five correspondences to determine the fundamental matrix in the uncalibrated case and three correspondences to determine the essential matrix given translation direction in the calibrated case.

4 . REPRESENTATION

If we can derive an accurate representation of the data likelihood $p(d|w)$ it can be converted into a representation of the posterior by multiplying with the prior. The representation of the posterior can then support any inferences we wish to make based on the data.

We consider the world state w to be represented by the fundamental matrix F and the data d to be represented by all the point correspondences, denoted by X. Bayes' rule then becomes

$$p(F|X) \propto p(X|F)p(F). \tag{1}$$

We store the hypotheses for the fundamental matrix in a two-dimensional array indexed by epipole in the first image. Our goal is to find the best fundamental matrix hypothesis for each cell of the array and the integral likelihood in each cell. Let $\Omega(e)$ denote the set of all fundamental matrices with the epipole e in the first image. The desired output from our approach is

$$F_{opt}(e) = \begin{array}{c} arg\ max \\ F \in \Omega(e) \end{array} p(X|F) \tag{2}$$

and

$$f(e) = \int_{F \in \Omega(e)} p(X|F) dF. \tag{3}$$

for all values of the epipole e. The latter can be computed by a Laplace approximation around the former.

Along the lines of our above motivation, it is assumed that the likelihood $p(X|F)$ has a unique narrow peak in $\Omega(e)$. By assuming that the prior $p(F)$ is smooth in comparison to the extent of the peak, the user of the output can make the approximation

$$p(e|X) \propto \int_{F\in\Omega(e)} p(X|F)p(F)dF \approx p(F_{opt}(e))f(e). \quad (4)$$

In a similar manner, most inferences that one may wish to make based on the data has to do with an integral of some function $g(F)$ times the posterior likelihood. Such integrals

$$\int_e \int_{F\in\Omega(e)} g(F)p(F|X)dFde \quad (5)$$

can be approximated as

$$\frac{\int_e g(F_{opt}(e))p(F_{opt}(e))f(e)de}{\int_e p(F_{opt}(e))f(e)de}. \quad (6)$$

The advantage is that the inferences can be made outside the relative orientation module with any choice of prior $p(F)$ using only $F_{opt}(e)$, $f(e)$ and easy two-dimensional integrals.

If this can be done efficiently and reliably, inferences can be made in an application-dependent manner based on the resulting representation, without major alterations to the core of the computer vision algorithm.

4.1 Prior Likelihood

In the simplest case, the prior likelihood $p(F)$ is set to uniform. In some cases we may have more prior information. For example, if we are calibrating a stereo-head, we typically have approximate knowledge of the location of the epipole and also of the relative rotation. We may also work in the uncalibrated setting, but use the prior to put approximate constraints on the calibration.

4.2 Posterior Likelihood

We use a Sampson approximation (see [9]):

$$s(x, x', F) = \frac{(x'^\top Fx)^2}{(Fx)_1^2 + (Fx)_2^2 + (x'^\top F)_1^2 + (x'^\top F)_2^2} \quad (7)$$

where the homogeneous coordinates for the points are assumed to be normalized such that their last coordinates are one. It approximates the squared sum of magnitudes of the smallest perturbation required to bring the image point correspondence $x \leftrightarrow x'$ into agreement with the epipolar geometry described by the fundamental matrix ($x'^\top Fx = 0$). This approximation has been found superior to symmetric epipolar distance and other approximations of similar computational complexity [27].

We model our data likelihood as

$$p(X|F) \propto (\prod_{i=1}^{N} \sigma^2(\sigma^2 + s(x_i, x'_i, F))^{-1})^{N^{-k}}, \quad (8)$$

where σ is a scale parameter, which we typically set to one pixel of a CIF image (352×288), N is the number of point correspondences, and $0 \le k \le 1$. We determine the value of k experimentally in section 6.4. We have also tried the standard way of assuming that the reprojection errors are conditionally independent given the world configuration ($k = 0$), dogmatically leading to a product of many independent factors, where each factor is related to a single point correspondence. However, we have found that although this produces sensible peak locations of the likelihood, it leads to an unrealistically rapid fall-off around the likelihood peak, resembling a delta-function and not a realistic model of any practical situation.

5 . HYPOTHESIS GENERATORS

The hypothesis generators we use in our experiments are:

- 5-Point (Calibrated)
- 7-Point (Uncalibrated)
- 8-Point (Uncalibrated)
- 3-Point+Epipole (Calibrated)
- 5-Point+Epipole (Uncalibrated)

For fully data-driven sampling in the calibrated case, we use the 5-point method (5pt)[16]. In the uncalibrated case, we use the 7-point (7pt) method and the 8-point (8pt) method [9].

The 3-point+epipole (3pt+e) and 5-point+epipole (5pt+e) methods are partially data-driven generators. The former was presented in [17]. It uses the point constraints and the known epipole to restrict the essential matrix to a 3-dimensional linear space. The calibration constraints are then added, leading to two conics that are intersected, which yields four solutions. This method can be carried out extremely fast in closed form. The latter is related to a classical result, which is that given five point correspondences, the epipoles correspond by a fifth-degree Cremona mapping, also discussed in [26]. This method gives a unique solution. It can for example be implemented by stacking linear constraints from the point correspondences and the known epipole into an 8×9 matrix, subsequently extracting the unique nullvector.

6 . EXPERIMENTS

6.1 Construction of the Likelihood Image

To determine the uncertainty of an estimated epipole, we first computed a quantized posterior likelihood over a hemisphere of epipoles. The sign of the epipole can only be determined using cheirality [9], which we do not enforce. We mapped the hemisphere onto a 300×300 image. In each cell, we computed the optimal fundamental matrix with translation direction in the cell. In the cases of the partially data-driven methods, we deterministically sampled the translation direction over all quantized translations. In the fully data-driven methods, the translation direction was determined by the generated hypothesis. We sampled the entire epipolar space, or about 70000 cells, in multiple sweeps, using random sets of point correspondences for each sample. In the partially data-driven methods, a small perturbation in the translation was added within each cell to more fully represent possible fundamental matrices.

We explored the likelihood images for both synthetic and real data. In the synthetic case, images with known relative orientation were created with a scene volume of random points. The image points were then perturbed with Gaussian noise equivalent to one pixel of a CIF image. Finally, outliers were simulated by uniformly scattering a percentage of the image points in one image. For real data, we tracked Harris corners, using normalized correlation for matching. The camera was calibrated in order to compare calibrated and uncalibrated methods.

6.2 Convergence of the Likelihood

We investigated how quickly each method converges to the likelihood over the entire hemisphere. A straightforward measure of the error in the estimated likelihood is given by

$$error = \int_e (p(e) - \hat{p}(e))de, \tag{9}$$

where p is the true likelihood and $\hat{p}$ is the estimated likelihood. Ideally, a full search over the space of fundamental matrices would be used to create p. Since this is infeasible, we approximated the true likelihood as the maximum found using all five tested methods in an extremely long computation. The final image, shown on the top left of Figure 2, was created with 1000 sweeps, or about 7×10^7 samples per method.

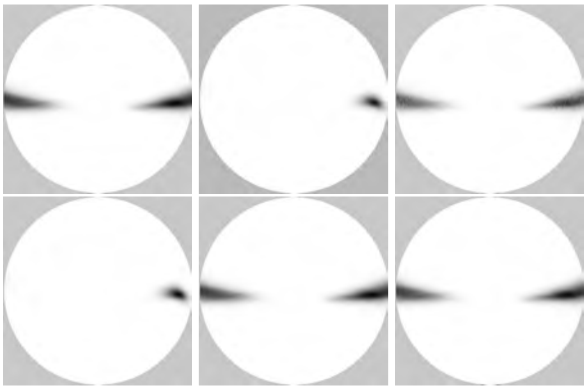


Figure 2: Posterior likelihood images of a scene with sideways translation over 1000 sweeps of the epipolar space. From left to right, top to bottom: true likelihood; 3pt+e method; 5pt+e method; 5pt method; 7pt method; 8pt method.

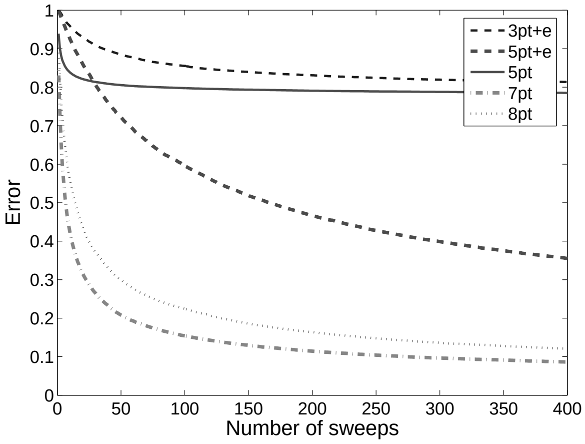


Figure 3: Comparison of convergence rates for the various hypothesis generation methods. Hypothesis generation times are not taken into account.

6.2.1 Comparison of Partially and Fully Data-Driven Methods

We compared the methods by examining the rate of convergence to the likelihood. Since the uncalibrated methods create hypotheses from the space of fundamental matrices, while the calibrated methods generate hypotheses from the more restricted space of essential matrices, the uncalibrated methods uncover a greater probability mass. Because we calibrated the image points, the true solution is an essential matrix, so the mass uncovered by the uncalibrated methods may be overestimated.

We sampled with all methods simultaneously and recorded the errors. Because several methods produce multiple solutions, it was important to ensure that the methods had equivalent numbers of samples. For the 3pt+e and 7pt methods, we disambiguated the solutions by scoring one additional point correspondence and choosing the hypothesis with the highest single point likelihood. For the 5pt method, which may produce up to 10 real solutions representing extra potentially valid solutions such as planar ambiguities, we stored the hypotheses and computed the likelihood of one hypothesis per sampling round.

As seen in Figure 3, the fully data-driven uncalibrated methods explore the greatest probability mass early in the computation, while the 5pt+e method slowly converges to the same value. The calibrated methods converge to a different posterior likelihood, although the fully data-driven method again converges faster than the partially data-driven method.

6.3 Estimation of Confidence Intervals

Once we have the posterior likelihood, we create confidence intervals by finding the global maximum in the posterior likelihood and measuring the fraction of the probability mass that lies within a certain distance of the max-

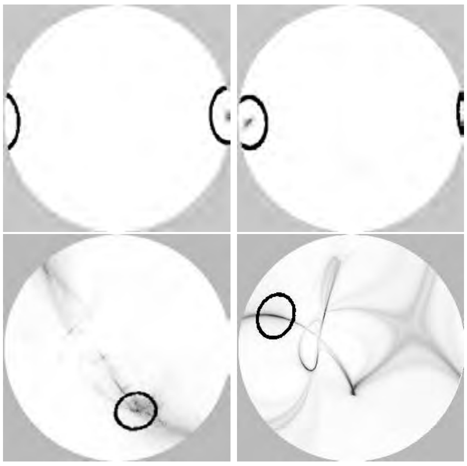


Figure 4: Examples of confidence intervals in an image sequence with a leftward translation. From left to right and top to bottom, the respective probability masses within each circled confidence interval are: 0.865, 0.567, 0.204, 0.065.

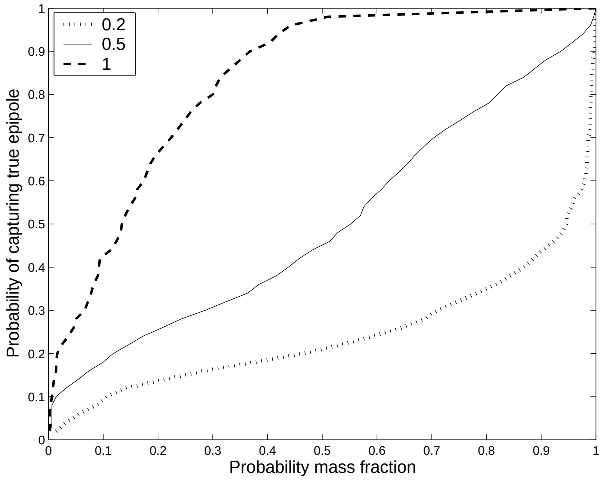


Figure 5: Cumulative distribution functions of confidence levels for varying values of k. Note that $k = 0.5$ most closely matches a uniform random variable.

imum. That is, we start from a maximal acceptable distance, which then in turn determines the confidence level. Typically, we used a distance of 5 degrees on the sphere. Figure 4 shows examples of confidence intervals in likelihood images. The top two images represent cases with many inlier point correspondences. The bottom left image represents a case with relatively few correspondences and low stability. The bottom right image represents a case that has a critically small number of correspondences. However, these deficiencies are apparent in the representation, due to the small probability mass within the confidence intervals.

6.4 Verification of Confidence Interval

If we construct confidence intervals and collect statistics on the confidence level needed to capture the true epipole, this confidence level should ideally be a uniformily distributed random variable. To explore the sensitivity of our confidence intervals to discrepencies between the assumed data model and the actual data model, we use synthetic data along with our cost function, and measure the deviation from uniform distribution. A synthetic scene with 30% outliers and a known epipole was created.

A 100×100 likelihood image was created using 10 sweeps of the 5pt+e method, and the probability mass required to capture the true epipole was recorded. This was repeated 500 times, and the cumulative distribution function of the mass fractions was plotted. A sublinear cdf indicates overconfidence, while a superlinear cdf indicates underconfidence.

We found the best value for k from Equation (8) to be approximately $1/2$. As seen in Figure 5, this achieves a balance in the confidence estimates, while $k = 1$ leads to underconfidence and $k = 0$ to overconfidence, with a highly peaked likelihood.

6.5 Finding Optimal Baseline in an Image Sequence

As a practical test of inference with our uncertainty representation, we aim to find a pair of frames in an image sequence that results in the best possible 3-D reconstruction of a scene. To accomplish this, we search for an optimal baseline between camera positions, such that we have a large translation required for accurate reconstruction while still maintaining a reasonable number of inlier point correspondences. Obtaining a confidence interval between different pairs of images allows us to choose the pair that has the greatest mass fraction in a fixed-size confidence interval, i.e. leads to the greatest confidence in capturing the true epipole to within a fixed angle. In our experiment, we used a video sequence with a camera undergoing sideways translation relative to the scene. We considered all the image pairs that include the first image (frame 0), leaving the second image frame for selection. Figure 6 shows the resulting fractions of the probability mass for each frame. The peak is located at a reasonable baseline spanning four frames. The sharp decline in mass after frame 7 is caused by falling below an acceptable number of inlier point correspondences.

7 . CONCLUSION

We have presented a framework for epipolar geometry estimation that draws upon both multiple view geometry and statistics. The central theme is to derive a representation that faithfully represents the posterior likelihood globally. This is accomplished with a representation parameterized by epipole location in the first image. We have explored

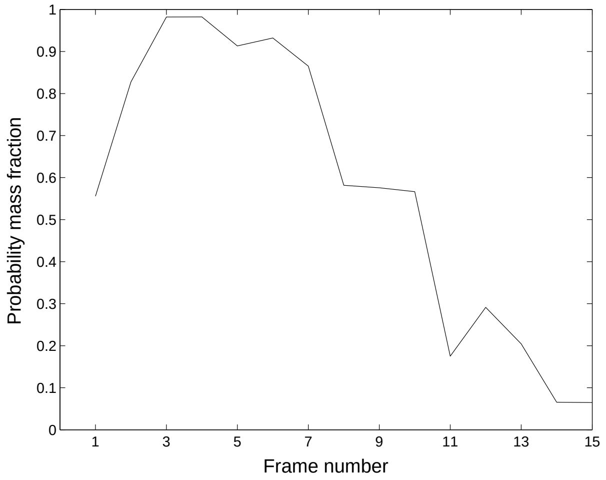


Figure 6: Probability mass lying within confidence interval over a series of video frames.

the efficiency of various fully and partially data-driven hypothesis generators in deriving the representation. We have presented experiments with confidence regions derived from our representation and we have experimentally validated the confidence regions through experiments with synthetic data. This was done by investigating the distribution of the confidence level needed to capture the true epipole in the confidence region, which should ideally be a uniformly distributed random variable. Finally, we have shown on real data how the uncertainty representation helps us accomplish inference tasks that are otherwise difficult, such as selecting which baseline to use when initializing automatic reconstruction from a video-sequence.

References

[1] P. Baker, R. Pless, C. Fermüller and Y. Aloimonos. Eyes from Eyes. *SMILE 00*, Springer-Verlag, p.204-217, 2001.

[2] P. Chang, M. Hebert. Robust tracking and structure from motion through sampling based uncertainty representation. *ICRA*, 2002.

[3] A. Chiuso, R. Brockett, S. Soatto. Optimal Structure from Motion: Local Ambiguities and Global Estimates. *IJCV*, vol. 39 n.3, p.195-228, Sept./Oct. 2000.

[4] J. Clark and A. Yuille. *Data Fusion for Sensory Information Processing Systems*. Kluwer Academic Publishers, ISBN 0-7923-9120-9, 1990.

[5] A. Doucet, N. de Freitas, N. Gordon, eds. Sequential Monte Carlo Methods In Practice. Springer-Verlag New York, 2001.

[6] O. Faugeras. Three-Dimensional Computer Vision. MIT Press, 1993.

[7] M. Fischler and R. Bolles. Random Sample Consensus: a Paradigm for Model Fitting with Application to Image Analysis and Automated Cartography. *Commun. Assoc. Comp. Mach.*, 24:381-395, 1981.

[8] W. Förstner. Uncertainty and Projective Geometry. To appear in: *Handbook of Computational Geometry for Pattern Recognition, Computer Vision, Neurocomputing and Robotics*

[9] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN 0-521-62304-9, 2000.

[10] B.K.P. Horn. Relative Orientation. *IJCV*, vol. 4, pp. 59-78, 1990.

[11] M. Irani, B. Rousso, and S. Peleg. Recovery of Ego-motion Using Region Alignment. *PAMI*, 19(3):268-272, March 1997.

[12] M. Isard and A. Blake. Condensation: conditional density propogation for visual tracking. IJCV, (1), 1998.

[13] R. Kaucic, R. Hartley, N. Dano. Plane-based Projective Reconstruction. *ICCV*, 2001.

[14] R. Kumar, P. Anandan, and K. Hanna. Shape Recovery from Multiple Views: a Parallax Based Approach. *ARPA Image Understanding Workshop*, November 1994.

[15] Y. Ma, S. Soatto, J. Košecká, S. Sastry. An Invitation to 3-D Vision: From Images to Geometric Models. Springer-Verlag New York, 2004.

[16] D. Nistér. An Efficient Solution to the Five-Point Relative Pose Problem. *PAMI*, 26(6):756-770, June 2004.

[17] D. Nistér and F. Schaffalitzky, What do Four Points in Two Calibrated Images Tell Us About the Epipoles?, *ECCV*, Springer Lecture Notes on Computer Science 3022:41-57, 2004.

[18] J. Oliensis. The Least-Squares Error for Structure from Infinitesimal Motion. *ECCV*, May 2004.

[19] G. Qian, R. Chellappa. Structure from Motion Using Sequential Monte Carlo Methods. *ICCV*, 2001.

[20] J. Ó Ruanaidh, W. Fitzgerald. Numerical Bayesian Methods Applied to Signal Processing. Springer-Verlag New York, 1996.

[21] R. Szeliski and P.H.S. Torr. Geometrically Constrained Structure from Motion : Points on Planes. *SMILE98*, pages 171-186, 1998.

[22] P.H.S. Torr; C. Davidson. IMPSAC: Synthesis of Importance Sampling and Random Sample Consensus. *IPAMI*, October 2002.

[23] B. Triggs. Plane+Parallax, Tensors and Factorization. *ECCV*, 2000.

[24] B. Triggs, P. McLauchlan, R. Hartley and A. Fitzgibbon. Bundle Adjustment - a Modern Synthesis. *Springer Lecture Notes on Computer Science*, Springer Verlag, 1883:298-375, 2000.

[25] Z. Tu, S. Zhu, H. Shum Image Segmentation by Data Driven Markov Chain Monte Carlo. *ICCV*, 2001.

[26] T. Werner, Constraints on Five Points in Two Images, *CVPR*, Volume 2, pp. 203-208, 2003.

[27] Z. Zhang. Determining the Epipolar Geometry and its Uncertainty: A Review. *IJCV*, March 1998.

SESSION 2

PERFORMANCE EVALUATION

MEASURING COMPLETE GROUND-TRUTH DATA AND ERROR ESTIMATES FOR REAL VIDEO SEQUENCES, FOR PERFORMANCE EVALUATION OF TRACKING, CAMERA POSE AND MOTION ESTIMATION ALGORITHMS

R. Stolkin[a] *, A. Greig [b], J. Gilby [c]

[a] Center for Maritime Systems, Stevens Institute of Technology, Hoboken, NJ 07030 USA – RStolkin@stevens.edu
[b] Dept. of Mechanical Engineering, University College London, WC1E 6BT UK – a_greig@meng.ucl.ac.uk
[c] Sira Ltd., South Hill, Kent BR7 5EH, UK – john.gilby@sira.co.uk

KEY WORDS: Vision, robotics, tracking, navigation, registration, calibration, performance, accuracy.

ABSTRACT:

Fundamental tasks in computer vision include determining the position, orientation and trajectory of a moving camera relative to an observed object or scene. Many such visual tracking algorithms have been proposed in the computer vision, artificial intelligence and robotics literature over the past 30 years. Predominantly, these remain un-validated since the ground-truth camera positions and orientations at each frame in a video sequence are not available for comparison with the outputs of the proposed vision systems.
A method is presented for generating real visual test data with complete underlying ground-truth. The method enables the production of long video sequences, filmed along complicated six degree of freedom trajectories, featuring a variety of objects, in a variety of different visibility conditions, for which complete ground-truth data is known including the camera position and orientation at every image frame, intrinsic camera calibration data, a lens distortion model and models of the viewed objects. We also present a means of estimating the errors in the ground-truth data and plot these errors for various experiments with synthetic data. Real video sequences and associated ground-truth data will be made available to the public as part of a web based library of data sets.

1. INTRODUCTION

An important and prolific area of computer vision research is the development of visual tracking and pose estimation algorithms. Typically these fit a model to features extracted from an observed image of an object to recover camera pose, track the position and orientation of a moving camera relative to an observed object or track the trajectory of a moving object relative to a camera.

Clearly, proper validation of such algorithms necessitates test images and video sequences with known ground-truth data, including camera positions and orientations relative to the observed scene at each frame, which can be compared to the outputs of proposed algorithms in order to compute errors. Surprisingly, very few such data sets or methodologies for creating them are discussed in the literature, with reported vision systems often validated in ad hoc ways.

Many papers attempt to demonstrate the accuracy of tracking algorithms by superimposing, over the observed image, a projection of the tracked object based on the positions and orientations output by the algorithm. In fact it can be shown (Stolkin 2004) that even very close 2D visual matches of this kind can result from significantly erroneous 3D tracked positions. One reason for this is that certain combinations of small rotations and translations, either of cameras or observed objects in 3D space, often make little difference to the resulting 2D images. This is especially true for objects with limited features and simple geometry. Such errors can only be properly identified and quantified by means of test images with accompanying complete 3D ground-truth.

It is relatively simple to construct artificial image sequences, with pre-programmed ground-truth, using commonly available graphics software (e.g. POV-Ray for windows) and this is also common in the literature. However, although testing computer vision algorithms on synthetic scenes allows comparison of performance, it gives only a limited idea of how the algorithms will perform on real scenes. Real cameras and real visibility conditions result in many kinds of noise and image degradation, far more complicated than Gaussian noise or "salt and pepper" speckling and it is not trivial or obvious how to realistically synthesise real world noise in an artificial image (Rokita, 1997; Kaneda, 1991). This becomes even more difficult when the scene is not viewed through clear air but through mist, smoke or turbid water. Artificial scenes do not completely reproduce the detailed variation of objects, the multitude of complex lighting conditions and modes of image degradation encountered in the real world. Vision and image processing algorithms often seem to perform much better on artificial (or artificially degraded) images than on real images. The only true test of computer vision algorithms remains their performance on real data.

To this end, several researchers have attempted to combine real image data with some knowledge of ground-truth. Otte, 1994, describes the use of a robot arm to translate a camera at known speeds, generating real image sequences for the assessment of optical flow algorithms. The measured ground-truth data is limited to known optic flow fields rather than explicit camera positions and the camera is only translated. Rotational camera motion is not addressed. McCane, 2001, also describes image sequences with known ground-truth motion fields. The work is limited to simple 2D scenes containing planar polyhedral objects against a flat background. The technique involves laborious hand-labelling of features in each image and so only very short sequences are usable. Wunsch, 1996, uses a robot arm to position a camera in known poses relative to an observed object. Similarly, Sim, 1999, generates individual images from known camera positions using a camera mounted on a gantry robot. In the work of both Wunsch and Sim, ground-truth positions are only measured for individual still images as opposed to video sequences. Both authors appear to obtain camera positions from the robot controller. It is not clear if or how the positions of the camera (optical centre) were measured relative to the robot end-effector. Agapito, 2001, generates ground-truth image sequences using their "Yorick" stereo head/eye platform. The work is limited to providing rotational motion with only two degrees of freedom. Although data for angles of elevation and pan can be extracted from the motor encoders of the platform, these are not in relationship to a particular observed object. The translational position of the camera remains unknown. Maimone, 1996, discusses various approaches for quantifying the performance of stereo vision algorithms, including the use of both synthetic images and real images with various kinds of known ground-truth. Maimone does mention the use of an image of a calibration target to derive ground-truth for a corresponding image of a visually interesting scene, filmed from an identical camera position. However, the techniques are limited to the acquisition of individual, still images from fixed camera positions. The additional problems, of generating ground-truth for extended video sequences, filmed from a moving camera, are not addressed.

In contrast, our method enables the production of long video sequences, filmed along a six degree of freedom trajectory, featuring a variety of objects, in a variety of different visibility conditions, for which complete ground-truth data is known including the camera position and orientation at every image frame, intrinsic camera calibration data, a lens distortion model and models of the viewed objects.

2. METHOD

2.1 Apparatus and procedure

An industrial robot arm (six degree of freedom Unimation PUMA 560) is used to move a digital cam-corder (JVC GR-DV2000) along a highly repeatable trajectory. "Test sequences", (featuring various objects of interest in various different visibility and lighting conditions), and "calibration sequences" (featuring planar calibration targets in good visibility) are filmed along identical trajectories (figures 1, 2).

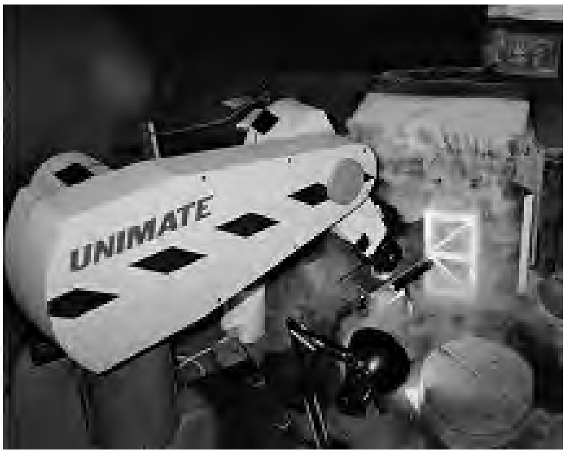


Figure 1. "Test sequence"-camera views a model oil-rig object in poor visibility.

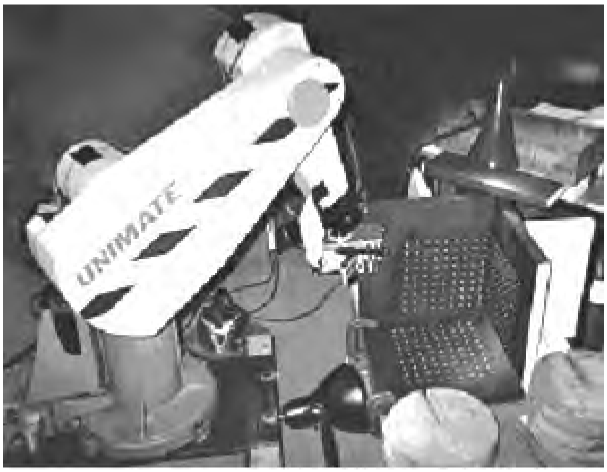


Figure 2. "Calibration sequence"-camera views calibration targets in good visibility.

A complete camera model, lens distortion model, and camera position and orientation can be extracted from the calibration sequence for every frame, by making use of the relationship between known world co-ordinates and measured image co-ordinates of calibration features. This information is used to provide ground-truth for chronologically corresponding frames in the visually interesting test sequences. Objects to be observed are measured, modeled and located precisely in the co-ordinate system of one of the calibration targets.

For those researchers interested in vision in poor visibility conditions (e.g. Stolkin 2000) dry ice fog can be used during the "test" sequences (figure 1) in addition to various lighting conditions (e.g. fixed lighting or spot-lights mounted on and moving with the camera).

Note, it is not feasible to extract camera positions from the robot control system since the position of the camera relative to the terminal link of the robot remains unknown; industrial robots, while highly *repeatable*, are not *accurate*; chronologically matching a series of robot positions to a series of images may be problematic.

2.2 Synchronisation

The "calibration" and "test" sequences are synchronised by beginning each camera motion with a view of an extra "synchronisation spot" feature (a white circular spot on black background). A frame from each sequence is found such that the "synchronisation spot" matches well when the two frames are superimposed. Thus the n^{th} frame from the matching frame in the test sequence is taken to have the same camera position as that measured for the n^{th} frame from the matching frame in the calibration sequence. The two sequences can only be synchronised to the nearest image frame (i.e. a worst case error of ±0.02 seconds at 25 frames per second). There are two ways of minimizing this error. Firstly, the camera is moved slowly so that temporal errors result in very small spatial errors. Secondly, many examples of each sequence are filmed, increasing the probability of finding a pair of sequences that match well (correct to the nearest pixel). If ten examples of each sequence are filmed, then the expected error is reduced by a factor of 100.

2.3 Feature extraction and labelling

The calibration targets are black planes containing square grids of white circular spots. The planes are arranged so that at least one is always in view and so that they are not co-planar. The positions of spots in images are determined by detecting the spots as "blobs" and then computing the blob centroid. A small number (at least 4) of spots in each of a few images scattered through the video sequence are then hand-labeled with their corresponding target plane co-ordinates. The remaining spots in all images are labeled by an automated process. The initial four labels are used to estimate the homography mapping between the target plane and the image plane. This homography is then used to project all possible target spots into the image plane. Any detected spots in the image are then assigned the labels of the closest matching projected spots. Spots in chronologically adjacent images are now labeled by assigning them the labels of the nearest spots from the previous (already labeled) image. These two processes, of projection and propagation, are iterated backwards and forwards over the entire image sequence until no new spot labels are found.

2.4 Camera calibration and position measurement

Our calibration method is adapted from that of Zhang, 1998, which describes how to calibrate a camera using a few images of a planar calibration target. Related calibration work includes Tsai, 1987. The following is a condensed summary of our implementation of these ideas.

2.4.1 Homography between an image and a calibration target: Since the calibration targets are planar, the mapping between the (homogeneous) target co-ordinates of calibration features, $\mathbf{X}_t = [X_t \quad Y_t \quad 1]^T$, and their corresponding (homogeneous) image co-ordinates, $\mathbf{x}_i = [u \quad v \quad 1]^T$, must form a homography, expressible as a 3×3 matrix:

$$\mathbf{x}_i = \mathbf{H}\mathbf{X}_t = [\mathbf{h}_1 \quad \mathbf{h}_2 \quad \mathbf{h}_3]\mathbf{X}_t \qquad (1)$$

Thus each calibration feature, whose position in an image is known and whose corresponding target co-ordinates have been identified, provides two constraints on the homography. A large number of such feature correspondences provides a large number of simultaneous equations:

$$\begin{bmatrix} w_1u_1 & w_2u_2 & . & . & w_nu_n \\ w_1v_1 & w_2v_2 & . & . & w_nv_n \\ w_1 & w_2 & . & . & w_n \end{bmatrix} = \mathbf{H} \begin{bmatrix} X_1 & X_2 & . & . & X_n \\ Y_1 & Y_2 & . & . & Y_n \\ 1 & 1 & . & . & 1 \end{bmatrix} \qquad (2)$$

A least squares fit homography is then found using singular value decomposition.

2.4.2 Constraints on the camera calibration parameters: The mapping between the target and image planes must also be defined by the intrinsic and extrinsic camera calibration parameters of the camera:

$$\mathbf{x}_i = \mathbf{H}\mathbf{X}_t = \mathbf{C}\mathbf{E}\mathbf{X}_t \qquad (3)$$

where **C** is the "intrinsic" or "calibration matrix":

$$\mathbf{C} = \begin{bmatrix} fk_u & 0 & u_0 \\ 0 & fk_v & v_0 \\ 0 & 0 & 1 \end{bmatrix}$$

(f is focal length, k_u and k_v are pixels per unit length in the u and v directions, (u_0, v_0) are the co-ordinates of the principal point, pixel array assumed to be square) and **E** is the "extrinsics matrix" defining the position and orientation of the camera (relative to the target co-ordinate system), i.e. $\mathbf{E} = [\mathbf{r}_1 \quad \mathbf{r}_2 \quad \mathbf{T}]$, where **r** and **T** denote rotation and translation vectors. Note that only two rotation vectors (not three) are needed since the calibration target plane is defined to lie at $Z = 0$ in the target co-ordinate system. Hence:

$$\mathbf{H} = [\mathbf{h}_1 \quad \mathbf{h}_2 \quad \mathbf{h}_3] = \mathbf{C}[\mathbf{r}_1 \quad \mathbf{r}_2 \quad \mathbf{T}] \qquad (4)$$

Since the column vectors of a rotation matrix are always mutually orthonormal, we have:

$$\mathbf{r}_1^T\mathbf{r}_2 = 0 \qquad (5)$$

$$\mathbf{r}_1^T\mathbf{r}_1 = \mathbf{r}_2^T\mathbf{r}_2 \qquad (6)$$

Since $\mathbf{r}_n = \mathbf{C}^{-1}\mathbf{h}_n$ these become:

$$\mathbf{h}_1^T\mathbf{C}^{-T}\mathbf{C}^{-1}\mathbf{h}_2 = 0 \qquad (7)$$

and $$\mathbf{h}_1^T\mathbf{C}^{-T}\mathbf{C}^{-1}\mathbf{h}_1 = \mathbf{h}_2^T\mathbf{C}^{-T}\mathbf{C}^{-1}\mathbf{h}_2 \qquad (8)$$

Thus one homography provides two constraints on the intrinsic parameters. Ideally, many homographies (from multiple images of calibration targets) are used and a least squares fit solution for the intrinsic parameters is found using singular value decomposition.

Once the intrinsic parameters have been found using a few different views of a calibration target, the extrinsic parameters can be extracted from any other single homography, i.e. the camera position and orientation can be extracted for any single image frame provided that it features several spots from at least one target.

2.4.3 Locating targets relative to each other: We use multiple calibration targets to ensure that at least one target is always in view during complicated (six degree-of-freedom) camera trajectories. Provided that at least one target is visible to the camera at each frame, the position of the camera can be computed by choosing one target to hold the world co-ordinate system and knowing the transformations which relate this target to the others. The relationship between any two targets is determined from images which feature both targets together, by determining the homography which maps between the co-ordinate systems of each target. For two targets, A and B:

$$\mathbf{x}_i = \mathbf{H}_A\mathbf{X}_A = \mathbf{H}_B\mathbf{X}_B \qquad (9)$$

where $\mathbf{X}_A$ and $\mathbf{X}_B$ are the positions of a single point in the respective co-ordinate system of each target. Thus:

$$\mathbf{X}_A = (\mathbf{H}_A)^{-1}\mathbf{x}_i = (\mathbf{H}_A)^{-1}\mathbf{H}_B\mathbf{X}_B \qquad (10)$$

2.4.4 Modeling lens distortion: Lens distortion is modelled as a radial shift of the undistorted pixel location (u, v) to the distorted pixel location $(\hat{u}, \hat{v})$, such that:

$$\hat{u} = u + (u - u_0)(k_1r^2 + k_2r^4) \qquad (11)$$

and $$\hat{v} = v + (v - v_0)(k_1r^2 + k_2r^4) \qquad (12)$$

where $$r^2 = (u - u_0)^2 + (v - v_0)^2$$

2.4.5 Refining parameter measurements with non-linear optimization: In practice, all important parameter measurements (camera intrinsics, lens distortion, target to target transformations, camera positions), which are initially extracted using the geometrical and analytical principles outlined above, can be further improved using non-linear optimisation. An error function is minimised, consisting of the sum of the squared distances (in pixels) between the observed image locations of calibration features and the locations predicted given the current estimate of the parameters being refined. This results in a maximum likelihood estimate for all parameters.

Firstly a small set (about 20) of images are used to compute camera intrinsic parameters, lens distortion parameters, camera position and orientation for each image (of the small set) and the transformations between the co-ordinate systems of each target. These parameters are then mutually refined over all views of all targets present in all images of the set, by minimising the following error function:

$$\sum_{\text{target } t=1}^{n} \sum_{\text{spot } s=1}^{m} \left\| \mathbf{x}_{image_{ts}} - \hat{\mathbf{x}}_{image_{ts}}\left(\mathbf{C}, k_1, k_2, \mathbf{R}_t, \mathbf{T}_t, \mathbf{X}_{target_{ts}}\right) \right\|^2 \qquad (13)$$

Where, for m points (spot centres) extracted from n target views, $\mathbf{x}_{image_{ts}}$ is the observed image in pixelated camera co-ordinates of the world co-ordinate target point $\mathbf{X}_{target_{ts}}$, and $\hat{\mathbf{x}}_{image_{ts}}$ is the expected image of that point given the current estimates of the camera parameters $(\mathbf{C}, k_1, k_2, \mathbf{R}_t, \mathbf{T}_t)$. Note that the values of the co-ordinates of $\mathbf{X}_{target_{ts}}$ are also dependent on the current estimates of target-to-target transformations and these transformations are also being iteratively refined.

Secondly, using the refined values for intrinsics, lens distortion parameters and target-to-target transformations, the camera position and orientation is computed for a single image taken from the middle of the "calibration sequence", again using analytical and geometrical principles. Keeping all other parameters constant, the six-degrees of freedom of this camera location are now non-linearly optimized, minimizing the error between the observed calibration feature locations and those predicted given the current estimate of the camera location and the fixed values (previously refined) of all other parameters.

Lastly, the camera position for the above single image is used as an initial estimate for the camera positions in chronologically adjacent images (previous and subsequent images) in the video sequence. These positions are then themselves optimized, the refined camera positions then being propagated as initial estimates for successive frames, and so on throughout the entire video sequence, resulting in optimized camera positions for every image frame along the entire camera trajectory.

3. RESULTS

3.1 Constructed data sets

We have filmed video sequences of around 1000 frames (at 25 frames per second) along a complicated six degree-of-freedom camera trajectory. Figure 3 shows the camera position at each frame, as calculated from the calibration sequence. The trajectory is illustrated in relation to the spots of the three calibration targets (30mm spacing between spots).

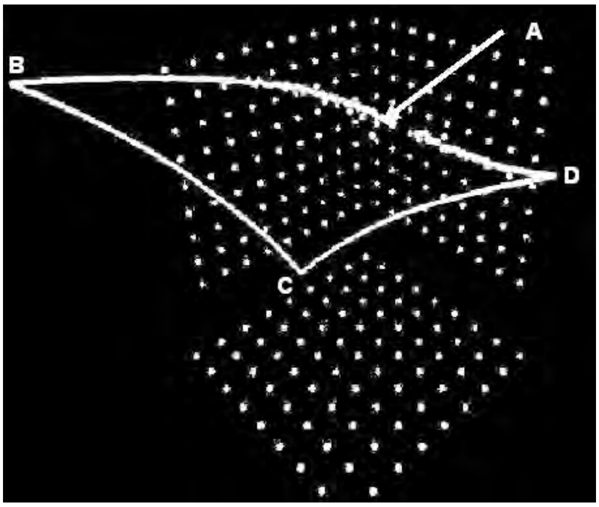


Figure 3. The computed trajectory for a six-degree of freedom of motion video sequence.

The sequences feature various different known (measured and modelled) objects (figure 4) in various different visibility and lighting conditions as well as a corresponding calibration sequence. Analysis of the calibration sequence has yielded a complete camera model, lens distortion model and a camera position and orientation for every frame in each of these sequences.

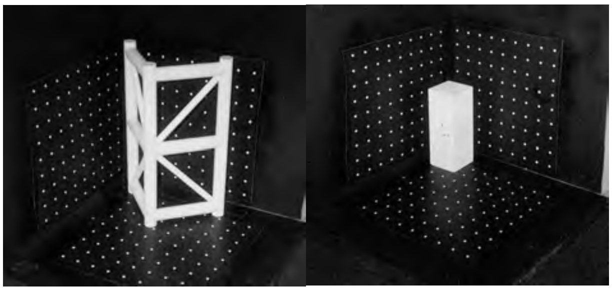
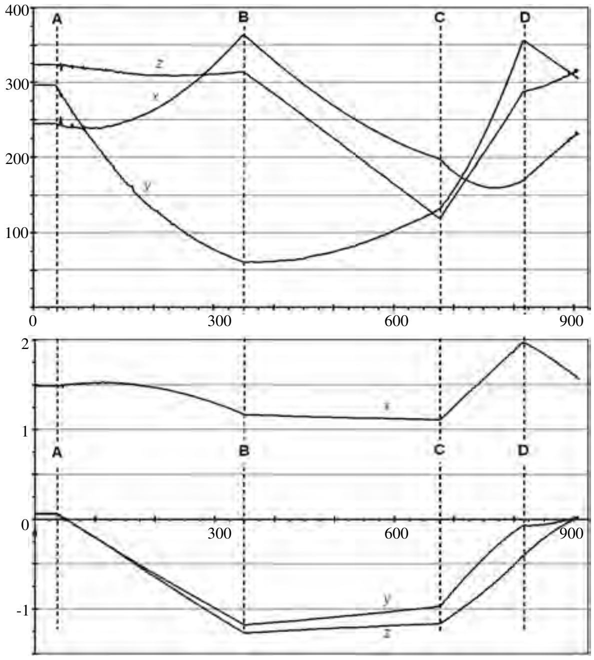


Figure 4. Two of the objects filmed in the video sequences, block and model oil-rig.

3.2 Smoothness of trajectory

One indicator of accuracy is the smoothness of the measured trajectory. Figure 3 is a useful visual representation of the trajectory and figures 5 and 6 are plots of the translational and rotational camera co-ordinates at each frame. Points A, B, C, D are corresponding way mark points between figures 3, 5 and 6.

For about the first 40 frames, the camera is stationary at point A. It will be noticed that small sections of the trajectory appear somewhat broken and erratic, approximately frames 40 – 160 and 880 – 910. These ranges correspond to the beginning and end of the trajectory during which the camera is moved from (and back towards) a position fixated on the "synchronization spot" (see section 2.2) at point A. During these periods, comparatively few calibration features are in the field of view. These sections of the video sequence do not correspond to visually interesting portions of the image sequence and are not used for testing vision algorithms. They are included only for synchronization. The remainder of the measured trajectory is extremely smooth, implying a high degree of precision. The robot is old, and its dynamic performance less than perfect, so the disturbance just after motion is initiated (shortly after point A) is probably due to the inertia of the system. Second and third peaks of decaying magnitude at exactly 20 and 40 frames later suggest that they have a mechanical origin.



Figures 5 & 6. Top graph shows translational components of camera motion along x, y and z axes. Vertical scale in mm. Bottom graph shows rotational components of camera motion about x, y and z axes. Vertical scale in radians. For both graphs, the horizontal scale is image frame number.

3.3 Robot repeatability

In order to assess repeatability, the robot was moved along a varied, six-degree of freedom motion that included pauses at three different positions during the motion. Several video sequences were filmed from the robot-mounted camera while moving in this fashion. Images from different sequences, filmed from the same pause positions, were compared. Superimposing the images reveals an error of better than ± one pixel. This implies that errors in image repeatability due to robot error approach the scale of the noise associated with the camera itself. Our robot is approximately twenty years old. Modern machines should produce even smaller errors.

3.4 Accuracy of scene reconstruction

In order to assess accuracy, the image positions of calibration features were reconstructed by projecting their known world co-ordinate positions through the measured camera model placed at the measured camera positions. Comparing these predicted image feature positions with those observed in the real calibration sequence yielded an rms error of 0.6 pixels per calibration feature (spot).

When some of the observed objects have been reconstructed in the same way, the errors are worse. Figure 7 shows an image from a sequence featuring a white block object. The measured camera position for the image frame has been used to project a predicted image (shown as a wire frame model) and this predicted image has been superimposed over the real image. This helps illustrate the errors involved (in this case ± 3 pixels discrepancy in block edges). This disparity in error magnitude (compared to 0.6 pixels above) may be due to over-fitting of the camera model to features in the calibration target planes and under-fitting to points outside those planes.

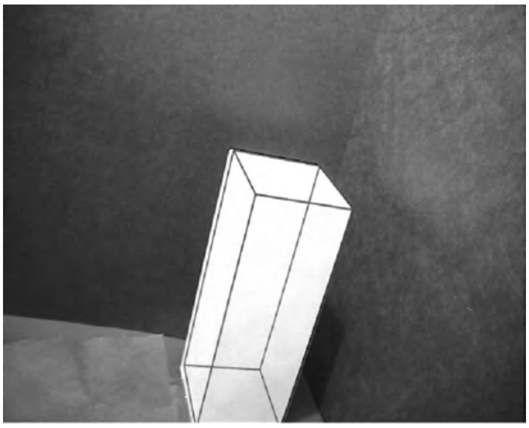


Figure 7. An image from a sequence featuring a block object. The superimposed wire frame image corresponds to the predicted image given the measured camera co-ordinates.

3.5 Accuracy of camera pose measurement

In order to estimate the potential overall accuracy of measured camera positions, we have used synthetic calibration data. Although, in general, synthetic images do not reproduce the noise inherent in real images, calibration sequences are filmed in highly controlled conditions which are more reasonably approximated by synthetic images. Graphics software (POV-Ray for windows) was used to generate computer models of calibration targets. A series of synthetic images were then rendered which would correspond to those generated by a camera viewing the targets from various positions. These images were fed into the calibration scheme. Ground-truth as measured by our calibration scheme was then compared with the pre-programmed synthetic ground-truth in order to quantify accuracy. For simplicity, we have used a synthetic camera array of 1000 by 1000 pixels-somewhat better than current typical real digital video resolution but far worse than typical real single image resolution. Over a set of 6 images filmed from several different ranges, but all featuring views of three approximately orthogonal calibration targets (see second paragraph of section 4), the error in measured principal point position was 1. 76 pixels and the error in measured focal length was 0.06%. The average error in measured camera position was 1.38mm and 0.024 degrees.

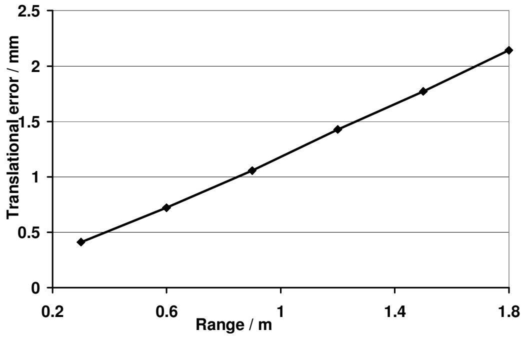
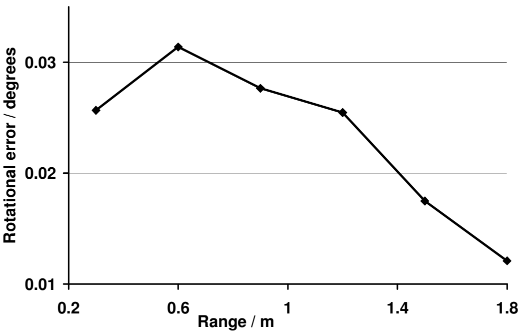


Figure 8. Variation in translational camera position error with range from calibration targets.



Figures 9. Variation in camera orientation error with range from calibration targets.

Figures 8 and 9 plot the variation of error with distance of the camera from the calibration target origin.

4. SUGGESTED IMPROVEMENTS

The problem, outlined in section 3.4, of over-fitting the camera model to points lying in the calibration target planes should be avoided in future work by using calibration images filmed at a variety of different ranges from the calibration targets.

Although it should be possible to determine the position of a calibrated camera given a view of a single calibration target (Zhang, 1998), in practice various small coupled translations and rotations of the camera can result in very similar views, causing measurement uncertainty. These errors can be constrained by ensuring that, throughout the motion of the camera, all three targets, positioned approximately orthogonally to each other, are always in view. In our original experiments with real video sequences, only one or two targets were viewed in most images and so our camera position accuracies are worse than can be achieved. Future researchers should ensure that the camera can always view three, approximately orthogonal, calibration targets in every image.

It is possible to further automate the labeling of calibration spots. By making a specific point, or points, on each target a different colour, it may be possible to eliminate the need to hand-label a small number of spots in each video sequence.

Viewing the "synchronization spot" after the cam-era has already started moving would eliminate the mechanical vibration problems of the step response noted at the start of the robot's motion.

The synchronisation problem (see section 2.2), that two sequences can only be synchronised to the nearest image frame (i.e. worst case error of ±0.02 seconds at 25 frames per second), might be eliminated by triggering the camera externally with a signal from the robot controller such that video sequences started at a specific location in the trajectory.

Note that test sequences can be filmed which feature virtually any kind of object. Even deforming or moving objects could conceivably be used although measuring ground-truth for the shapes and positions of such objects would pose additional challenges. Specifically, the use of objects with known textures might benefit researchers with an interest in surface reconstruction or optic flow. With appropriate equipment, it should also be possible to create real underwater sequences using our technique.

5. CONCLUSION

The field of computer vision sees the frequent publication of many novel algorithms, with comparatively little emphasis placed on their validation and comparison. If vision researchers are to conform to the rigorous standards of measurement, taken for granted in other scientific disciplines, it is important that our community evolve methods by which the performance of our techniques can be systematically evaluated using real data. Our method provides an important tool which enables the accuracy of many proposed vision algorithms, for registration, tracking and navigation, to be explicitly quantified.

REFERENCES

Agapito, L., Hayman, E., Reid, I., 2001. Self-Calibration of Rotating and Zooming Cameras. International Journal of Computer Vision. Vol. 45(2), pages 107-127.

Kaneda, K., Okamoto, T., Nakamae, E., Nishita, T., 1991. Photorealistic image synthesis for outdoor scenery. The Visual Computer. Vol. 7, pages 247-258.

Maimone, M., Shafer, S., 1996. A taxonomy for stereo computer vision experiments. ECCV workshop on performance characteristics of vision algorithms. Pages 59-79.

McCane, B., Novins, K., Crannitch, D., Galvin, B., 2001. On Benchmarking Optical Flow. Computer Vision and Image Understanding. Vol. 84, pages 126-143.

Otte, M., Nagel, H., 1994. Optical Flow estimation: Advances and Comparisons. Proc. 3rd European Conference on Computer Vision. Pages 51-60.

POV-Ray for windows, http://www.povray.org.

Rokita, P., 1997.Simulating Poor Visibility Conditions Using Image Processing. Real-Time Imaging, 3, pages 275-281.

Sim, R., Dudek, G., 1999. Learning and Evaluating Visual Features for Pose Estimation. International Conference on Computer Vision. Vol. 2.

Stolkin, R, 2004. Combining observed and predicted data for robot vision in poor visibility. PhD thesis, Department of Mechanical Engineering, University College London.

Stolkin, R., Hodgetts, M., Greig, A., 2000. An EM/E-MRF Strategy for Underwater Navigation. Proc.11th British Machine Vision Conference.

Tsai, R. A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf tv camera and lenses. IEEE Journal of Robotics and Automation. Vol. 3(4), pages 323-344. 1987.

Wunsch, P., Hirzinger, G., 1996. Registration of CAD-Models to Images by Iterative Inverse Perspective Matching. Proceedings of the 13th International Conference on Pattern Recognition. Pages 77-83.

Zhang, Z., 1998. A Flexible New Technique for Camera Calibration. Microsoft Research Technical Report, MSR-TR-98-71.

PERFORMANCE EVALUATION OF A LOCALIZATION SYSTEM RELYING ON MONOCULAR VISION AND NATURAL LANDMARKS

Eric Royer[†][*], Maxime Lhuillier[†] , Michel Dhome[†], Jean-Marc Lavest[†]

[†]LASMEA UMR6602 CNRS et Université Blaise Pascal, 24 avenue des Landais - 63177 AUBIERE Cedex
Eric.ROYER@lasmea.univ-bpclermont.fr — http://www.lasmea.univ-bpclermont.fr/Personnel/Eric.Royer/

KEY WORDS: Vision, Reconstruction, Performance, Real-time, Navigation

ABSTRACT:

We present a real-time localization system based on monocular vision and natural landmarks. In a learning step, we record a reference video sequence and we use a structure from motion algorithm to build a model of the environment. Then in the localization step, we use this model to establish correspondences between the 3D model and 2D points detected in the current image. These correspondences allow us to compute the current camera localization in real-time. The main topic of this paper is the performance evaluation of the whole system. Four aspects of performance are considered : versatility, accuracy, robustness and speed.

1 INTRODUCTION

In this paper we evaluate the performance of an algorithm designed to compute the localization of a camera in real-time. Only one camera and natural landmarks are required. In a first step, we record a video sequence along a trajectory. Then this sequence goes through a structure from motion algorithm to compute a sparse 3D model of the environment. When this model has been computed, we can use it to compute the localization of the camera in real-time as long as the camera stays in the neighborhood of the reference trajectory. We have developed this system for outdoor autonomous navigation of a robotic vehicle, but other applications such as indoor robotics or augmented reality can use the same localization system. The main topic of the paper is the performance evaluation of the localization system. The algorithm is only briefly presented here, more details can be found in (Royer et al., 2005).

As soon as a map of the environment is available, it is possible to compute a localization for the camera with reference to the map. Several approaches for building the map are possible. Simultaneous Localization And Mapping (SLAM) is very attractive because localization is possible as soon as the system starts working. But map building is the most computer intensive part, so doing this with monocular vision in real-time is difficult. However, monocular SLAM has been achieved in real-time (Davison, 2003). But the main drawback is that it's not possible to handle a large number of landmarks in the database. Computing a localization from the video flow can also be done by ego-motion estimation or visual odometry (Nistér et al., 2004). But this method is subject to error accumulation because there is no global optimization and the localization accuracy decreases with the distance covered.

Another possible approach is to build a map first and use this map to localize the camera. The main advantage is that there is no real-time constraint on map building. So algorithms providing more accuracy can be used. This approach has been used several times for robot localization. Cobzas et al. (2003) use a camera mounted on a rotating platform and a laser range finder to build a panoramic image enhanced with 3D data of the environment. After the 3D model is built, a single 2D image is enough to compute the localization of the camera. Kidono et al. (2002) also use a map building step before the localization. Map building consists in recording the video sequence along a reference trajectory, then localization is possible in the neighborhood of this trajectory as in our method. It works under the assumption that the ground is planar and the sensors used are a stereo vision rig and an odometer. In our case, the ground can be irregular and we use only one calibrated camera. Camera calibration is important in order to use fish eye lenses with up to 130° field of view. Map building is done with a structure from motion algorithm.

In section 2 we briefly present the algorithms we use to build the map from the reference video sequence, and how this map is used for the localization process. In section 3 we show some localization results and we discuss the performance of the system. Four aspects of performance are considered : versatility, accuracy, robustness and speed. The results come from experiments carried out indoors and outdoors. The results provided by the vision algorithm are compared to the ground truth whenever possible.

2 ALGORITHM

2.1 Map building

Every step in the reconstruction as well as the localization relies on image matching. Interest points are detected in each image with Harris corner detector (Harris and Stephens, 1988). For each interest point in image 1, we select some candidate corresponding points in a rectangular search region in image 2. Then a Zero Normalized Cross Correlation score is computed between their neighborhoods, and the pairs with the best scores are kept to provide a list of corresponding point pairs between the two images. This matching method is sufficient when the camera doesn't rotate much around the optical axis which is the case when the camera is mounted on a wheeled robot. Matching methods with rotational invariance might be used depending on the application but they would require more computing power.

The goal of the reconstruction is to obtain the position of a subset of the cameras in the reference sequence as well as a set of landmarks and their 3D location in a global coordinate system. The structure from motion problem has been studied for several years and multiple algorithms have been proposed depending on the assumptions we can make (Hartley and Zisserman, 2000). For our experiments, the camera was calibrated using a planar calibration

[*]Corresponding author.

pattern (Lavest et al., 1998). Camera calibration is important because the wide angle lens we use has a strong radial distortion. With a calibrated camera, the structure from motion algorithm is more robust and the accuracy of the reconstruction is increased. In our robotic application, the motion is mostly along the optical axis of the camera. Point triangulation must be done with small angles, which increases the difficulty of obtaining an accurate 3D reconstruction.

In the first step of the reconstruction, we extract a set of key frames from the reference sequence. Then we compute camera motion between key frames. Additionally, the interest points are reconstructed in 3D. These points will be the landmarks used for the localization process.

2.1.1 Key frame selection

If there is not enough camera motion between two frames, the computation of the epipolar geometry is an ill conditioned problem. So we select images so that there is as much camera motion as possible between key frames while still being able to match the images. The first image of the sequence is always selected as the first key frame I_1. The second key frame I_2 is chosen as far as possible from I_1 but with at least M common interest points between I_1 and I_2. When key frames $I_1 \dots I_n$ are chosen, we select I_{n+1} (as far as possible from I_n) so that there is at least M interest points in common between I_{n+1} and I_n and at least N common points between I_{n+1} and I_{n-1}. In our experiments we detect 1500 interest points per frame and we choose $M = 400$ and $N = 300$.

2.1.2 Camera motion computation

For the first three key frames, the computation of the camera motion is done with the method given by Nistér (2003) for three views. It involves computing the essential matrix between the first and last images of the triplet using a sample of 5 point correspondences. There are at most 10 solutions for E. Each matrix E gives 4 solutions for camera motion. The solutions for which at least one of the 5 points is not reconstructed in front of both cameras are discarded. Then the pose of the remaining camera is computed with 3 out of the 5 points in the sample. This process is done with a RANSAC (Fischler and Bolles, 1981) approach : each 5 point sample produces a number of hypothesis for the 3 cameras. The best one is chosen by computing the reprojection error over the 3 views for all the matched interest points and keeping the one with the higher number of inlier matches. We need an algorithm to compute the pose of the second camera. With a calibrated camera, three 3D points whose projections in the image are known are enough to compute the pose of the camera. Several methods are compared by Haralick et al. (1994). We chose Grunert's method with a RANSAC approach.

For the next image triplets, we use a different method for computing camera motion. Assume we know the location of cameras C_1 through C_N, we can compute camera C_{N+1} by using the location of cameras C_{N-1} and C_N and point correspondences over the image triplet $(N-1, N, N+1)$. We match a set of points X^i whose projections are known in each image of the triplet. From the projections in images $N-1$ and N, we can compute the 3D coordinates of point X^i. Then from the set of X^i and their projections in image $N+1$, we use Grunert's calibrated pose estimation algorithm to compute the location of camera C_{N+1}. In addition the 3D locations of the reconstructed interest points are stored because they will be the landmarks used for the localization process. The advantage of this iterative pose estimation process is that it can deal with virtually planar scenes. After the pose computation, a second matching step is done with the epipolar constraint based on the pose that has just been computed. This second matching step allows to increase the number of correctly reconstructed 3D points by about 20 %.

2.1.3 Hierarchical bundle adjustment

The computation of camera C_N depends on the results of the previous cameras and errors can build up over the sequence. In order to correct this problem, we use a bundle adjustment which provides a better solution. The bundle adjustment is a Levenberg-Marquardt minimization of the cost function $f(C_E^1, \cdots, C_E^N, X^1, \cdots, X^M)$ where C_E^i are the external parameters of camera i, and X^j are the world coordinates of point j. For this minimization, the radial distortion of the 2D point coordinates is corrected beforehand. The cost function is the sum of the reprojection errors of all the inlier reprojections in all the images :

$$f(C_E^1, \cdots, C_E^N, X^1, \cdots, X^M) = \sum_{i=1}^{N} \sum_{j=1, j \in J_i}^{M} d^2(x_i^j, P_i X^j)$$

where $d^2(x_i^j, P_i x^j)$ is the squared euclidian distance between $P_i X^j$ the projection of point X^j by camera i, and x_i^j is the corresponding detected point. P_i is the 3×4 projection matrix built from the parameters values in C_E^i and the known internal parameters of the camera. And J_i is the set of points whose reprojection error in image i is less than 2 pixels at the beginning of the minimization. After a few iteration steps, J_i is computed again and more minimization iterations are done. This inlier selection process is repeated as long as the number of inliers increases.

Computing all the camera locations and use the bundle adjustment only once on the whole sequence could cause problems because increasing errors could produce an initial solution too far from the optimal one for the bundle adjustment to converge. Thus it is necessary to use the bundle adjustment throughout the reconstruction of the sequence. So we use the adjustment hierarchically (Hartley and Zisserman, 2000). A large sequence is divided into two parts with an overlap of two frames in order to be able to merge the sequence. Each subsequence is recursively divided in the same way until each final subsequence contains only three images. Each image triplet is processed as described in section2.1.2. After each triplet has been computed we run a bundle adjustment over its three frames. Then we merge small subsequences into larger subsequences and we use a bundle adjustment after each merging operation. In order to merge two subsequences, we compute a best-fit rigid transformation so that the first two cameras of the second subsequence are transformed into the last two cameras of the first subsequence. Merging is done until the whole sequence has been reconstructed. The reconstruction ends with a global bundle adjustment. The number of points used in the bundle adjustment is on the order of several thousands.

2.2 Real-time localization

The output of the learning process is a 3D reconstruction of the scene : we have the pose of the camera for each key frame and a set of 3D points associated with their 2D positions in the key frames. At the start of the localization process, we have no assumption on the vehicle localization. So we need to compare the current image to every key frame to find the best match. This is done by matching interest points between the two images and computing a camera pose with RANSAC. The pose obtained with the higher number of inliers is a good estimation of the camera pose for the first image. This step requires a few seconds but is needed only at the start. After this step, we always have an approximate pose for the camera, so we only need to update the pose and this can be done much faster.

The current image is noted I. First we assume that the camera movement between two successive frames is small. So an approximate camera pose (we note the associated camera matrix

P_0) for image I is the same as the pose computed for the preceding image. Based on P_0 we select the closest key frame I_k in the sense of shortest euclidian distance between the camera centers. I_k gives us a set of interest points A_k reconstructed in 3D. We detect interest points in I and we match them with A_k. To do that, for each point in A_k, we compute a correlation score with all the interest points detected in I which are in the search region. For each interest point in A_k we know a 3D position, so with P_0 we can compute an expected position of this point in I. In the matching process the search region is centered around the expected position and its size is small (20 × 12 pixels). After this matching is done, we have a set of 2D points in image I matched with 2D points in image I_k which are themselves linked to a 3D point obtained during the reconstruction process. With these 3D/2D matches a better pose is computed using Grunert's method through RANSAC to reject outliers. This gives us the camera matrix P_1 for I. Then the pose is refined using the iterative method proposed by Araújo et al. (1998) with some modifications in order to deal with outliers. This is a minimization of the reprojection error for all the points using Newton's method. At each iteration we solve the linear system $J\delta = e$ in order to compute a vector of corrections δ to be subtracted from the pose parameters. e is the error vector formed with the reprojection error of each point in x and y. J is the Jacobian matrix of the error. In our implementation, the points used in the minimization process are computed at each iteration. We keep only the points whose reprojection error is less than 2 pixels. As the pose converges towards the optimal pose, some inliers can become outliers and conversely. Usually, less than five iterations are enough.

3 PERFORMANCE EVALUATION

3.1 Versatility

This localization system was used with several cameras in different kind of environments. We used normal and fish eye lenses with a field of view ranging from 50° to 130°. The localization system is performing well both indoors and outdoors with changing weather conditions (cloudy, sunny, or with snow on the ground) with a single learning sequence. According to the environment we used different methods to evaluate the accuracy and the robustness of the algorithm. The results of these experiments are detailed in the following paragraphs.

3.2 Accuracy

3.2.1 Indoor experiments To evaluate the accuracy of the localization we used a table where we could measure the position of the camera with a 1 millimeter accuracy in a 1.2 m × 1.0 m rectangle. We first recorded a reference video sequence on the left side of the table. The trajectory was a 1.2 m long straight line oriented along the optical axis of the camera (Z). Figure 1 illustrates the setup with two images taken on each side of the localization area (1 m apart). Another pair of such images is present on Figure 9. Most of the objects visible were along the wall of the room which was about 3.5 m in front of the localization area. There were 13 key frames and we built a 3D reconstruction from these images. Then we moved the camera by 10 cm increments in X or Z in the localization area in order to cover the whole rectangle. For each position we ran the localization algorithm and compared the position given by the vision algorithm to the true position measured on the table. This gave us 131 measurements: the position error $e_{i,j}$ was made for $X = 0.1i$ and $Z = 0.1j$ for each $(i, j) \in \{0..11\} \times \{0..10\}$. For each lateral deviation ($X = constant$) we computed the average value of the error and the standard deviation. The result is shown on Figure 2. As long as we stay on the reference trajectory, the localization error is only a few millimeters. The order of magnitude of the error depends on the distance of the observed 3D points. The outdoor experiments show a ten fold increase in localization error because the objects observed can be at 30 m rather than 3 m.

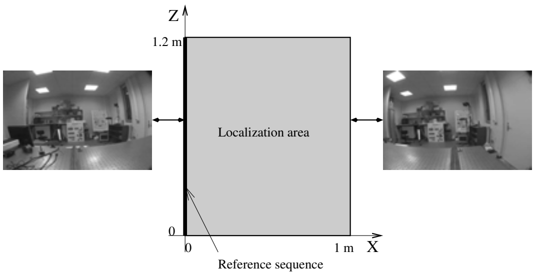


Figure 1: Setup for the indoor experiment

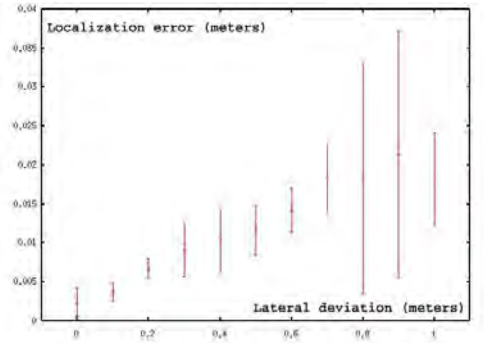


Figure 2: Localization error for a given lateral deviation (average value and standard deviation)

We also made an experiment to evaluate the rotational accuracy. The camera was mounted on a rotating platform. The angle of the platform can be read with about ±0.1° accuracy. We compared the orientation α provided by the vision algorithm to the angle α_0 given by the platform. We used the same fish eye lens as in the previous experiment, providing a 130° field of view (in the diagonal) and we made a measurement for each angle from $\alpha_0 = -94°$ to $\alpha_0 = 94°$ with a 2° increment. The reference trajectory was a straight line (1 m long) oriented along the optical axis (which was in the 0° direction). The result of this experiment appears on Figure 3. The algorithm was not able to provide the pose of the camera when the angle reached 95° because there were not enough point correspondences. The angular accuracy measured with this setup is about ±0.1°, which is about the same as what can be read on the platform. The algorithm provides a useful angular information for a deviation up to 94° on either side with this camera. Of course, with such an angular deviation from the reference frame, the part of the image which can be used is very small, and the localization becomes impossible if there is an occultation in this area. Images captured for 0°, 45° and 90° are shown on Figure 4.

3.2.2 Outdoor experiment For outdoor situations, the camera is mounted on the roof of a robotic vehicle along with a Differential GPS (DGPS) sensor to record the ground truth. According to the manufacturer, the DGPS has an accuracy of 1 cm in an horizontal plane (it is only 20 cm along a vertical axis with our hardware). Measuring the accuracy of our algorithms is not straightforward. Two operations are needed so that both data sets can be compared. First the GPS sensor is not mounted on the vehicle at the same place as the camera. The GPS is located at the

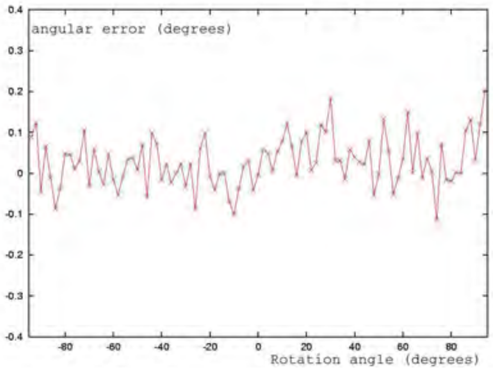


Figure 3: Angular error

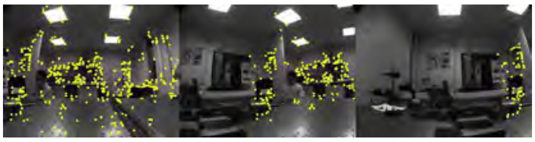


Figure 4: From left to right images taken at 0°, 45° and 90° orientation, with interest points correctly matched

mid-point between the rear wheels of the car, while the camera is between the front wheels. So the two sensors don't have the same trajectory. From the GPS positions, we computed a "virtual" GPS which indicates what a GPS would record if it was at the same place as the camera. In addition, the 3D reconstruction is done in an arbitrary euclidian coordinate system, whereas the GPS positions are given in another coordinate system. So the whole 3D reconstruction has to be transformed using a rotation, translation and scale change. The approach described by Faugeras and Herbert (1986) is used to compute this transformation. After these transformations have been made, for each camera we are able to compute the error on the position in meters. Because of the lack of accuracy of the DGPS along the vertical axis, all the localization errors reported for the outdoor experiments are measured in an horizontal plane only.

Four sequences called $outdoor_1$ through $outdoor_4$ were recorded by driving manually the vehicle along a 80 m trajectory. The four sequences were made approximately on the same trajectory (with at most a 1 m lateral deviation), the same day. Each sequence was used in turn as the reference sequence. So we made twelve experiments : we computed a localization for $outdoor_i$ using $outdoor_j$ as the reference sequence for each $j \in \{1, 2, 3, 4\}$ and $i \neq j$. A few images extracted from $outdoor_1$ are shown in Figure 5. The positions of the key frames computed from this sequence are shown in Figure 6 (as seen from the top) along with the trajectory recorded by the DGPS. Depending on the sequence, the automatic key frame selection gave between 113 and 121 key frames. And at the end of the reconstruction there were between 14323 and 15689 3D points.

We define two errors to measure the reconstruction and the localization accuracy. We want to distinguish the error that is attributed to the reconstruction algorithm and the error coming from the localization algorithm. The reconstruction error is the average distance between the camera positions obtained from the structure from motion algorithm and the true positions given by the DGPS (after the two trajectories have been expressed in the same coordinate system). The reconstruction error for each of the sequences was 25 cm, 40 cm, 34 cm and 24 cm for a 80 m long trajectory with two large turns. This error is mostly caused by a slow drift of the reconstruction process. It increases with the length and complexity of the trajectory. That means the 3D model we build is not perfectly matched to the real 3D world and computing a global localization from this model would give at least about 30 cm of error.

Figure 5: A few images from $outdoor_1$

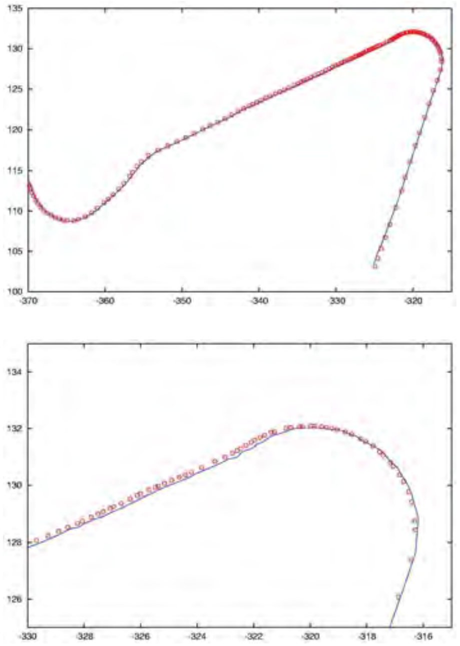


Figure 6: Position of the key frames (circles) with reference to the trajectory recorded by the DGPS (continuous line). Whole trajectory on top and close up view at the bottom (units in meters)

However, in many applications, a global localization is not required. For example, in our application a robot needs to compute a self-localization so that it is able to follow the reference trajectory. In this case, we only need to compute the distance between the current robot position and the reference trajectory as well as the angular deviation from the reference trajectory. A global localization is not necessary, only a relative position with respect to the reference trajectory is needed. We define the localization error in order to measure the error we make in computing this relative localization with the vision algorithm. We need a somewhat more complicated definition for the localization error. First we compute the lateral deviation between the current robot position and the closest robot position on the reference trajectory. This is illustrated on Figure 7. The robot position is always defined by the position of the middle point of the rear axle of the vehicle. This position is directly given by the DGPS. When working with vision it must be computed from the camera position and orientation. First we apply a global scale to the 3D reconstruction so that the scale is the same between the GPS data and vision data. We start with the localization of the camera C_1 given by the localization part of the vision algorithm. From C_1 we compute the corresponding GPS position G_1 (it is possible because we measured the positions of the GPS receiver and the camera on the vehicle). Then we find the closest GPS position in the reference

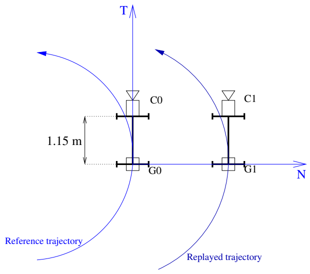


Figure 7: Computing the lateral deviation from the reference trajectory

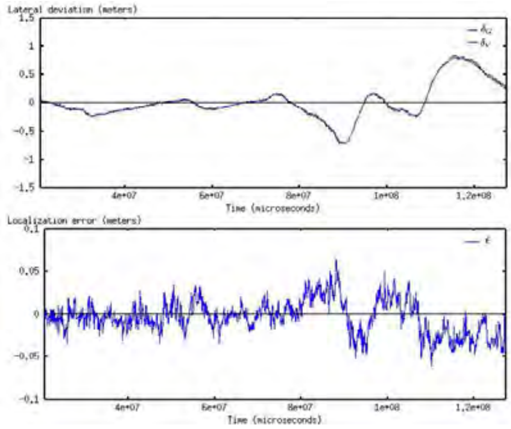


Figure 8: Lateral deviation (top) measured with the DGPS δ_G (blue) or with vision δ_V (red) and localization error ϵ (bottom)

trajectory : we call it G_0. At point G_0 of the reference trajectory, we compute the tangent $\overrightarrow{T}$ and normal $\overrightarrow{N}$ to the trajectory. The lateral deviation computed with vision is $\delta_V = \overrightarrow{G_0G_1} \cdot \overrightarrow{N}$. The lateral deviation is computed from the GPS measurements as well and we get δ_G (in this case we have directly G_0 and G_1). δ_G and δ_V are the same physical distance measured with two different sensors. Then the localization error is defined as $\epsilon = \delta_V - \delta_G$. From this we can compute the standard deviation of ϵ for a whole trajectory : we call this the average localization error.

We computed the average localization error for each of the twelve experiments : the smallest was 1.4 cm, the largest was 2.2 cm and the mean over the twelve videos was 1.9 cm. Figure 8 shows the lateral deviation and localization error for one experiment with a 1.9 cm average localization error. To make sure that it is a valid method to measure the localization accuracy, we used a control law to drive the robotic vehicle. We used in turn the GPS sensor and the vision algorithm to control the robot. Both methods allowed to drive the robot with the same accuracy (4 cm in straight lines and less than 35 cm lateral deviation in curves for both sensors). This shows that the accuracy of the GPS and the vision algorithm is equivalent for the autonomous navigation application. The error can be attributed more to the difficulty of controlling the robot than to the localization part.

3.3 Robustness

3.3.1 Indoor experiment We made two experiments to evaluate the robustness of the localization algorithm. First, we made no change to the environment between the reference sequence and the localization step, but up to 6 persons went in front on the camera to mask a part of the scene. In the second experiment, we started the localization process with the same environment as in the reference sequence and we gradually modified the scene. We moved or removed some objects, changed the illumination, and added some occultations. The modifications were made in 8 steps. For both experiments, we recorded the error between the computed localization and the true localization. We did this for two different camera positions : one on the reference sequence (on axis) and one for a position with 1 m lateral deviation from the reference trajectory (off axis). The reference trajectory was the same as in the indoor accuracy experiment. Figure 10 shows the closest key frame found and some of the images for which the localization was computed. Correctly identified interest points are also drawn. Figure 9 shows the images used in the off axis occultation experiment. The localization error is given in Table 1 for the occultation experiment and in Table 2 for the scene modification experiment. These results show that the algorithm is robust to large changes in the environment (modifications of the scene, occultations and changing light conditions). The reason is that we have a large number of features stored in the database and only a few of them are needed to compute an accurate localization. Moreover the constraints on feature matching are severe enough so that additional objects that are added to the scene are not taken erroneously as inliers. The performance degradation is visible only with a large lateral deviation and strong changes to the environment.

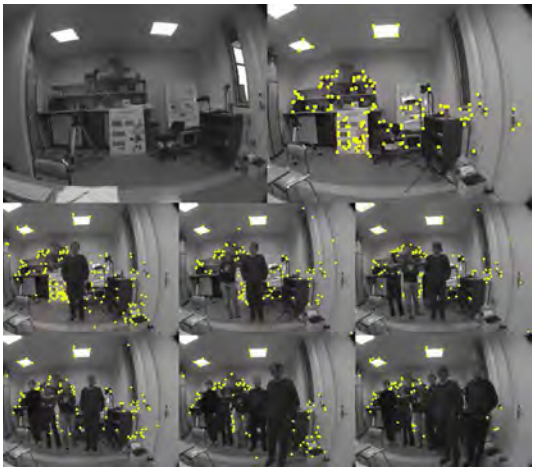


Figure 9: Images for the off-axis occultation experiment. Top left : reference image on axis, top right : off-axis image with no occultation. Second and third rows : occultation by 1 to 6 persons

Number of persons	0	1	2	3	4	5	6
position error on axis (mm)	2	1	1	1	1	1	2
position error off axis (mm)	8	11	4	11	20	44	132

Table 1: Localization error for the occultation experiment

3.3.2 Outdoor experiments For outdoors use, a localization system must be robust to changes in illumination and weather. Since the system was developed, we have had the opportunity to

Modification step	1	2	3	4	5	6	7	8
error on axis (mm)	1	1	2	0	2	5	2	5
error off axis (mm)	29	16	18	24	51	100	21	183

Table 2: Localization error for the scene modification experiment

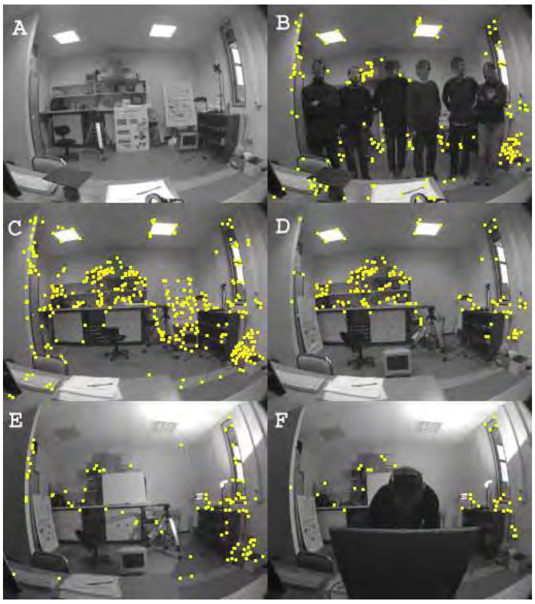


Figure 10: Images for robustness evaluation on axis : original image (A), occultation by 6 persons (B), modifications step 2 (C), step 4 (D), step 6 (E) and step 8 (F)

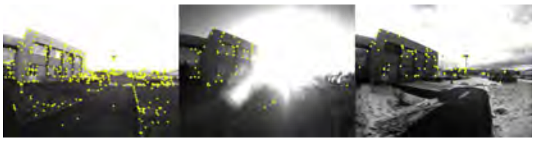


Figure 11: Localization robustness to weather changes

try it under different conditions. The robot was able to localize itself and to navigate autonomously in bright sunlight (even with the sun in the field of view of the camera) and with snow on the ground even if the reference sequence was recorded on a cloudy day without snow. Figure 11 shows the reference sequence on the left with all the interest points available in the database. Two images extracted from navigation experiments are shown on the right with the interest points correctly identified. The map building process is also robust to moving objects in the scene. We have been able to compute 3D reconstructions for sequences with up to 500 m long including pedestrians and moving vehicles (Royer et al., 2005).

3.4 Speed

The timings were made on a 3.4 GHz Pentium 4 processor with an image size of 640x480 pixels and 1500 interest points detected in each frame. The code uses the SSE2 instruction set for all the image processing. The reconstruction time for a sequence such as $outdoor_1$ is about 1 hour. The whole localization runs in 60 ms. Detecting interest points takes 35 ms, matching takes 15 ms and computing the pose takes 10 ms.

4 CONCLUSION

We have presented a localization algorithm and shown its performance under different conditions. It has been used both indoors and outdoors and with various cameras. The accuracy with reference to the learning trajectory is good enough for most robotic applications. Guidance applications based on this localization system have been successfully conducted outdoors with an accuracy similar to those obtained with a differential GPS sensor. The algorithm runs in real-time for the localization part. The approach proposed here works well for our intended application : that is driving a robot near the reference trajectory. For more complex navigation tasks either wide baseline matching techniques or a map with more keyframes from different viewing locations would be necessary. Future work will be more directed towards an improvement of robustness to changes in the environment. Even if the experiments presented in this paper have shown that the localization algorithm is robust to some changes, it may not be enough for an ever changing environment. For example in a city, cars parked along the side of the road change from day to day, trees evolve according to the season, some buildings are destroyed while others are built or modified. So our goal is to have a method to update the map automatically in order to take these modifications into account.

REFERENCES

Araújo, H., Carceroni, R., and Brown, C., 1998. A fully projective formulation to improve the accuracy of Lowe's pose estimation algorithm. *Computer Vision and Image Understanding*, 70(2):pp. 227–238.

Cobzas, D., Zhang, H., and Jagersand, M., 2003. Image-based localization with depth-enhanced image map. In *International Conference on Robotics and Automation*.

Davison, A. J., 2003. Real-time simultaneous localisation and mapping with a single camera. In *Proceedings of the 9th International Conference on Computer Vision, Nice*.

Faugeras, O. and Herbert, M., 1986. The representation, recognition, and locating of 3-d objects. *International Journal of Robotic Research*, 5(3):pp. 27–52.

Fischler, O. and Bolles, R., 1981. Random sample consensus: a paradigm for model fitting with application to image analysis and automated cartography. *Communications of the Association for Computing Machinery*, 24:pp. 381–395.

Haralick, R., Lee, C., Ottenberg, K., and Nolle, M., 1994. Review and analysis of solutions of the three point perspective pose estimation problem. *International Journal of Computer Vision*, 13(3):pp. 331–356.

Harris, C. and Stephens, M., 1988. A combined corner and edge detector. In *Alvey Vision Conference*, pp. 147–151.

Hartley, R. and Zisserman, A., 2000. *Multiple view geometry in computer vision*. Cambridge University Press.

Kidono, K., Miura, J., and Shirai, Y., 2002. Autonomous visual navigation of a mobile robot using a human-guided experience. *Robotics and Autonomous Systems*, 40(2-3):pp. 124–1332.

Lavest, J. M., Viala, M., and Dhome, M., 1998. Do we need an accurate calibration pattern to achieve a reliable camera calibration ? In *European Conference on Computer Vision*, pp. 158–174.

Nistér, D., 2003. An efficient solution to the five-point relative pose problem. In *Conference on Computer Vision and Pattern Recognition*, pp. 147–151.

Nistér, D., Naroditsky, O., and Bergen, J., 2004. Visual odometry. In *Conference on Computer Vision and Pattern Recognition*, pp. 652–659.

Royer, E., Lhuillier, M., Dhome, M., and Chateau, T., 2005. Localization in urban environments : monocular vision compared to a differential GPS sensor. In *Conference on Computer Vision and Pattern Recognition*.

ROBUST METRIC STRUCTURE FROM MOTION
FOR AN EXTENDED SEQUENCE WITH OUTLIERS AND MISSING DATA

Chia-Ming Cheng, Po-Hao Huang, Shang-Hong Lai

Department of Computer Science, National Tsing Hua University, Hsinchu 300, Taiwan, R.O.C.
Email: lai@cs.nthu.edu.tw

KEY WORDS: *structure from motion, robust estimation, projective reconstruction, metric upgrade, LMedS, M-estimation*

ABSTRACT:

In this paper, we propose a robust metric structure from motion (SfM) algorithm for an extended sequence with outliers and missing data. There are three main contributions in the proposed SfM algorithm. The first is a novel jury-based preemptive LMedS procedure to achieve efficient outlier detection. The second contribution is a new iterative two-step scheme that consists of robust estimation techniques for projective structure from motion. The third contribution is a novel algorithm for robust metric upgrade by applying the M-estimator to the traditional linear constraints for metric upgrade. In addition, comparisons of the proposed algorithm with some previous methods through experiments on simulated data are shown to demonstrate the efficiency and robustness of the proposed algorithm
.

1. INTRODUCTION

Structure from motion (SfM) has been one of the central problems in computer vision. Recent advances on multi-view geometry have been summarized in some representative books [1,2]. Since the outlier and missing data problems are inevitable during the process of automatic extraction and correspondence of feature points in practice, recent researches on SfM has focused on improving the robustness of SfM. In this paper, we proposed a novel algorithm to achieve the metric SfM for a long sequence with large missing data and outliers. We compare the proposed algorithm with previous methods through experiments on simulated data.

Some previous works on dealing with the missing data problem in SfM are briefly reviewed in the following. For the projective SfM, Fitzgibbon and Zisserman [4] proposed a solution based on trifocal tensor. Later, Martinec and Pajdla [3] proposed an algorithm that combines Sturm and Triggs' projective factorization method [5] and Jacob's fitting method [6] based on the subspace constraint. Note that this algorithm is used for comparison with the proposed method in the experimental results. On the other hand, several related works were developed under affine camera assumption, i.e. [6, 7], which simplifies the SfM to a linear system. This affine approximation of the SfM problem makes it equivalent to principle component analysis (PCA) with missing data [8], which is easier than the projective SfM in principle.

In addition, let us consider the other closely related issue - outlier problem. Up to now, there still exists no solution to handle outliers under projective SfM for a long sequence, though there were some previous methods developed based on pairwise or triplet views. For example, Torr [9] proposed the MAPSAC technique to estimate the fundamental matrix. Aanaes et al. [10] proposed to apply the robust M-estimators under the assumption of affine camera, thus leading to a linear system equivalent to the problem of robust PCA with outliers [11]. In this paper, we proposed a robust projective SfM algorithm to handle outliers in a long sequence.

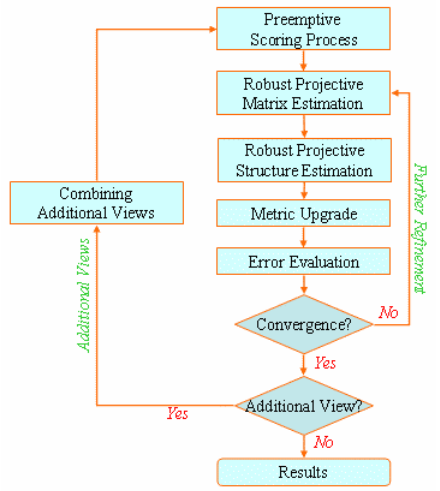

Preemptive Scoring Process
Robust Projective Matrix Estimation
Robust Projective Structure Estimation
Metric Upgrade
Error Evaluation
Convergence?
No
Further Refinement
Yes
Additional View?
Yes
No
Results
Combining Additional Views
Additional Views

Figure 1. Flow diagram of the proposed metric SfM algorithm.

The main challenges in SfM come from the input data contaminated by missing features, mismatches, and false positions. It is obvious that the subspace / rank constraint on SfM can alleviate the influence due to Gaussian image noises. However, the subspace constraint from the measurement matrix cannot effectively handle outliers. In addition, the high degree of freedom in the projective matrices as well as the unknown projective depth makes the detection of outliers difficult.

The main goal in this work is to develop a robust algorithm for metric SfM from contaminated data without pre-setting any case-by-case parameters. The flow diagram of the proposed algorithm is shown in Figure 1, and the details are given in the next section. There are three main innovative ideas in the proposed SfM algorithm. First of all, we propose a preemptive jury-based consensus process, which dramatically improves the computational efficiency of LMedS estimation for outlier elimination. Secondly, an iterative projective reconstruction algorithm is developed to achieve the desired robustness. In this algorithm, each iteration involves first using the preemptive LMedS procedure to determine the projective matrices and then applying the robust M-estimator to optimize the projective structure as well as the projective depth with the projection matrices fixed. Thirdly, a robust metric upgrade process by using the iterative reweighted least squared approach is proposed. For self-calibration, in order to reduce the sensitivity of decomposing the projection matrix into camera calibration matrix and metric camera motion, we further take advantage of hard constraints on the calibration matrix to achieve a more robust solution.

The rest of this paper is organized as follows. An overview of the proposed algorithm is given in the next section. In section 3, we describe the proposed preemptive jury-based LMedS technique. Then the proposed iterative two-step projective SfM algorithm is described in section 4. Section 5 presents the robust metric upgrade process as well as a self-calibration process. Subsequently, we demonstrate the performance of the proposed algorithm on both simulated and real data. Finally, we conclude this paper in the last section.

2. SYSTEM OVERVIEW

The structure from motion problem is to recover camera motions as well as object structure from a given image sequence. To focus on the 3D reconstruction problem, we assume the feature point correspondences across different views in the video are given. Note that the given correspondences may include imperfect data, i.e. missing data and outliers. The camera information includes intrinsic and extrinsic parameters: the intrinsic parameters are represented by the camera calibration matrix, K; the extrinsic parameters determine the 3×3 rotation matrix, R, and a camera translation vector $\mathbf{t}$.

The flow diagram of the proposed algorithm is shown in figure 1. Started from the preemptive scoring process, we score each observation from the two-view geometry, i.e. fundamental matrix. The second stage is the robust projective factorization via an iterative two-step reconstruction algorithm. Then a robust estimation approach is applied to the upgrade the projective reconstruction into a metric one. The error evaluation, obtained from the residues between the data matrix and the reconstructed projection and structure matrices, provides information for further refinement. Followed by combining additional views, we return to the first stage until all views are integrated.

3. JURY-BASED PREEMPTIVE LMEDS

RANSAC [12] and LMedS [2] are two traditional robust techniques to eliminate outliers. However, these techniques are computationally expensive. Therefore, we proposed a more efficient procedure to speed up the computational process. Motivated by Nister's preemptive RANSAC [14], we developed the so-called preemptive jury-based LMedS.

Referred to Nister's literature [14], the preemptive scheme can be categorized into the depth first and breath first manners. The depth first manner, noted as an order rule in the preemption scheme, dominates the hypothesis generation. This rule selects the inliers with higher likelihood for hypothesis generation according to previous experiences. On the other hand, the breath first fashion, noted as the preference rule, efficiently evaluates the hypotheses on equal footing. Not all observations are used to score all the selected hypotheses, but this rule eliminates bad hypotheses in the scoring procedure.

In principal, Nister's breath first preemptive scheme has a potential problem that the final result strongly depends on the scoring series. In his algorithm, the measurement is not on equal footing because earlier selected observations possess greater power in the hypotheses elimination than the later selected observations. We can declare that the breath first preemptive scheme works well only when the outlier rate is relatively small. Some experimental results will be shown later to support this argument.

To overcome the above problem with the breath first scheme and to further improve the efficiency in the LMedS technique, we develop a jury-based preemptive scheme in conjunction with the LMedS process. Instead of a single observation as used in the breath first scheme of the preemptive RANSAC method, we select a set of observations into a jury. Under the assumption of random sampling, the outlier rate in jury is approximately the same as that in whole. Thus, we can approximate the median value efficiently. The proposed jury-based preemptive LMedS process is given as follows,

```
1. Generate the hypotheses indexed with h = 1,…, $f(1)$.
2. Randomly permute the observations and classify them into $m$ juries.
3. Compute the scores $L_1(h)$ = median{ $\rho(j, h)$ | $j$ belongs to jury $1$} for $h = 1, . . . , f(1)$. Set $i = 2$.
4. Reorder the hypotheses so that the range $h = 1, . . . , f(i)$ contains the best $f(i)$ remaining hypotheses according to $L_{i-1}(h)$
5. If $i > m$ or $f(i) = 1$, quit with the best remaining hypothesis as the preferred one. Otherwise,
   compute the scores $L_i(h)$ = median{ $\rho(j, h)$ | $j$ belongs to jury $1 .. i$,} for $h = 1, . . . , f(i)$, set $i = i + 1$ and go to Step 4.
```

Algorithm 1. Jury-based preemptive LMedS algorithm

Note that, in Algorithm 1, $f(i) = \left\lfloor M\, 2^{-\left\lfloor \frac{i}{B} \right\rfloor} \right\rfloor$, where M is the total number of hypotheses and B is the block size, denotes a decreasing preemption function that indicates how many hypotheses are to be kept at each stage. The scoring function, $\rho(j,h)$, gives a scalar value representing the log likelihood of the observation, j, given that the hypothesis, h, is the correct motion model. Note that observations are random selection of the input matches which are not used for building hypothesis. For more details of the theoretical derivation of the preemptive scheme, the readers can refer to Nister's original paper [14]. The notations in this section follow those used in [14]. We modify the scoring process in the proposed preemptive LMedS scheme and improve the computational efficiency. Some experiments on simulated data are demonstrated to show its performance in section 6.

We applied this procedure to the two stages of our algorithm. One is the computation of fundamental matrix in the preemptive scoring process; the other is the projective factorization in the stage of robust projection matrix estimation.

4. ROBUST PROJECTIVE STRUCTURE FROM MOTION

For projective SfM, the factorization approach can be formulated as follows,

$$\mathbf{W} \equiv \mathbf{D} \otimes \mathbf{U} = \mathbf{P}\mathbf{X} = \left(\mathbf{P}\mathbf{H}\right)\left(\mathbf{H}^{-1}\mathbf{X}\right) = \tilde{\mathbf{P}}\tilde{\mathbf{X}} \qquad (1)$$

where $\mathbf{W}$ is the measurement matrix formed by input data matrix, $\mathbf{U}$, and their corresponding projective depths, $\mathbf{D}$. The operator $\otimes$ denotes the scale (projective depth) multiplying its corresponding vectors (homogeneous image coordinates). The matrix form is as follows,

$$\begin{bmatrix} \lambda_1^1\mathbf{u}_1^1 & \lambda_2^1\mathbf{u}_2^1 & \cdots & \lambda_n^1\mathbf{u}_n^1 \\ \lambda_1^2\mathbf{u}_1^2 & \lambda_2^2\mathbf{u}_2^2 & \cdots & \lambda_n^2\mathbf{u}_n^2 \\ \vdots & \vdots & \ddots & \vdots \\ \lambda_1^m\mathbf{u}_1^m & \lambda_2^m\mathbf{u}_2^m & \cdots & \lambda_n^m\mathbf{u}_n^m \end{bmatrix} = \begin{bmatrix} \mathbf{P}^1 \\ \mathbf{P}^2 \\ \vdots \\ \mathbf{P}^m \end{bmatrix} \begin{bmatrix} \mathbf{X}_1 & \mathbf{X}_2 & \cdots & \mathbf{X}_n \end{bmatrix} \qquad (2)$$

where $\mathbf{u}_j^i$ denotes the image position of the *j*-th point at the *i*-th view represented in a homogeneous 3-vector $\left(u_j^i, v_j^i, 1\right)^T$, X_j is the corresponding three-dimensional position of the *j*-th point represented in a homogeneous coordinate, $\mathbf{P}^i$ is a 3×4 camera projection matrix of the *i*-th view, and λ_j^i is the corresponding projective depth in the projective geometry. The projective 3D-to-2D transformation is written as $\lambda_j^i\mathbf{u}_j^i = \mathbf{P}^i\mathbf{X}_j$. Applying the singular value decomposition (SVD) to the measurement matrix enforces the subspace constraint, i.e. rank-4 constraint. Referred to [1], the algorithm iteratively tunes the projective depths with the subspace constraint to achieve the projective reconstruction. The convergence property is further discussed in [1].

4.1 Robust Determination of the Projection Matrices

Projective factorization is very sensitive to outliers. To overcome such challenges, robust techniques, such as RANSAC, can strategically be applied to the original algorithm to improve the results under outlier disturbance. The basic requirement for such robust techniques to eliminate outliers is that there exist more than necessary constraints, so that the reliability of each constraint can be consensually evaluated. For the projective factorization, the minimal reconstruction set requires 3 views with 6 corresponding points. The robust version for projective factorization is to apply the preemptive jury-based LMedS to the projective factorization [1]. Note that the feature selection is based on the Monte-Carlo process according to the preemptive scores.

4.2 M-estimator to Compute the Projective Structure

Given a set of 2D image points with the associated projection matrices, the corresponding three-dimensional feature points can be computed with a closed-from solution once the projective depths are known. However, the projective depths are generally unknown. Therefore, we carry out an iteratively approach to estimate the optimal three-dimensional structure by adjusting the projective depths appropriately. For the *k*-th iteration, we denote the current depth matrix as $\mathbf{D}^{(k)}$, and the closed-from solution can be formulated as follows,

$$\mathbf{X}^{(k)} = \mathbf{P}^{+}\left(\mathbf{D}^{(k)} \otimes \mathbf{U}\right) \qquad (3)$$

where $\mathbf{P}^{\dagger}$ denotes the pseudo-inverse of the matrix $\mathbf{P}$. Each projective depth entry in $\mathbf{D}^{(k+1)}$ at the next iteration is then updated as $\lambda_i^{j(k+1)} = \left\|\mathbf{P}^i\mathbf{X}_j^{(k)}\right\| / \left\|\mathbf{u}_j^i\right\|$. However, to further improve the robustness, we apply a robust measure at the outer loop. Thus, the RLS (re-weighted least square) solution replaces the LS (least square) solution in equation (3). We make use of the robust ρ function, such as the Lorentzion (or Cauchy) function [14] commonly used in robust statistics, to develop the M-estimation for the projective factorization. The robust ρ function is defined [14] as follows

$$\rho(r) = \log(1 + \frac{r^2}{2\hat{\sigma}^2}) \qquad (4)$$

The minimization of the robust energy function can be achieved by the iteratively RLS minimization. In this case, the weights are associated with the given projective matrices, and the residue is the norm of the error between the 2D image points and the obtained projective reconstruction which is determined at the inner-loop by adjusting the depths as described above. Thus the energy function to be minimized can be written as the following dynamic energy function, which is changing from iteration to iteration.

$$\mathbf{W}^{(k)}\mathbf{P}\mathbf{X} = \mathbf{W}^{(k)}\left(\mathbf{D} \otimes \mathbf{U}\right) \qquad (5)$$

where the weights associated with the residue is given as follows

$$w_i = \frac{2\hat{\sigma}^2}{2\hat{\sigma}^2 + r^2} \qquad (6)$$

Note that $\hat{\sigma} = 1.4826\left(1 + 5/(n-p)\right)\sqrt{E_{med}}$ is the median absolute deviation (MAD) estimation [14].

```
1. Initialize all the weights to 1, i.e. W = I.
    2a. Initialize all $\lambda_i^{j(0)} = 1$ for $\mathbf{D}^{(0)}$ and set $k = 0$.
    2b. Normalize the depths by multiplying each column
        of D with a constant factor.
    2c. Solve $\mathbf{X}^{(k)} = \mathbf{P}_{\mathbf{W}}^{+}\left(\mathbf{D}^{(k)} \otimes \mathbf{M}\right)$
    2d. Update $\lambda_i^{j(k+1)}$. Set $k = k + 1$
    2e. Exit the inner-loop if converged, else go to step
        2b.
3. Update the weights by the M-estimator from eq(5)
4. Exit if converged, else go to the inner loop in step 2.
```

Algorithm 2. Robust M-estimation of the projective structure with projective matrices given.

With this modification, the pseudo-inverse of $\mathbf{P}$, turns from the

LS solution, $\mathbf{P}^{\dagger}=\left(\mathbf{P}^{T}\mathbf{P}\right)^{-1}\mathbf{P}^{T}$, to the RLS solution, $\mathbf{P}_{\mathbf{W}}^{\dagger}=\left(\mathbf{P}^{T}\mathbf{W}^{2}\mathbf{P}\right)^{-1}\mathbf{P}^{T}\mathbf{W}^{2}$. Given the projective matrix, the algorithm2 shows the robust estimation of the homogeneous three-dimensional structure. Note that step 2 in Algorithm 2 is the inner-loop in order to iteratively determine the projective depths; step 1, 3, and 4 is for evaluating the reliability of the input projective matrices for robust estimation of the projective structure.

5. ROBUST METRIC UPGRADE

To upgrade the projective reconstruction to a metric one, we have to determine the ambiguity matrix, $\mathbf{H}$ in equation (1). This has to employ additional constraints, which may come from the prior knowledge of the camera calibration matrix. According to the absolute quadric constraint [17], the projection of the absolute quadric in the image yields the dual image absolute conic. This formulation of the absolute quadric constraint is shown as follows,

$$\omega_i^{*} = K_i K_i^{T} \;\infty\; P_i \Omega^{*} P_i^{T} \qquad (7)$$

The following assumptions provide linear constraints for the entries in a symmetric 4×4 rank-3 matrix Ω^{*}, i.e

$$\begin{array}{ll}
\hline
f_x = f_y & \mathbf{P}_i^{(1)}\Omega^{*}\mathbf{P}_i^{(1)T} = \mathbf{P}_i^{(2)}\Omega^{*}\mathbf{P}_i^{(2)T} \\
\hline
s = 0 & \mathbf{P}_i^{(1)}\Omega^{*}\mathbf{P}_i^{(2)T} = 0 \\
\hline
\left(u_o, v_o\right) & \mathbf{P}_i^{(1)}\Omega^{*}\mathbf{P}_i^{(3)T} = 0 \\
 & \mathbf{P}_i^{(2)}\Omega^{*}\mathbf{P}_i^{(3)T} = 0 \\
\hline
\end{array} \qquad (8)$$

Note that $\left(f_x, f_y\right)$ are the focal lengths along x- and y-axis, respectively, s denotes the skew factor, $\left(u_o, v_o\right)$ is the principle point or image center, and $\mathbf{P}_i^{(j)}$ denotes the *j*-th row of $\mathbf{P}_i$. The linear (closed-form) solution is referred to [15].

In order to obtain a more robust solution, we weight each constraint with the robust M-estimator, which is similar to the computation of the robust projective structure introduced in section 4.2. For each view, we have the following linear equations,

$$\begin{cases}
w_i\left(\mathbf{P}_i^{(1)}\tilde{\Omega}\mathbf{P}_i^{(1)T} - \mathbf{P}_i^{(2)}\tilde{\Omega}^{*}\mathbf{P}_i^{(2)T}\right) = 0 \\
w_i\left(\mathbf{P}_i^{(1)}\tilde{\Omega}^{*}\mathbf{P}_i^{(2)T}\right) = 0 \\
w_i\left(\mathbf{P}_i^{(1)}\tilde{\Omega}^{*}\mathbf{P}_i^{(3)T}\right) = 0 \\
w_i\left(\mathbf{P}_i^{(2)}\tilde{\Omega}^{*}\mathbf{P}_i^{(3)T}\right) = 0
\end{cases} \qquad (9)$$

In order to clarify the notation, we denote $\tilde{\mathbf{\Omega}}^{*}$ as the initial absolute quadric computed from the above equations, which may not be exactly rank-3. At the beginning, we set equal weights for each view, i.e. $w_i = 1$, to determine $\tilde{\mathbf{\Omega}}^{*}$. By enforcing the rank-3 constraint on $\tilde{\mathbf{\Omega}}^{*}$, we determine the absolute quadric $\mathbf{\Omega}^{*}$ via SVD. However, this step leads to additional errors in the linear system, thus we define the following residue for each projection matrix,

$$\begin{aligned}
r_i^{2} &= \left(\mathbf{P}_i^{(1)}\mathbf{\Omega}^{*}\mathbf{P}_i^{(1)T} - \mathbf{P}_i^{(2)}\mathbf{\Omega}^{*}\mathbf{P}_i^{(2)T}\right)^{2} \\
&\quad + \left(\mathbf{P}_i^{(1)}\mathbf{\Omega}^{*}\mathbf{P}_i^{(2)T}\right)^{2} + \left(\mathbf{P}_i^{(1)}\mathbf{\Omega}^{*}\mathbf{P}_i^{(3)T}\right)^{2} + \left(\mathbf{P}_i^{(2)}\mathbf{\Omega}^{*}\mathbf{P}_i^{(3)T}\right)
\end{aligned} \qquad (10)$$

According to the residues, we re-adjust the weights as equation (6), i.e. Lorentzion (or Cauchy) function [14] mentioned in 4.2. The RWLS process, iteratively reducing the residues under the rank-3 constraint on $\mathbf{\Omega}^{*}$ through tuning the weights, turns out to be an M-estimator for robust metric upgrade.

6. EXPERIMENTAL RESULTS

In this section, we show some experimental results of the proposed algorithm in comparison with some previous methods on simulated data. We first show the experimental comparison of the proposed jury-based preemptive LMedS algorithm, followed by the experimental comparison for the proposed SfM algorithm.

We used 200 point correspondences with additive Gaussian noises (standard deviation = 1.5) in image as well as 5~40% gross outliers. We compared the proposed preemptive LMedS, which uses the block size $B=1$ and 10 observations in a jury, with Nister's preemptive RANSAC, with the block size $B=10$, LMedS, and MAPSAC algorithms with this experiment on fundamental matrix estimation with contaminated data._For a fair comparison, we used the same Torr's seven-point method for computing the fundamental matrix for all the above four algorithms. Furthermore, the four algorithms share the same set of hypotheses which were randomly generated from 1000 samples, so that we examined which of these four methods makes the best use of the hypothesis. The experimental results shown in Figure 2 indicate that the proposed scheme approximates the performance of the full-scoring procedures, i.e. LMedS and MAPSAC, and it reduces 90% of the full-scoring burdens. Thus, it shows the advantage in the computational efficiency of the proposed algorithm.

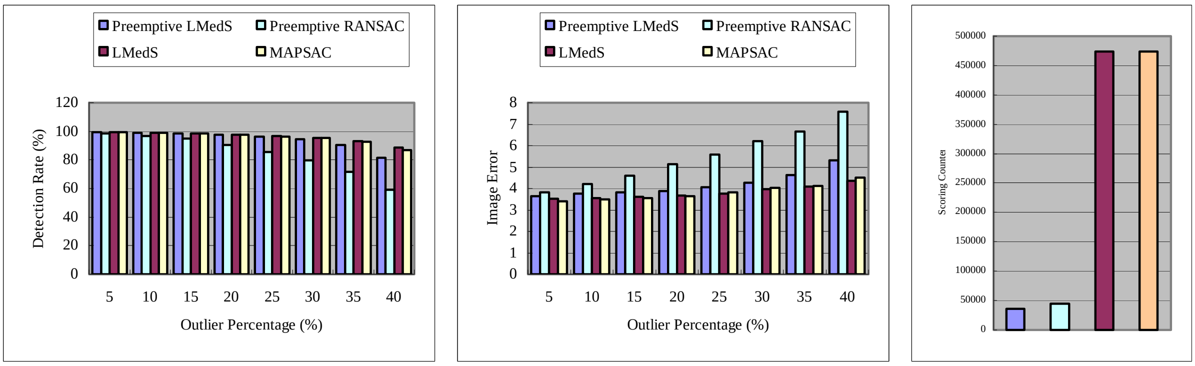


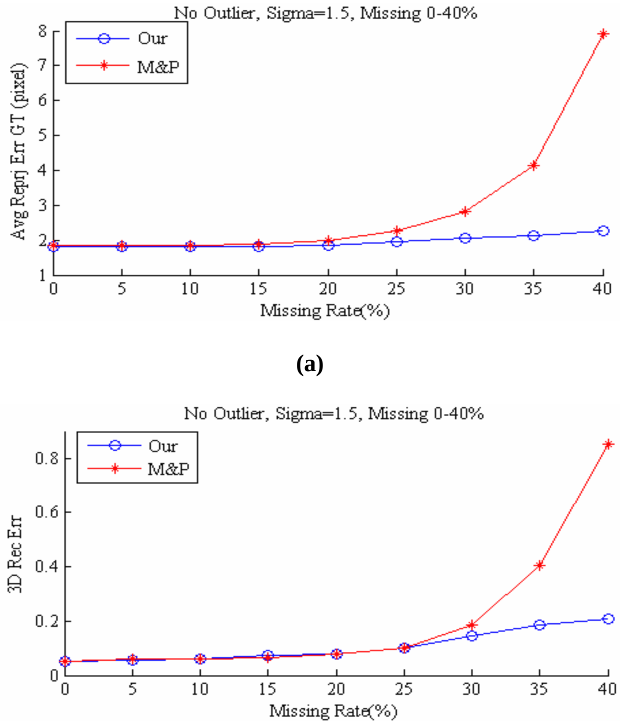
Figure 2. Robust computation for increasing outlier rate with the same set of hypotheses. Left: detection rate of the outliers; Middle: Image error that represents the correctness of fundamental matrix; Right: scoring counter.

In the following, we show the experimental comparison of the SfM methods on simulated data. The simulations are designed to examine the performance under different data missing and outlier rates. We compare the proposed SfM algorithm with Martinec and Pajdla's algorithm [1]. Their code is available from the public domain. Note that their metric upgrade process is removed since it crashed for some simulated cases. So we upgraded their projective reconstruction according to the ground truth.

The simulation environment is as follows. First of all, we randomly generated 300 points within a 20 unit length squared box in 3D space, and its center is randomly located around the world center in the radius of 10 unit length. Thirty cameras are located in a circle of radius 100 unit length, and their viewing directions are the world center plus additional random rotation within 5 degrees in Euler angle. Calibration matrices are constant with focal lengths within 1500, and skew parameters 1.5, and image center (1000, 650). Each observation point is perturbed with Gaussian noises $\sigma = 1.5$, followed by the rounding operation. The outliers are randomly selected according to the simulated outlier rate. The farthest points to the current camera are selected as the missing points at that view according to a given missing rate. One hundred trials are made to obtain the final results.

We examine the reprojection error, 3D reconstruction error, and reconstruction rate at different data missing rates and outlier rates to compare the performance. The reprojection error evaluates the error between reconstructed and measured points in image space, and the 3D error is measured in RMS of the Euclidean distance of the simulated unit length. The reconstruction rate is the ratio of the reconstructed points to the total number of points.

In the first simulation, we examine the algorithms with different missing rates as shown in Figure 3. In the second simulation, we examine the robustness under different outlier rates with a constant missing rate as shown in Figure 4. Then, we test the performance with more views integrated as shown in Figure 5.



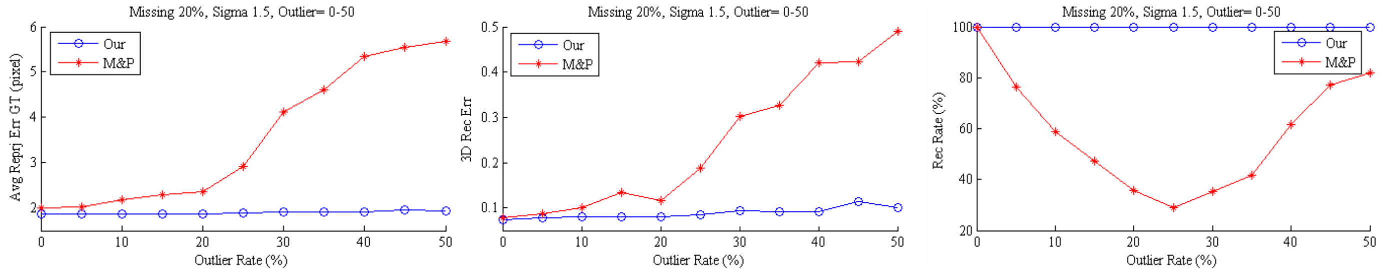
(a)

(b)

Figure 3. (a) The reprojection errors and (b) the 3D errors at different missing rates for the proposed algorithm and Martinec and Pajdla's algorithm.

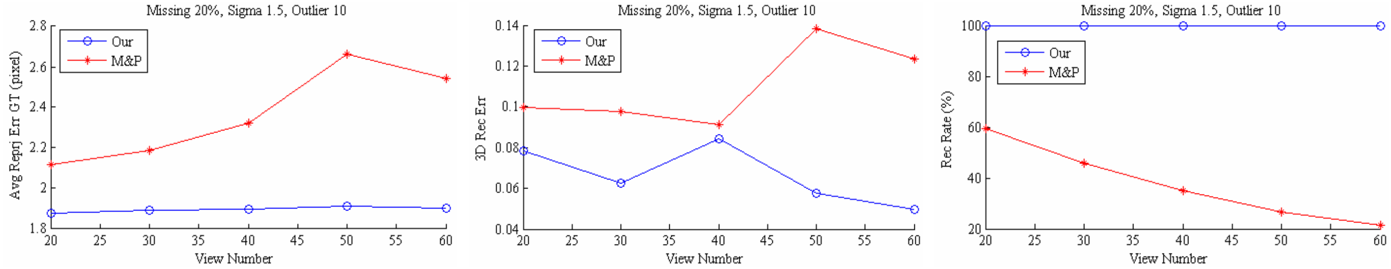
7. CONCLUSION

In this paper, we proposed a novel robust metric structure from motion algorithm for a long sequence with outliers and large missing data. The jury-based preemptive LMedS procedure was developed to achieve efficient outlier detection in the robust projective SfM. In addition, we also applied robust estimation techniques in the projective SfM as well as the metric upgrade processes. We demonstrate the robustness, accuracy and efficiency of the proposed SfM algorithm through experimental comparisons with previous methods.



(a) (b) (c)

Figure 4. (a) The reprojection errors; (b) the 3D errors; (c) the reconstruction rates at different outlier rates for the proposed algorithm and Martinec and Pajdla's algorithm.



(a) (b) (c)

Figure 5. (a) The re-projection errors; (b) the 3D errors; (c) the reconstruction rates at different views for the proposed algorithm and Martinec and Pajdla's algorithm.

ACKNOWLEDGEMENTS

This work was supported by National Science Council (project code: 93-2213-E-007-003).

REFERENCES

[1] Hartley, R. and Zisserman A., 2000. *Multiple View Geometry in Computer Vision*. Cambridge University Press, Cambridge, UK.

[2] Faugeras, O., Luong, Q.-T., and Papadopoulo, T., 2001. *The Geometry of Multiple Images*, MIT Press.

[3] Martinec, D. and Pajdla, T., 2002. Structure from many perspective images with occlusion. *Proc. European Conf. Computer Vision*, pp. 355-369.

[4] Fitzgibbon, A. W. and Zisserman, A., 1998. Automatic camera recovery for closed or open image sequences. *Proc. European Conference on Computer Vision*, Springer-Verlag, pp. 311–326.

[5] Sturm, P. and Triggs, B., 1996. A factorization based algorithm for multi-image projective structure and motion. *Proc. European Conference on Computer Vision* (II), pp. 709–720.

[6] Jacobs, D., 1997. Linear fitting with missing data: Applications to structure from motion and to characterizing intensity images. Proc. *IEEE Conf. Computer Vision and Pattern Recognition*, pp.206 - 212.

[7] Chen, P. and Suter, D., 2004. Recovering the missing components in a large noisy low-rank matrix: application to SFM. *IEEE Transactions on Pattern Analysis Machine Intelligence*, Vol. 26, No. 8, pp. 1051-1063.

[8] Wiberg, T., 1976. Computation of principal components when data is missing. *Proc. Second Symp. Computational Statistics*, pages 229-236.

[9] Torr, P. H. S., 2002. Bayesian model estimation and selection for Epipolar geometry and generic manifold fitting. *International Journal of Computer Vision*, 50(1), pp. 35-61.

[10] Aanaes, H., Fisker, R., Astrom, K., and Carstensen, J.M., 2002. Robust factorization. *IEEE Transactions Pattern Analysis Machine Intelligence*, 24(9) , pp. 1215-1225.

[11] Torre F. D. and Black, M. J., 2003. A framework for robust subspace learning. *International Journal of Computer Vision*, 54 (1/2/3), pp. 117-142.

[12] Fischler, M. A. and Bolles, R. C., 1981. Random sample consensus: A paradigm for model fitting with application to image analysis and automated cartography. *Communications of the ACM*. (24), pp.381-395.

[13] Rousseeuw, P. and Leroy, A. 1987. *Robust Regression and Outlier Detection*. John Wiley & Sons: New York.

[14] Nister, D. 2003. Preemptive RANSAC for live structure and motion estimation. *International Conference Computer Vision*, pp.199 – 206.

[15] Li, G.., 1985. Robust regression. in *Exploring Data Tables, Trends, and Shapes* (D. C. Hoaglin, F. Mosteller and J. W. Tukey Ed.), Wiley, New York, pp. 281-343.

[16] Pollefeys, M., Koch, R., Van Gool, L., 1998. Self-calibration and metric reconstruction in spite of varying and unknown intrinsic camera parameters. *Proc. International Conference Computer Vision*, pp.90-95.

[17] Triggs, B., 1997. The absolute quadric. *Proc. IEEE Conference Computer Vision Pattern Recognition*, pp. 609-614

SESSION 3

SURFACE RECONSTRUCTION

DIRECT SHAPE FROM ISOPHOTES

V.Dragnea, E.Angelopoulou

Computer Science Department, Stevens Institute of Technology, Hoboken, NJ 07030, USA -
(vdragnea, elli)@ cs.stevens.edu

KEY WORDS: Isophotes, Reflectance Map

ABSTRACT:

This article describes a new method of Shape from Shading, Shape from Isophotes. Shape from Isophotes uses the image isophotes for recovering the object surface normals. It is a propagation method. It initially directly recovers a small number of surface normals and then uses them to estimate normals at neighboring points in either adjacent isophotes or within the same isophote. The propagation can start either from occluding contours or singular points. Shape from Isophotes explicitly addresses brightness quantization errors, which can affect the performance of traditional Shape from Shading techniques. Since our method is based on the relationship between isophote curves and changes in surface normals [9,10], it is mainly applicable to smooth diffuse surfaces. The accuracy of the proposed technique was measured using synthetic images of simple objects with Lambertian reflectance, as well as real objects of known geometry. The normal map was recovered with accuracy of well bellow 7º average error. The method requires some interpolation, as it is possible that we may not be able to recover the surface normals at each pixel

1. INTRODUCTION

There is a big body of work done in the field of Shape from Shading (SFS), [1,2,3,4,5,7,8,11,12,14,15] just to mention a few. Despite their inherent limitations (they often need input images of scenes with strong parallel illumination rays and/or impose object surface restrictions), they are still often used especially for the shape recovery of smooth, featureless surfaces.

Among the first SFS approaches was Horn’s use of a set of five differential equations whose solution produces a curve [5], a characteristic strip. The direction of characteristic strips is the direction of the intensity gradients, and in the case of a rotationally symmetric reflectance map they are the curves of steepest descent. Though Horn’s characteristic strip technique demonstrated that the recovery of a normal map is feasible from a single image, it may, like most gradient descent methods, produce erroneous results (for more details see section 2.1). Newer algorithms that are still based on gradient descent like Dupuis and Oliensis [3], and Bichsel and Pentland [1] are still suffering from the same limitation.

Other researchers used different techniques to recover the surface normals from intensity images. For example, Kimmel and Bruckstein [8] use level sets for recovering shape from shading. Another class of SFS algorithms treats shape recovery from irradiance as a minimization problem. For example, Horn’s [7] minimization approach replaces the smoothness constraint with an integrability constraint. Frankot and Chellappa’s [4] minimization approach places emphasis in enforcing integrability. Zheng and Chellappa [15] replace the smoothness constraint with an intensity gradient constraint. Pentland uses a local approach in [11] based on the local sphericity assumption. However, they too have their limitations. Most minimization approaches have a tendency to be slow. Furthermore, standard variational algorithms may not reconstruct a surface from noisy images even after thousands of iterations [2].

Other shape recovery methods obtained very good results: photometric stereo [13], stereopsis, moving light source, structured light. These methods require multiple images often obtained under controlled conditions. Photometric stereo uses multiple images of the same object taken under different lighting conditions. The multiple illumination setup creates a system of irradiance equations which are used to recover the normal map (and albedo). Binocular (polyocular) stereo does not typically impose any restrictions on illumination, but requires capturing a scene from multiple viewpoints using at least two cameras. It tries to identify features in two or more images that are projections of the same entity in the 3D world. Structured light uses pictures of objects illuminated by a pattern of light. The camera senses the result as a grid distorted by the surface structure and its pose. Thus, although these shape recovery methods do not have the same constraints as SFS methods, they too have their inherent limitations.

Our methodology, Shape from Isophotes, is focusing on overcoming some of the shortcomings of SFS methods. More specifically, Shape from Isophotes avoids differentiation which results in improved (pixel level) accuracy. Unlike most of the previous methods, it is not gradient descent based and can be applied on piecewise smooth surfaces. It also explicitly addresses complications that may arise from brightness quantization errors. Our technique is not without limitations. Like other SFS methods, we too assume single distant point light source and orthographic projection.

The Shape from Isophotes (SFI) method is based on the close relationship between isophotes (regions of constant brightness) and surface normals [9, 10]. It initially directly recovers a small number of surface normals either on occluding contours or on singular points, in general on any pixel where a surface normal estimate can be directly obtained from the image. We then use the structure of isophote regions in an image to recover the remaining surface normals. Specifically, our method is composed of 2 parts: a) the method that deals with surface normals at the border of isophote regions, which we will refer from now on as *the border method* and b) the method that propagates the surface normal information within an isophote region which we call *the interior method*. The border method calculates new values for the surface normal when there is a change in the image brightness between adjacent pixels, the interior method chooses the propagation direction for a known surface normal as long as there is no change in the image brightness.

2. OVERVIEW OF SHAPE FROM ISOPHOTES METHOD

2.1 Intensity Gradients and Curves of Steepest Descent

Many SFS algorithms [1, 3, 5, 8] use the direction of steepest descent, in recovering the surface normals. This principle states that if a step is taken in the image plane in a direction parallel to the gradient of the reflectance map, the corresponding step in gradient space is parallel to the gradient in the image. For rotationally symmetric reflectance maps, the direction of intensity gradient is also the direction of steepest descent [6]. However, these algorithms may fail to orient the surface normal correctly, particularly in regions where different surfaces may result in similar intensity gradients (for a graphic representation of this problem see figure 2). The effect is more pronounced when surface orientation is recovered based on local information.

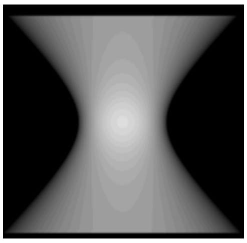


Figure 1. Synthetic image of a hyperboloid, Lambertian surface, illuminated by a distant point light source aligned with the optic axis.

Consider, for example a hyperboloid Lambertian surface, illuminated by a single distant light source aligned with the optic axis (see fig. 1). We use the notation (p,q) for the gradient of the surface. p is the slope of the surface with respect to the X-axis, i.e. $p = \partial f(x,y)/\partial x$ while q is the slope of the surface with respect to the Y-axis, i.e. $q = \partial f(x,y)/\partial y$. Assume that the reflectance map R(p,q) has a unique isolated maximum at (p_0,q_0), which means $R(p,q)<R(p_0,q_0)$ for all $(p,q) \neq (p_0,q_0)$ where R(p,q) is the reflectance map of the surface. Assume that at some point (x_0,y_0) in the image, the image irradiance $E(x_0,y_0)=R(p_0,q_0)$. This point is called a singular point and the gradient (p,q) at this point is uniquely determined to be (p_0,q_0) [6]. In our example, the singular point is at the centre of the image. The isophotes close to the singular point are almost circular. Still the surface is far from being rotationally symmetric. This suggests that there are cases when the characteristic curves methods may produce erroneous results For instance, as shown in fig. 3, a characteristic strips method can lead to incorrect normals during propagation, even though it starts with accurate normal recovery at the singular point.

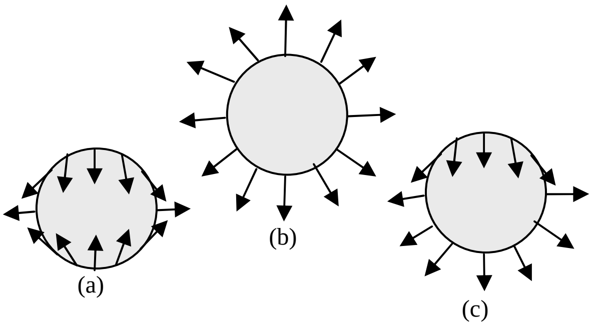


Figure 2. Three possible surface normal orientations on a singular region border

In fact, the singular point is often a singular region in images, due to brightness quantization, and most of its normals are neither identical, nor parallel to the incident light direction. Figure 2 graphically demonstrates a few possible normal orientations at the border of the singular region. The normals' directions inside the singular region are close enough to the light direction so that the whole region has maximum brightness. The propagation methods often approximate the region around the singular point with a spherical cap.

Characteristic strips methods are particularly sensitive in surface recovery in hyperbolic points, partly because they propagate normal information along the direction of intensity gradients. In order to overcome this, we decoupled the surface normal recovery computation from the propagation process. Our method, Shape from Isophotes, does not use the direction of steepest descent. More specifically, the surface normal recovery computation is done during the border method phase (see section 2.2) and the propagation during the interior method phase (see section 2.3).

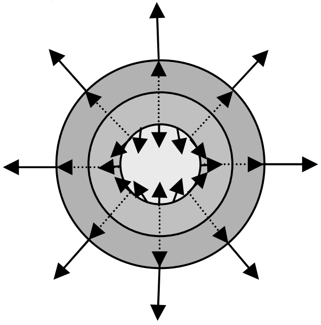


Figure 3. Characteristic strips (in dotted line).

Figure 4 shows the expected outcome of Shape from Isophotes for the same example of singular region. The solid arrows represent normals, the dotted lines the propagation path; the grey areas are isophote regions and the lines between them the borders between the isophote regions. The singular region has surface normals on the border similar to those in fig. 2(a).

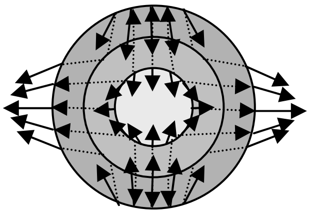


Figure 4. Shape from isophotes method.

2.2 Surface Normals at the Border of Isophote Regions

Consider two infinitesimally small planar surface patches p_1 and p_2 such that they correspond to image patches close to the isophote border which also lie on different sides of the border. Assume that we know the normal $\mathbf{n}_1$ to the surface at a point P_1 inside the surface patch p_1. P_1 is adjacent to the border with the other isophote region. Then there is enough information to find the normal $\mathbf{n}_2$ at the point P_2 inside the surface patch p_2.

Let **n** be the normal to the plane p which is perpendicular to the image plane and tangent to the isophote border (see figs. 5, 6, and 7). Then the normals $\mathbf{n}_1$, $\mathbf{n}_2$ and **n** are coplanar, being normals to three planes whose intersection is a line. $\mathbf{n}_1$ and **n** constraint $\mathbf{n}_2$. Let S_2 be the set of possible surface normal for patch p_2 (see fig. 7). Since $\mathbf{n}_2$ must be coplanar to $\mathbf{n}_1$ and **n**, there is a very small solution space for S_2. For example, for a Lambertian surface, the normals $\mathbf{n}_1$ and **n** must lie on a cone centred at the light source direction (p_S,q_S). Then we can find the possible $\mathbf{n}_2$ values by applying the co-planarity constraint to S_2.

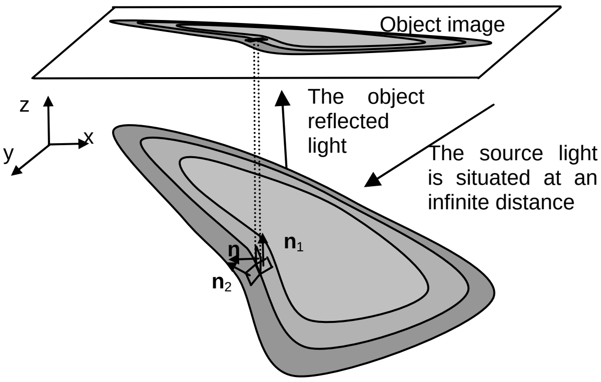

Object image
z
x
y
The object reflected light
The source light is situated at an infinite distance
n
n1
n2

Figure 5. The image formation and the patch-plane correspondence. An object, its image, the isophote regions, the p_1 and p_2 patches, the p plane and the $\mathbf{n}_1$, $\mathbf{n}_2$ and **n** normals.

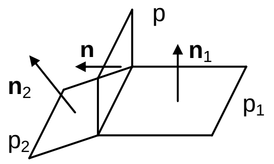


Figure 6. $\mathbf{n}_1$, $\mathbf{n}_2$ and **n** are coplanar if the planes intersect on a line. The p_1 and p_2 patches, the p plane and the $\mathbf{n}_1$, $\mathbf{n}_2$ and **n** normals isolated

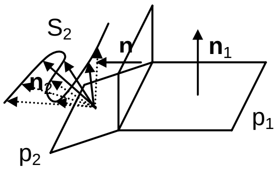


Figure. 7. The intersection between S_2 (the set of possible normals for the brightness of the point P_2 and light direction) and the plane $(\mathbf{n}_1,\mathbf{n})$ gives three solutions in this particular case.

However, the only feasible $\mathbf{n}_2$ values must lie on the intersection of the cone with the plane that contains $\mathbf{n}_1$ and **n**. Then 0, 1 or 2 solutions are obtained. Normally, 0 solutions should never be obtained. If 0 solutions are obtained, the starting normal had the wrong value and one should restart the computation with a different starting normal value. 1 solution is obtained inside a singular region. 2 solutions are obtained everywhere else. In practice, quantization errors make obtaining 0 solutions possible even if there should be 2 very close solutions. We chose to stop the propagation if the solutions were close to each other. In conclusion, by using our border method, we can estimate the surface normals at a pixel bordering an isophote region, once the normal at the other side of the border is known.

2.3 Surface Normals within an Isophote Region

Once a normal of an isophote region is known we can propagate that information inside an isophote region. We assume that locally the isophote is a developable surface with generator line L'_0. We propagate the surface normal information along the generator line or its approximation since that is the direction of minimal change. For approximately Lambertian surfaces it suffices to propagate along the plane of incidence. The plane of incidence at a point P is the plane defined by the incidence beam and the surface normal at point P (see fig. 8).

If the intersection between the incidence plane and the surface is a line, then it is the generator line L'_0. However, brightness quantization can create isophote regions resulting in the loss of finer shading information which cannot be recovered. Though we can not recover the original object shape, we can still extract the shape of a polygonal object which approximates the original object shape. For the approximate polygonal object, the image isophotes correspond to the planes of the object. We need to find out the generator lines L'_0 which generate these planes. The surface normal does not change along that line.

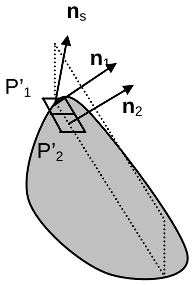
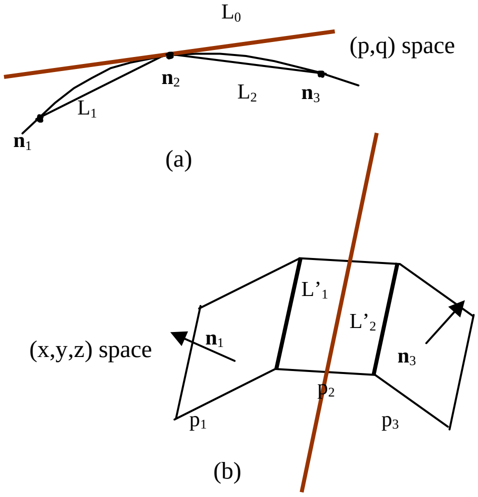


Figure 8. The construction for Surface Normals within an Isophote Region for a Lambertian surface

For diffuse surfaces in general, we can compute L'_0 as follows. Assume that we choose three points $\mathbf{n}_1$, $\mathbf{n}_2$, $\mathbf{n}_3$ in (p,q) space so that they lie on the locus of the same constant intensity (see fig. 9a). Each of these three points $\mathbf{n}_i$ is represented by a plane p_i in (x,y,z) space (see fig. 9b). The intersection between any pair of these planes in (x,y,z) space is a line L'_i which is represented in (p,q) space by the line L_i that connects the corresponding points of those planes. The intersection between the planes p_1 and p_2 is represented by the line L_1 which connects $\mathbf{n}_1$ and $\mathbf{n}_2$. Now if $\mathbf{n}_1$, $\mathbf{n}_2$, $\mathbf{n}_3$ are very close to each other, the lines that connect them are almost parallel to the tangent L_0 to the curve in the middle point $\mathbf{n}_2$. This means that the intersections of the planes p_1, p_2 and p_3 become almost parallel to the line L'_0 in (x,y,z) that corresponds to that tangent. One can think of the planes p_1, p_2 and p_3 as part of a developable surface whose generator is line L'_0.

Consider now all the points of the locus on which $\mathbf{n}_1$, $\mathbf{n}_2$ and $\mathbf{n}_3$ lie. Each of the points on that locus corresponds to a plane in (x,y,z) space. For each triplet of closely spaced consecutive points on the locus, one can apply the same logic. Thus, the normals on the entire locus corresponds to a developable surface whose generator is line L'_0. The entire developable surface is contained in the same isophote in our image.

The normals in other points of the same isophote region can be obtained either by interpolation, or by applying either the interior method or the border method to another known normal.



(a)

(b)

Figure 9. Three points on a curve both in a) (p,q) space and b) (x,y,z) space

2.4 Shape from Isophotes

In order to extract the surface normals at the whole surface, we combine both the border method and the interior method. On the isophote regions' borders the border method is applied, while in the interior of the isophotes regions' the interior method is applied. Both methods introduce some errors due to quantization, but the errors are within acceptable limits (see sections 4.1 and 4.2).

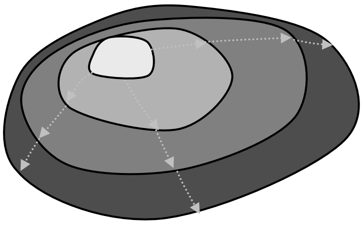


Figure 10. Shape from Isophotes can start at a singular point, as in this example, of from occluding contours.

Figure 10 shows the method applied to a simple convex object. Each different shade of grey represent a distinct isophote region. The propagation in this example started from the singular point. The arrows represent propagation directions.

Figure 11 shows the first 4 cycles of the method. The object was a sphere. The starting points are the border of the singular region. The upper row shows the propagation curves. The lower row shows the detected isophote region borders.

To summarize, the Shape from Shading algorithm is as follows:
1. Directly compute starting surface normals where available (e.g. at singular points or occluding contours)
2. Propagate away from the starting normals as follows:

If the current normal is adjacent to an isophote region apply the "border method, else apply the "interior method"

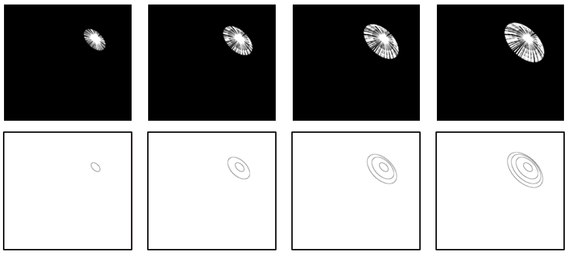


Figure 11. Hyperboloid, Lambertian surface, light perpendicular to the image.

3. EXPERIMENTS

In order to test the limitations of our method we first performed a series of test on synthetic images. That allows us to provide quantitative error measurements and examine the sensitivity of our method to noise. The objects' geometry and surface normals are known.

3.1 Synthetic Data

The sample images were created using a Lambertian surface, the error was calculated by averaging the angle between the calculated normal and the known normal at every point of the image that represents the object. The light source is a point light source at infinite distance. We tested the algorithm on two shapes: a hyperboloid and a cone. Two errors were calculated for each object: one before there was any interpolation done and one after the interpolation. Our first test (see figures 12 and 13) assumed ideal data, with no noise, so our only source of inaccuracies is quantization error. In order to test the sensitivity of our algorithm to noise, we added a random noise of +/-2 intensity values at each pixel (see figures 14 and 15).

Shape	**Average error before interpolation**	**Average error after interpolation**
Hyperboloid – noise free	1.502% (2.704º)	1.700% (3.060º)
Cone – noise free	1.665% (2.998º)	1.769% (3.184º)
Hyperboloid – noisy	2.719% (4.892º)	2.800% (5.039º)
Cone – noisy	2.134% (3.842º)	2.431% (4.376º)

Table 1. Average error for the Hyperboloid and Cone synthetic images

We started the propagation from the points on the occluding boundary. The albedo of the objects was known in advance. In each of the figures 12, 13, 14 and 15 the image that was analyzed is shown on the left, the recovered normal map is shown in the middle and an error image is shown on the right. The error image shows the error distribution on the surface. Darker areas have smaller errors, lighter ones bigger ones. The contrast was enhanced from the original image so that white corresponds to 100% error and black to 0% error for a better visibility. The biggest errors occur in the interpolated areas.

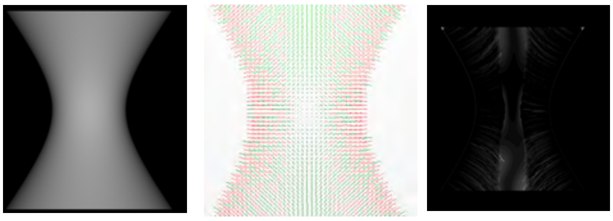


Figure 12. . Hyperboloid, Lambertian surface, light perpendicular to the image

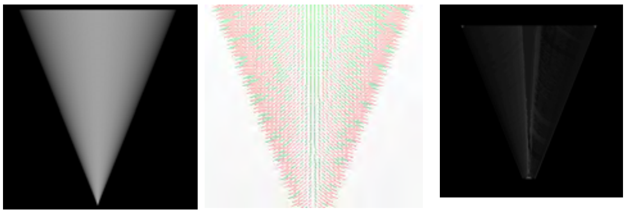


Figure 13. Cone, Lambertian surface, light perpendicular to the image

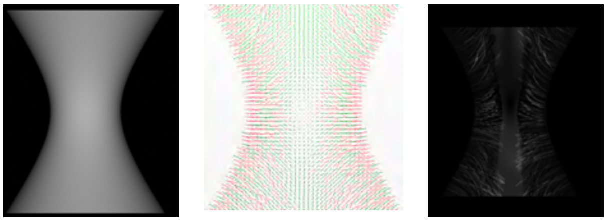


Figure 14. Hyperboloid, Lambertian surface, light perpendicular to the image

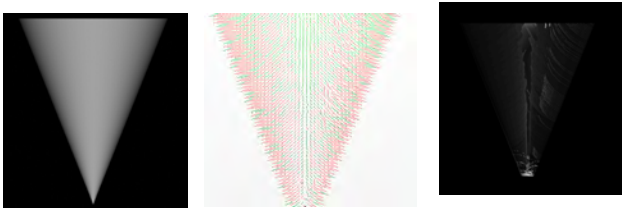


Figure 15. Cone, Lambertian surface, light perpendicular to the image.

3.2 Real Data

We also applied our algorithms on real data. The images were taken in a controlled environment. The objects were less than 10cm tall and the single light source was positioned about 50 cm away from the objects. The light source was positioned roughly above the XCDSX900 Sony camera. We used cross-polarization to eliminate specularities. We performed experiments on a white torus (fig. 16) and a billiard cue ball (fig. 17).

Shape	**Average error before interpolation**	**Average error after interpolation**
Billiard ball	3.813% (6.864º)	3.724% (6.704º)

Table 2. Average error for the Billiard ball image

Figures 16 and 17 show the images of the objects on the left and the recovered normals on the right. Since the billiard cue ball has known dimensions we performed error analysis on that object. As expected, the error is bigger than in the synthetic images, but an average error of less than 7º is a strong performance. The bigger error of the billiard ball sample is due to image noise, reflections, non-uniformity of the illuminant and inaccuracies in the position of the light source.

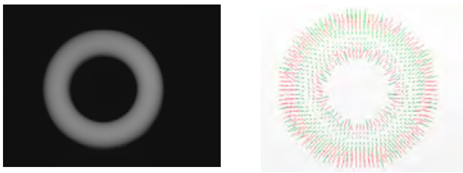


Figure 16. Torus

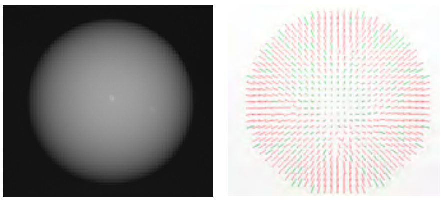
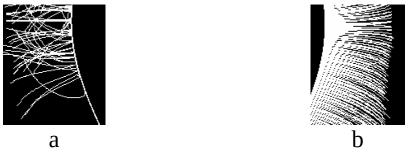


Figure 17. Billiard ball

4. LIMITATIONS

Our method has its limitations. Its accuracy depends on how accurately we can recover the surface normals at occluding contours. We tested the sensitivity of Shape From Isophotes to the accuracy of the estimates at the occluding contours for the hyperboloid synthetic image. Each normal at the occluding contour was perturbed by at most 1º, 2º, 3º, 4º and 5º accordingly. As expected the accuracy of the recovered normals was affected proportionally (see table 3), but the error in the recovered map was still below 18º. Furthermore, when the normals at occluding contours are error prone, the propagation paths are short and erratic (Figure 18a). Thus, one can detect that feature and associate a reliability measurement to each recovered normal.



a b

Figure 18. Propagation curves for a) erroneous starting surface normals at the occluding contours and b) accurate starting surface normals at the occluding contours

Initial normals errors	Average Recovered normals error
0	1.465% (2.638º)
Between -1º and 1º	1.911% (3.441º)
Between -2º and 2º	3.460% (6.228º)
Between -3º and 3º	5.382% (9.687º)
Between -4º and 4º	7.921% (14.259º)
Between -5º and 5º	9.664% (17.396º)

Table 3. Average recovered normals error for an hyperboloid image when there are starting normals errors

Lastly, like other SFS methods, our technique, as is, is currently applicable to diffuse surfaces only. The surface generated by propagating from the starting curve might not cover the whole visible surface, so additional curves or some interpolation might be needed.

5. UNKNOWN LIGHT SOURCES

The light source does not need to be parallel to the viewer direction. Figure 19 shows normals recovered when the light is not parallel to the viewer direction, but is known. More specifically, in each of the synthetic images shown in figure 19 (from left to right) we moved the light source along the X-axis so that it would form a 5.7º, 11.3º and 16.7º angle with the optic axis accordingly. The error in the recovered map is 1.616% (2.908º), 1.452% (2.613º) and 1.621% (2.918º) accordingly. After interpolation the error becomes 1.747% (3.144º), 2.172% (3.910º) and 2.447% (4.404º).

If our estimate of the light source position is erroneous, the normal map can still be recovered, but the larger our error in the light source position, the bigger our error in the recovered normal map. More specifically, for light source direction error of 5.7º the normal map error is 2.389% (4.301º), for 11.3 º it is 4.407% (7.933º) after the initial light source direction error was again subtracted from the obtained normals. In this latter case the coverage was reduced to one side of the object and the strips were looking chaotic (figure 18a).

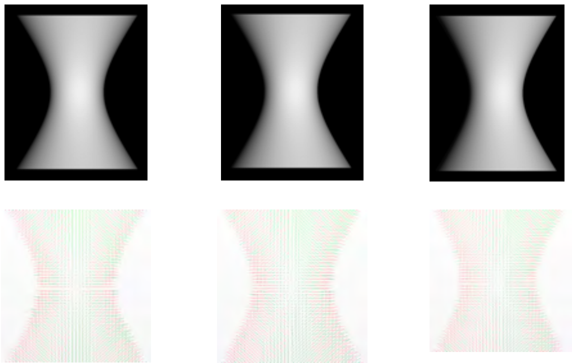


Figure 19. Hyperboloid, the source light comes from the right 5.7º, 11.3º and 16.7º

One can take advantage of the localized irregularities in the propagation direction and use them for iteratively improving the light source estimate. For example, the following algorithm could be used for detecting the light source direction, starting from an initial light source estimate:
1. Calculate starting normals using the light source direction
2. Calculate the normal map
3. Check strip coverage.
4. If the strip coverage contains erratic strips, move the light source toward the opposite direction of the object region which contains those strips and go to 1. Otherwise end the algorithm and keep the recovered light direction.

6. CONCLUSIONS

We developed a new Shape from Shading method that uses the image isophotes in recovering surface normals. Unlike minimization methods it does not suffer from numerical instabilities and because the propagation direction is decoupled from the gradient direction it is less error prone than characteristic strip methods in areas where the gradient is zero. Our quantitative error analysis showed an improved performance with average error of less than 7º. The errors are attributed to noise and to the fact that real images do not fully satisfy the simplifying assumptions of our theory (i.e. inter-reflections, not truly distant light sources, etc.). Future work includes experiments using complex surfaces and relaxing more initial conditions. We are currently investigating isophote based techniques for iteratively estimating the light source direction and the normal map. We also want to expand our method to work with more complex reflectance maps, i.e textured surfaces and surfaces with specularities and shadows.

7. REFERENCES

[1] M.Bichsel and A.P.Pentland, “A Simple Algorithm for Shape from Shading” *IEEE Proc. Computer Vision and Pattern Recognition*, pp. 459-465, 1992.

[2] Paul Dupuis, John Oliensis, “Direct Method For Reconstructing Shape From Shading”, *Proc. SPIE Conf. 1570 on Geometric Methods in Computer Vision*, pp. 116-128, July 1991.

[3] Paul Dupuis, John Oliensis, “Direct Method For Reconstructing Shape From Shading”, *IEEE Proc. Computer Vision and Pattern recognition*, pp. 720-721, June 1993.

[4] R.T.Frankot and R.Chellappa, “A Method for Enforcing Integrability in Shape from Shading Algorithms”, *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 10, pp 439-451, 1988.

[5] B.K.P. Horn, “Shape from Shading: A Method for Obtaining the Shape of a Smooth Opaque Object from One View”, *PhD thesis, Massachusetts Inst. Of Technology*, 1970.

[6] B.K.P. Horn, *Robot Vision* MIT Press, Cambridge, MA, 1987.

[7] B.K.P. Horn, “Height and Gradient from Shading”, *Int’l Journal of Computer Vision*, pp 37-75, 1989.

[8] R.Kimmel and A.M.Bruckstein, “Tracking Level Sets by Level Sets: A Method for Solving the Shape from Shading problem”, *Computer Vision and Image Understanding*, vol. 62, pp 47-58, July 1995.

[9] J.J.Koenderink and A.J. Van Doorn, “Photometric Invariants Related To Solid Shape”, *Journal of Modern Optics*, July 1980, vol. 27, no. 7, pp. 981-996

[10] Jan J.Koenderink, “What does the occluding contour tell us about solid shape?”, *Perception*, 1984, vol. 13, 321-330

[11] A.P.Pentland, “Local Shading Analysis”, *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 6, pp. 170-187, March 1984.

[12] Ariel Tankus, Nir Sochen, Yehezkel Yeshurun, “A New Perspective [on] Shape-from-Shading” *Proceedings of the Ninth IEEE International Conference on Computer Vision - Volume 2, 2003.*

[13] Robert J. Woodham, “Photometric method for determining surface orientation from multiple images”, *Optical Engineering*, January/February 1980, vol. 19 No.1

[14] Ruo Zhang, Ping-Sing Tsai, James Edwin Cryer, and Mubarak Shah, “Shape from Shading: A Survey”, *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol.21, NO.8, August 1999.

[15] Q.Zheng and R.Chellappa, “Estimation of Illuminant direction, Albedo, and Shape from Shading” *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 13, no. 7, pp. 680-702, July 1991.

COMPARISON OF PHOTOCONSISTENCY MEASURES USED IN VOXEL COLORING

Oğuz Özün[a], Ulaş Yılmaz[b]*, Volkan Atalay[a]

[a]Department of Computer Engineering, Middle East Technical University, Turkey – {oguz, volkan}@ceng.metu.edu.tr
[b]Computer Vision & Remote Sensing, Berlin University of Technology, Germany – ulas@cs.tu-berlin.de

KEY WORDS: photoconsistency, voxel coloring, three-dimensional object modeling, standard deviation, Minkowsky distance, adaptive threshold, histogram, color caching.

ABSTRACT

A framework for the comparison of photoconsistency measures used in voxel coloring algorithm is described. With this framework, the results obtained in generalized voxel coloring algorithm using certain photoconsistency measures are discussed qualitatively and quantitatively. The photoconsistency measures are based on standard deviation, Minkowsky distance, adaptive threshold, histogram, and color caching. Quantitative measurement is performed with root-mean-square-error and normalized-cross-correlation-ratio. The results show that, the photoconsistency measures which require threshold(s) may produce better/worse results depending on the selected threshold(s). Also, modeling textured objects always produce better reconstruction results.

1 INTRODUCTION

The main goal of volumetric scene modeling approaches is to find out, whether a given point is on an object's surface in the scene or not. According to the information being used in reconstruction, these approaches are classified into two groups: *shape-from-silhouette* and *shape-from-photoconsistency*. In shape-from-silhouette approaches, the model is extracted using back projections of the silhouettes onto the images: Each back projected silhouette corresponds to a volume in the space; intersection of these volumes gives the model (Mülayim et al., 2003). In shape-from-photoconsistency approaches, on the other hand, the photoconsistency of light coming from a point in the scene is taken into account: If the light coming from a point in the scene is not photoconsistent, then this point cannot be on a surface in the scene. In both approaches, the space is modeled using volume elements, *voxels*. *Voxel coloring* and its variations (*space carving (Broadhurst and Cipolla, 2000, Kutulakos and Seitz, 2000)*, *generalized voxel coloring (Culbertson et al., 1999)*, *multihypothesis voxel coloring (Steinbach et al., 2000)*, etc.) are in shape-from-photoconsistency group. These variations differ in the way they determine the visibility of a given voxel. When the same photoconsistency measure is used, a significant difference in the output is not observed. So to say, in voxel coloring algorithm and in its variations, the reconstructed model highly depends on the used photoconsistency measure. In this study, generalized voxel coloring algorithm is used to compare the effect of different photoconsistency measures. The experiments are performed on 3 image sequences. Results obtained through different photoconsistency measures are then compared using image comparison techniques, *root-mean-square-error* and *normalized-cross-correlation-ratio*.

The organization of the paper is as follows: In the following section, voxel coloring algorithm used in this study is explained in detail. This section is followed by Section 3 in which photoconsistency measures used in voxel coloring are presented. Qualitative comparison of photoconsistency measures is described in Section 4. Results obtained in the framework of this study are given in Section 5. The paper concludes with Section 6, in which the results are discussed and comments about the presented photoconsistency measures are made.

2 SHAPE-FROM-PHOTOCONSISTENCY

Depending on surface properties, lighting conditions and viewing direction, color of light coming from a point in the scene varies. Nevertheless, different observations should be coherent. In other words, a point on a surface should be seen with similar colors when it is not occluded. This phenomenon is called *photoconsistency*. Shape-from-photoconsistency approaches of volumetric scene reconstruction are based on this property of surfaces. If the light coming from an unconcluded portion of the scene is photoconsistent, then this point should be on a surface in the scene. Otherwise, it should be empty. This claim is based on a couple of assumptions: Objects in the scene have Lambertian surfaces, and projection of any point in the scene on the images can be computed (Kutulakos and Seitz, 2000).

In this study, due to its easiness in implementation, generalized voxel coloring algorithm is used. In order to improve computational cost, convex hull of the object is computed, and it is used as the input of voxel coloring algorithm. For each view, voxel space is divided into layers according to the distance to the view point. From nearest to furthest, layers are traversed, and the voxels are checked for visibility. Initially, all pixels in all images are unmarked. At each level, visible pixels are marked, so that the visibility of voxels at the following layers can be decided. Assume that a voxel v at some further layer is visited. Compared to the other voxels which are at nearer layers, it should have less number of visible pixels in each image. Having extracted visible pixels from all images, a set of colors is obtained. If this set is photoconsistent, then the voxel is on the surface. The pixels are marked and next voxel is processed. If the set is not photoconsistent, then the voxel is removed from the model. The ordinal visibility constraint and traversal of voxels based on layers makes it possible to use pixel marking as an efficient tool to handle occlusions: Single sweep through the voxel space is enough to reduce the voxel set to a more photoconsistent state. This procedure is iterated until all voxels are photoconsistent.

3 PHOTOCONSISTENCY MEASURES

As it is mentioned in the previous section, removal or coloring decision of a voxel depends on the set of colors, which is obtained by projecting the voxel onto the images. Given n images,

*Corresponding author.

assume that $I_0, I_1, ..., I_{p-1}$ are the images in which voxel v is not occluded. Then, for v, a nonempty set of colors, π, is extracted from the images as shown in Equation 1 where π_j is the set of colors extracted for v from image j, and $c_0, c_1, ..., c_m$ are the extracted color values.

$$\pi = \bigcup_{j=0}^{p-1} \pi_j = \{c_0, c_1, ..., c_m\} \tag{1}$$

Once this set is extracted, its photoconsistency can be defined in various ways. Some criteria used in the literature are as follows:

1. Standard deviation (Seitz and Dyer, 1999, Kutulakos and Seitz, 2000, Culbertson et al., 1999, Broadhurst and Cipolla, 2000),
2. adaptive threshold (Slabaugh et al., 2004),
3. Minkowsky distance (Slabaugh et al., 2001),
4. histogram (Slabaugh et al., 2004),
5. color caching (Chhabra, 2001).

3.1 Standard Deviation

Using standard deviation σ as a photoconsistency measure is proposed by Seitz and Dyer (Seitz and Dyer, 1999). If the standard deviation σ of π is less than a threshold τ, π is photoconsistent, which means the v is on the surface.

$$consistent(v) = \left\{ \begin{array}{ll} true, & \sigma < \tau \\ false, & otherwise \end{array} \right\} \tag{2}$$

3.2 Adaptive Threshold

The consistency measure based on standard deviation works well when the object's surface color is homogeneous. Otherwise, it easily diverges. In order to handle this problem, Slabaugh et al. (Slabaugh et al., 2004) proposed a new photoconsistency measure called adaptive threshold: If a voxel is on an edge or on a textured surface, then the variation of the extracted color values is higher. Larger thresholds should be used so that the photoconsistency measure converges. Assume that a voxel v, which is on an edge or on a textured surface, is visible from p views, $I_0, I_1, ..., I_{p-1}$. Then, the color sets extracted from these images for v are $\pi_0, \pi_1, ..., \pi_{p-1}$, and the standard deviations of these sets are $\sigma_0, \sigma_1, ..., \sigma_{p-1}$, respectively. These standard deviations should be high, since v is on an edge or on a textured surface. The average of these standard deviations, which is given in Equation 3, should also be high. By using this observation, authors define a new photoconsistency measure called adaptive threshold as given in Equation 4.

$$\overline{\sigma} = \frac{1}{p-1} \sum_{j=0}^{p-1} \sigma_j \tag{3}$$

$$consistent(v) = \left\{ \begin{array}{ll} true, & \sigma < \tau_1 + \overline{\sigma}\tau_2 \\ false, & otherwise \end{array} \right\} \tag{4}$$

This measure brings an important advantage over the measure based on standard deviation: The value of threshold is variable according to the place of the voxel. If the voxel is on the edge or textured surface, this situation can be detected with high standard deviation in each image, and a greater threshold can be used. Adaptive threshold measure is actually superset of the measure based on standard deviation. The need for 2 thresholds is its main disadvantage.

3.3 Minkowsky Distance

Photoconsistency of a set can also be defined using Minkowsky distances, L_1, L_2 and L_∞. Minkowsky distance between two points x and y in $\Re^k$ is given in Equation 5.

$$L_p(x, y) = \left(\sum_{i=0}^{k} |x_i - y_i| \right)^{\frac{1}{p}} \tag{5}$$

Assume that a voxel v is visible from p views, $I_0, I_1, ..., I_{p-1}$, and the color sets extracted from these images are $\pi_0, \pi_1, ..., \pi_{p-1}$. Every color entity in each of these color sets should be in a certain distance to the color entities of the other sets. Through this idea, photoconsistency of v is defined as in Equation 6. The distance between two color sets is given in Equation 7.

$$consistent(v) = \left\{ \begin{array}{ll} true, & \forall_{i,j}, consistent_{i,j}(v) \\ false, & otherwise \end{array} \right\} \tag{6}$$

$$consistent_{i,j}(v) = \left\{ \begin{array}{ll} true, & \forall_{c_l \in \pi_i, c_m \in \pi_j}, L_p(c_l, c_m) < \tau \\ false, & otherwise \end{array} \right\} \tag{7}$$

The most important benefit of using Minkowsky distance as a photoconsistency measure is the following. During the photoconsistency check, if the voxel is found to be inconsistent, there is no need to continue to check photoconsistency of that voxel. That means, having found a pair of colors whose difference is greater or equal to the threshold, voxel cannot be photoconsistent.

3.4 Histogram

In order to get rid of thresholds, Slabaugh et al (Slabaugh et al., 2004) proposed a new photoconsistency measure based on color histogram. Photoconsistency check is performed in two steps: histogram construction and histogram intersection. In the first step, visible pixels of the voxel v are extracted and a color a histogram is constructed for each image. Next step is photoconsistency check. To check whether v is photoconsistent or not, all pairs of histograms of v have to be compared: Two views i and j of v are photoconsistent with each other, if their histograms H_{v_i} and H_{v_j} match. A matching function, $match(H_{v_i}, H_{v_j})$, which compares two histograms, and returns a similarity value should be defined. Then the decision about the photoconsistency v is made according to the measure given in Equation 8.

$$consistent(v) = \left\{ \begin{array}{ll} true, & \forall_{i,j}, match(H_{v_i}, H_{v_j}) \neq 0 \\ false, & otherwise \end{array} \right\} \tag{8}$$

The advantage of histogram based photoconsistency measure is that there is no need for a preset threshold. Furthermore, paired tests can be very efficient in some circumstances. For instance, if the voxel is found to be inconsistent for a pair, there is no need to test other pairs of views for photoconsistency.

3.5 Color Caching

Color caching is a photoconsistency measure proposed by Chhabra et al. (Chhabra, 2001). It brings a solution to the limitations caused by Lambertian assumption. Photoconsistency of a voxel is checked twice before it is removed: If a voxel is found inconsistent in the first step, it is passed to the second step for another check. At the first step, surface parts which show Lambertian reflectance properties are tested. Those surface parts which fail Lambertian assumptions for some reason (material properties, viewing orientation, position of the light sources, etc.) are tested at the second step. The irradiance from these parts can be inconsistent. In order to prevent carving of these parts, before carving a voxel, it should be checked with another measure which takes care of viewing orientation. Given an image, for each voxel v, a cache is constructed. Each cache holds the visible colors of v from the relevant image. Having constructed color caches for each image, these caches are checked to find out, if there is a similar or a common color in all pairs of caches. If there is a match between all pairs of caches, v is labeled as consistent. If there is any pair of views whose caches do not contain a similar or a common color, v is labeled as inconsistent. Chhabra (Chhabra, 2001) defines similarity measure between two colors $c_i = (R_i, G_i, B_i)$ and $c_j = (R_j, G_j, B_j)$ as in Equation 9, and similarity of two images I_i and I_j as in Equation 10.

$$similarity(c_i, c_j) = \left\{ \begin{array}{ll} true, & \Delta_{i,j} \leq \tau_1 \\ false, & otherwise \end{array} \right\} \tag{9}$$

$$\Delta_{i,j} = \sqrt{(R_i - R_j)^2 + (B_i - B_j)^2 + (B_i - B_j)^2}$$

$$similarity(I_i, I_j) = \left\{ \begin{array}{ll} true, & \exists_{c_l \in cache_i} \exists_{c_m \in cache_j} similarity(c_l, c_m) \\ false, & otherwise \end{array} \right\} \tag{10}$$

So, the photoconsistency of voxel v is defined as in Equation 11.

$$consistent(v) = \left\{ \begin{array}{ll} true, & \forall_{i,j} consistent(I_i, I_j) \\ false, & otherwise \end{array} \right\} \tag{11}$$

4 COMPARISON

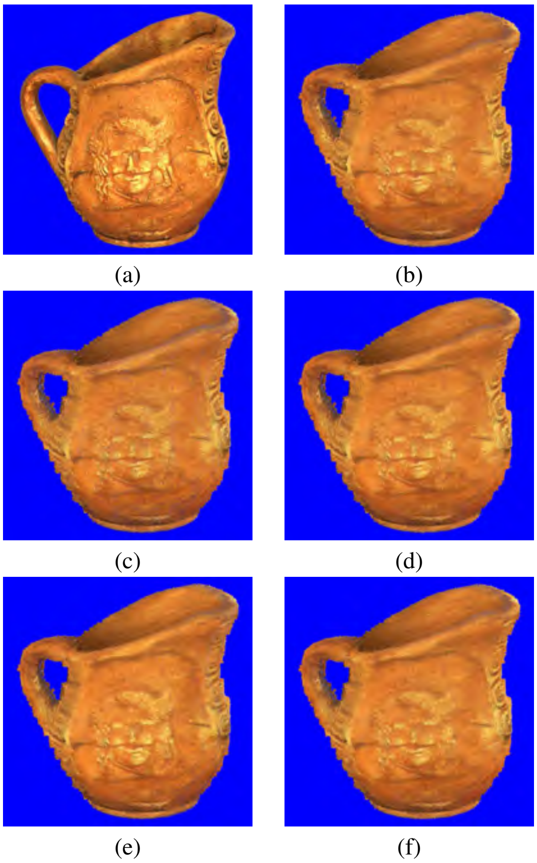
In order to compare photoconsistency measures, one should be able to measure quantitatively the quality of the reconstructed models. In this study, similarity between captured and rendered images of model is used to obtain a quantitative quality measure. The images are compared using root-mean-square-error (RMSE) and normalized cross-correlation-ratio (NCCR). Definitions of these measures are given in Equations 12 and 13, where M and N are the image dimensions, and G is the maximum intensity value.

$$RMSE(A, B) = \frac{1}{G\sqrt{MN}} \sqrt{\sum_{i,j}^{M,N} (A_{ij} - B_{ij})^2} \tag{12}$$

$$NCCR(A, B) = 1 - \frac{\sum_{i,j}^{M,N} A_{ij} B_{ij}}{\sqrt{(\sum_{i,j}^{M,N} A_{ij}^2)(\sum_{i,j}^{M,N} B_{ij}^2)}} \tag{13}$$

5 EXPERIMENTAL RESULTS

Photoconsistency measures are tested using 3 objects: “cup”, “star”, and “box”. Voxel space resolution is set to 450×450×450 for all objects. Its handle makes the **“cup”** object hard to model. Furthermore, the available texture information is not adequate to obtain good results. The image sequence consists of 18 images, 16 of which are used for reconstruction and 2 for testing. Reconstructed models are shown in Figure 1 and measured quality of the reconstruction is tabulated in Table 1 and Table 2. There is no significant difference between the reconstructed models quantitatively. Qualitative results support this result. Histogram based photoconsistency measure would be the best choice for this image sequence, since there is no need for threshold tuning in this approach.



(a) (b) (c) (d) (e) (f)

Figure 1: (a) Original image, and artificially rendered images of “cup” object obtained using (b) standard deviation, (c) histogram, (d) adaptive threshold, (e) L_1 norm, and (f) color caching.

Image No	Measure	$NCCR$ (%)	$RMSE$ (%)
08	standard deviation	2.96	10.24
	histogram	2.95	10.22
	adaptive threshold	2.95	10.22
	L_1 norm	3.00	10.31
	color caching	2.95	10.21
17	standard deviation	1.68	7.99
	histogram	1.69	8.04
	adaptive threshold	1.68	8.00
	L_1 norm	1.86	8.43
	color caching	1.67	7.97

Table 1: Error analysis using test images for “cup” object.

Measure	$NCCR$ (%)	$RMSE$ (%)
standard deviation	1.55	7.37
histogram	1.58	7.45
adaptive threshold	1.55	7.37
L_1 norm	1.63	7.56
color caching	1.56	7.38

Table 2: Average error for "cup" object.

In order to test the photoconsistency measures on an object with a simple geometry, **"box"** object is used. 12 images are taken around the object and all of them are used during the reconstruction. In Figure 2 the result obtained using L_1 norm as photoconsistency measure is illustrated. Measured quality of the reconstruction is tabulated in Table 3. Rather than its quantitative results, the visual quality of the reconstructed model gives a clue. It is about the success of shape-from-photoconsistency approaches in general: The finer the voxel space, the better is the reconstruction and the more is the computational complexity.

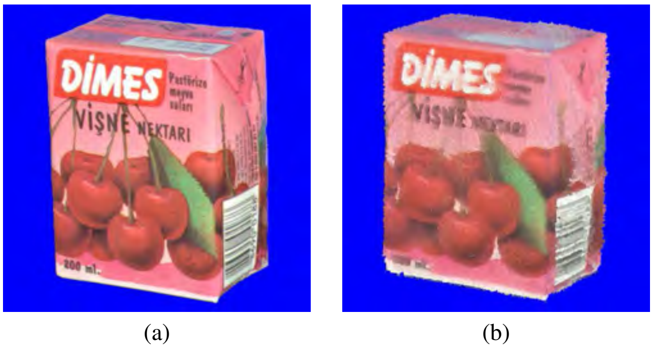


Figure 2: (a) Original image, and (b) artificially rendered image of "box" object obtained using L_1 norm.

Measure	$NCCR$ (%)	$RMSE$ (%)
standard deviation	2.15	10.73
histogram	2.17	10.85
adaptive threshold	2.09	10.57
L_1 norm	2.15	10.70
color caching	2.09	10.61

Table 3: Average error for "box" object.

"star" object is a good example of objects that has not texture information but a complex geometry. The image sequence for this object consists of 18 images, 9 of which are used for reconstruction and 9 for testing. Reconstructed models are shown in Figure 3 and measured quality of the reconstruction is tabulated in Table 4. Adaptive threshold seem to produce a better result than the others. Selecting low values for the thresholds causes overcarving. On the other hand, the higher is the threshold, the coarser is the reconstructed model. There is a high dependency on selecting proper thresholds for poor-textured objects. There is no need for a threshold in histogram-based method. However, in this case, lack of texture information causes some voxels not to be carved. Uncarved voxels have the color blue, i. e. the background color. Similar effect is also observed when color caching is used as photoconsistency measure.

6 CONCLUSIONS

A framework for the comparison of photoconsistency measures used in voxel coloring algorithm is described. Reconstruction results of 3 objects, which are obtained using generalized voxel coloring algorithm are discussed. The methods, in which thresholds are used, generally give the best results, if suitable thresholds are set. Better thresholds can be found empirically. Standard deviation gives appropriate results for the voxels, which are at the edges on the images or which are projected onto highly-textured regions of the images. Using adaptive thresholds, the threshold is changed according to the position of the voxel. But this change is controlled by another threshold, which is actually the bottleneck of the approach. The second threshold prevents carving voxels which are at the edge or highly-textured. When the second threshold is not a suitable value, it might generate cusps in the final model. When objects to be modeled are highly-textured, it is better to use histogram-based photoconsistency measure. In this approach, there is no need for a threshold. Minkowsky distance does not have a specific benefit as a photoconsistency measure. However, Minkowsky distance is a monotonically increasing function. So, if the color set is found to be inconsistent from some views, then there is no need to check the visible pixels from other views. This speeds the computation up. Color caching is an appropriate photoconsistency measures to eliminate the highlights. However, it is also possible to eliminate the specularities using background surface and selected lighting.

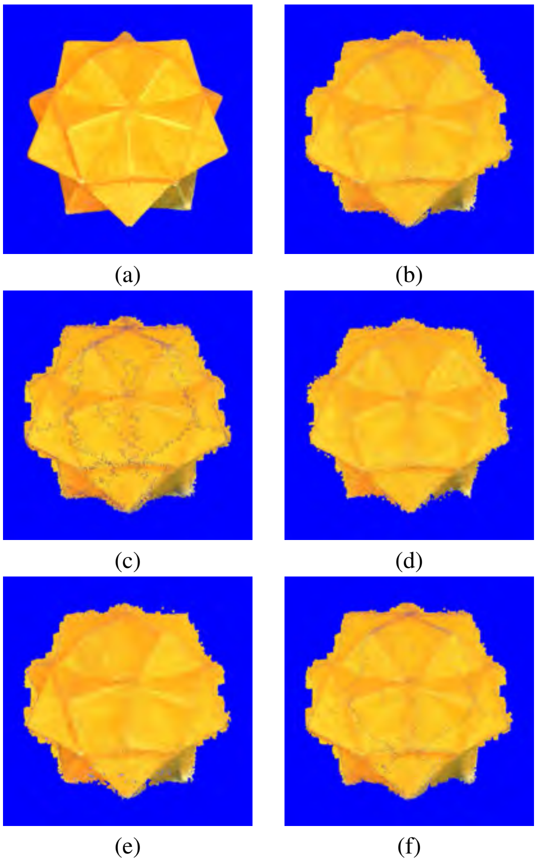


Figure 3: (a) Original image, and artificially rendered images of "star" object obtained using (b) standard deviation, (c) histogram, (d) adaptive threshold, (e) L_1 norm, and (f) color caching.

Measure	$NCCR$ (%)	$RMSE$ (%)
standard deviation	0.85	6.39
histogram	0.91	6.54
adaptive threshold	0.74	5.89
L_1 norm	0.85	6.35
color caching	0.86	6.40

Table 4: Average error for "star" object.

REFERENCES

Broadhurst, A. and Cipolla, R., 2000. A statistical consistency check for the space carving algorithm. In: British Machine Vision Conference, pp. 282–291.

Chhabra, V., 2001. Reconstructing specular objects with image based rendering using color caching. Master's thesis, Worcester Polytechnic Institute.

Culbertson, W. B., Malzbender, T. and Slabaugh, G. G., 1999. Generalized voxel coloring. In: International Conference on Computer Vision, Vision Algorithms Theory and Practice, pp. 100–115.

Kutulakos, K. N. and Seitz, S. M., 2000. A theory of shape by space carving. International Journal of Computer Vision 38(3), pp. 199–218.

Mülayim, A. Y., Yılmaz, U. and Atalay, V., 2003. Silhouette-based 3d model reconstruction from multiple images. IEEE Transactions on Systems, Man and Cybernetics Part B: Cybernetics Special Issue on 3-D Image Analysis and Modeling 33(4), pp. 582–591.

Seitz, S. M. and Dyer, C. R., 1999. Photorealistic scene reconstruction by voxel coloring. International Journal of Computer Vision 35(2), pp. 157–173.

Slabaugh, G. G., Culbertson, W. B., Malzbender, T. and Schafer, R. W., 2001. A survey of volumetric scene reconstruction methods from photographs. In: Volume Graphics, pp. 81–100.

Slabaugh, G. G., Culbertson, W. B., Malzbender, T., Stevens, M. R. and Schafer, R. W., 2004. Methods for volumetric reconstruction of visual scenes. International Journal of Computer Vision 57(3), pp. 179–199.

Steinbach, E., Girod, B., Eisert, P. and Betz, A., 2000. 3-d object reconstruction using spatially extended voxels and multi-hypothesis voxel coloring. In: International Conference on Pattern Recognition, pp. 774–777.

3D SURFACE RECONSTRUCTION BASED ON COMBINED ANALYSIS OF REFLECTANCE AND POLARISATION PROPERTIES: A LOCAL APPROACH

Pablo d'Angelo and Christian Wöhler

DaimlerChrysler Research and Technology, Machine Perception
P. O. Box 2360, D-89013 Ulm, Germany

KEY WORDS: surface reconstruction, polarization vision, shape from shading, quality inspection

ABSTRACT

An image-based 3D surface reconstruction technique based on simultaneous evaluation of reflectance and polarisation features is introduced in this paper. The proposed technique is suitable for single and multi-image (photopolarimetric stereo) analysis. It is especially suited for the difficult task of 3D reconstruction of rough metallic surfaces with non-Lambertian reflectance. The reflectance and polarisation properties are used to determine the surface gradients individually for each image pixel. The presented multi-image technique is invariant to variations of the surface albedo. We evaluate our algorithm based on synthetic ground truth data as well as on a raw forged iron surface. The results we obtain for the real world example demonstrate the applicability of our method in the domain of industrial quality inspection.

1 INTRODUCTION

Three-dimensional reconstruction of surfaces has become an important technique in the context of industrial quality inspection. In the field of optical metrology, the currently most widely used active approaches are primarily based on *projection of structured light* (Batlle et al., 1998). While such methods are accurate, they require a highly precise mutual calibration of cameras and structured light sources. Multiple structured light sources may be needed for 3D reconstruction of non-convex surfaces. Hence, for inline quality inspection of industrial part surfaces, less intricate passive image-based techniques are desirable.

A well-known passive image-based surface reconstruction method is *shape from shading*. This approach aims at deriving the orientation of the surface at each pixel by using a model of the reflectance properties of the surface and knowledge about the illumination conditions (Horn and Brooks, 1989). The integration of shadow information into the shape from shading formalism and applications of such methods in the context of fast inline quality inspection have been demonstrated (Wöhler and Hafezi, 2005).

A further approach to reveal the 3D shape of a surface is to utilise polarisation data. Most current literature concentrates on dielectric surfaces, as for smooth dielectric surfaces, the direction and degree of polarisation as a function of surface orientation are governed by elementary physical laws (Miyazaki et al., 2004). For smooth dielectric surfaces a 3D surface reconstruction framework is proposed relying on the analysis of the polarisation state of reflected light, the surface texture, and the locations of specular reflections (Miyazaki et al., 2003). In previous work, reflectance and polarisation properties of metallic surfaces are examined, but no physically motivated polarisation model is derived (Wolff, 1991). Furthermore, it has been demonstrated that polarisation information can be used to determine surface orientation (Rahmann and Canterakis, 2001). Applications of such *shape from polarisation* approaches to real-world scenarios, however, are rarely described in the literature. A variational combined shape from shading and polarisation algorithm relying on the minimisation of a global error function is introduced in (d'Angelo and Wöhler, 2005) and applied to 3D reconstruction of metallic surfaces.

In this paper we present an image-based method for 3D surface reconstruction by simultaneous evaluation of information about reflectance and polarisation. This method will be applied relying on a pair of polarisation images of the surface (*photopolarimetric stereo*). It is assumed that the scene is illuminated by unpolarised point light sources situated at known locations. The reflectance and polarisation properties of the surface material are measured over a wide range of surface orientations by evaluating a series of images acquired through a linear polarisation filter under different rotation angles, respectively. Parameterised phenomenological models will then be fitted to the obtained measurements. Both reflectance and polarisation features are used to determine the surface gradient individually for each image pixel, without introducing global constraints like smoothness (d'Angelo and Wöhler, 2005).

We systematically evaluate our method on a synthetically generated surface in order to examine its accuracy, convergence behaviour, and noise-robustness. We furthermore investigate the accuracy of our 3D reconstruction technique for the real-world example of a raw forged iron surface.

2 REFLECTANCE AND POLARISATION MODELS

2.1 Measurement of reflectance properties

The pixel intensity $I(u, v)$ observed by a camera is governed by the *reflectance function* of the surface material,

$$I(u, v) = R\left(\vec{n}(u, v), \vec{s}, \vec{v}\right), \tag{1}$$

which depends on the surface normal $\vec{n}$, the illumination direction $\vec{s}$, and the direction $\vec{v}$ to the camera. We assume that both light source and camera are situated at infinite distance from the object, such that $\vec{s}$ and $\vec{v}$ are assumed to be constant. In the following, the surface normal $\vec{n}$ will be represented in *gradient space* by the directional derivatives $p = z_x$ and $q = z_y$ of the surface function $z(x, y)$ with $\vec{n} = (-p, -q, 1)^T$. We define accordingly $\vec{s} = (-p_s, -q_s, 1)^T$ and $\vec{v} = (-p_v, -q_v, 1)^T$ in gradient space.

A well-known special case is the Lambertian reflectance function $R\left(\vec{n}, \vec{s}\right) = \rho(u, v) \cos\theta_i$ with $\cos\theta_i = \vec{n} \cdot \vec{s} / \left(|\vec{n}||\vec{s}|\right)$ and $\rho(u, v)$ as the *surface albedo*. In this paper, however, we regard

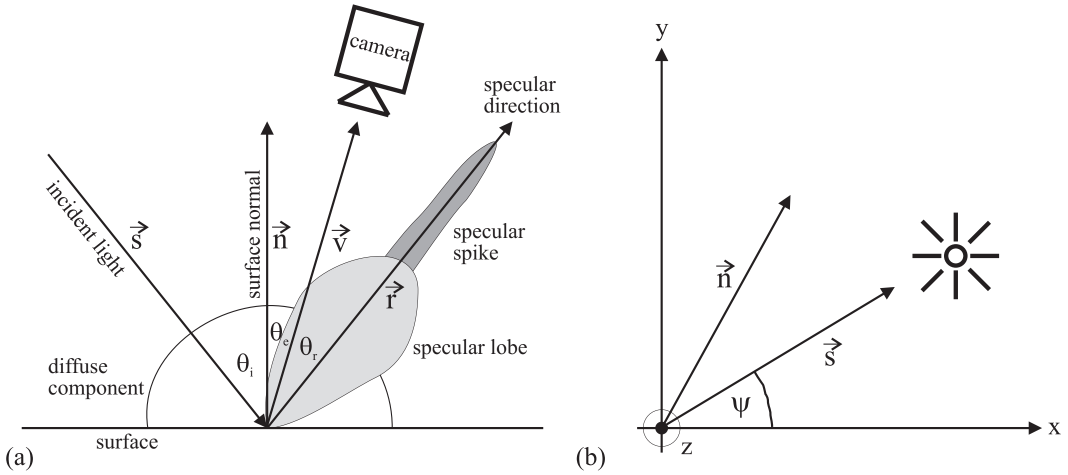


Figure 1: (a) Plot of the three reflectance components. (b) Definition of the world coordinate system and the azimuth angle ψ.

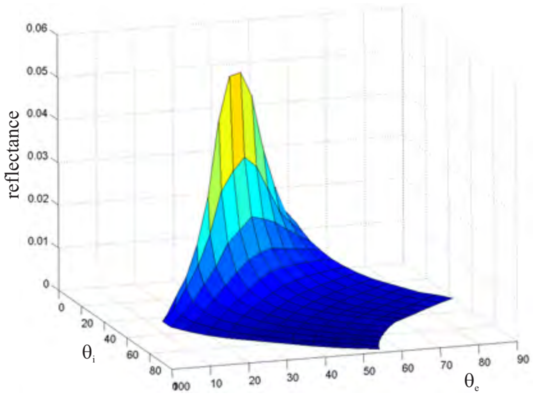


Figure 2: Left: Measured reflectance of a raw forged iron surface for $\alpha = 75°$. The parameters of the reflectance function (cf. Eq. 2) amount to $\sigma_1 = 3.85$, $m_1 = 2.61$, $\sigma_2 = 9.61$, and $m_2 = 15.8$, where the specular lobe is described by σ_1 and m_1 and the specular spike by σ_2 and m_2.

metallic surfaces with a strongly non-Lambertian reflectance behaviour. We will assume that the reflectance of a typical rough metallic surface consists of three components: a diffuse (Lambertian) component, the *specular lobe*, and the *specular spike* (Nayar et al., 1991). The diffuse component is generated by internal multiple scattering processes. The specular lobe, which is caused by single reflection at the surface, is distributed around the specular direction and may be rather broad. The specular spike is concentrated in a small region around the specular direction and represents mirror-like reflection, which is dominant in the case of smooth surfaces. Fig. 1a illustrates the three components of the reflectance function. We define an analytical form for the reflectance for which we perform a least-mean-squares fit to the measured reflectance values, depending on the incidence angle θ_i, the angle θ_r between the specular direction $\vec{r}$ and the viewing direction $\vec{v}$ (cf. Fig. 1a), and the phase angle α between the vectors $\vec{s}$ and $\vec{v}$:

$$R(\theta_i, \theta_r, \alpha) = \rho \left[\cos\theta_i + \sum_{n=1}^{N} \sigma_n \cdot (\cos\theta_r)^{m_n} \right]. \quad (2)$$

The angle θ_r can be expressed in terms of incidence angle, emission angle, and phase angle according to

$$\cos\theta_r = 2\cos\theta_i \cos\theta_e - \cos\alpha, \quad (3)$$

such that our phenomenological reflectance model only depends on the incidence angle θ_i, the emission angle θ_e, and the phase angle α. Note that $\alpha \leq \theta_i + \theta_e$ in the general three-dimensional case. For $\theta_r > 90°$ only the diffuse component is considered. The albedo ρ is assumed to be constant over the surface. The shapes of the specular components of the reflectance function are approximated by $N = 2$ terms proportional to powers of $\cos\theta_r$. The coefficients $\{\sigma_n\}$ denote the strength of the specular components relative to the diffuse component, while the parameters $\{m_n\}$ denote their widths. All introduced phenomenological parameters generally depend on the phase angle α. For our measurements we use a goniometer to adjust the angles θ_i and θ_e. The phase angle α between the vectors $\vec{s}$ and $\vec{v}$ is assumed to be constant over the image.

For each configuration of θ_i, θ_e, and α, we acquire a high dynamic range image by combining several images taken with different shutter times. The reflectance of the sample surface under the given illumination conditions is then obtained by computing the average greyvalue over an area in the high dynamic range image that contains a flat part of the sample surface. A reflectance measurement typical for raw forged or cast iron surfaces is shown in Fig. 2 for $\alpha = 75°$.

2.2 Measurement of polarisation properties

In our scenario, the incident light is unpolarised. For smooth metallic surfaces the light remains unpolarised after reflection at the surface. Rough metallic surfaces, however, partially polarise the reflected light (Wolff, 1991). The measurement of the polarisation properties of the surface is similar to the reflectance measurement. For each configuration of goniometer angles, five high dynamic range images are acquired through a linear polarisation filter at multiple orientation angles ω between 0° and 180°. For each filter orientation ω, an average pixel intensity over an image area containing a flat part of the sample surface is computed as described in Section 2.1. To the measured pixel intensities we fit a sinusoidal function (Wolff, 1991) of the form

$$I(\omega) = I_c + I_v \cos(\omega - \Phi). \quad (4)$$

The filter orientation Φ for which maximum intensity $I_c + I_v$ is observed corresponds to the *polarisation angle* ($\omega = \Phi$). The *polarisation degree* amounts to $D = I_v / I_c$. In principle, three measurements would be sufficient to determine the three parameters I_c, I_v, and Φ, but the fit becomes less noise-sensitive and thus more accurate when more measurements are used. The parameter I_c represents the reflectance of the surface.

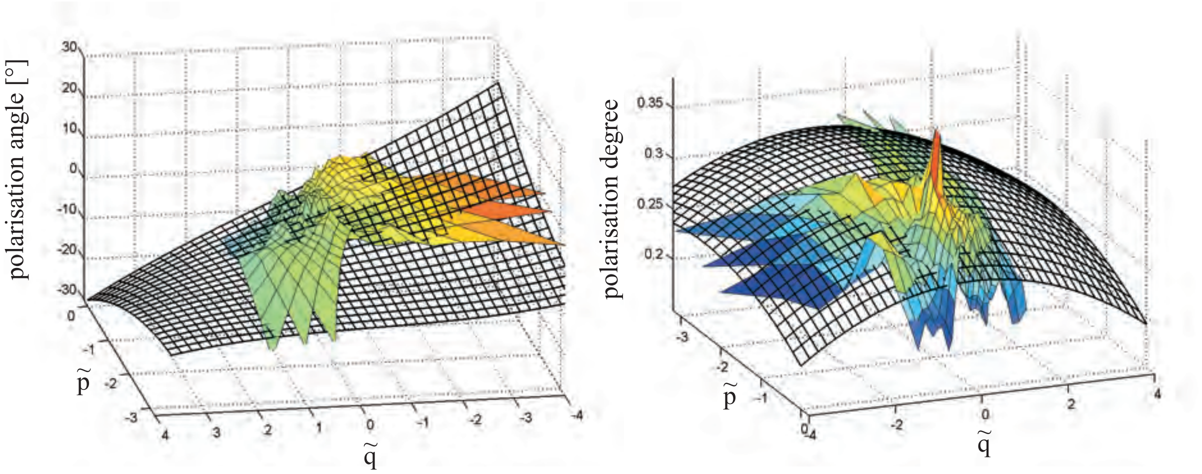


Figure 3: Measured and modelled polarisation properties of a raw forged iron surface. Left: polarisation angle. Right: polarisation degree.

According to Fig. 1b, the rotation angles of the goniometer define the surface normal $\tilde{\vec{n}} = (-\tilde{p}, -\tilde{q}, 1)$ of the sample surface in a coordinate system with positive x and zero y component of the illumination vector $\vec{s}$, corresponding to $p_s < 0$ and $q_s = 0$. Without loss of generality we will in the following assume a viewing direction $\vec{v} = (0,0,1)^T$. The surface normal $\vec{n}$ in the world coordinate system, in which the azimuth angle of the light source is denoted by the angle ψ, is related to $\tilde{\vec{n}}$ by a rotation $R_z(\psi)$ around the z axis, leading to

$$\begin{aligned} \tilde{p} &= p\cos\psi + q\sin\psi \\ \tilde{q} &= -p\sin\psi + q\cos\psi. \end{aligned} \tag{5}$$

Due to the lack of an accurate physically motivated model for the polarisation properties of rough metallic surfaces, we perform a polynomial fit in terms of the surface gradients $\tilde{p}$ and $\tilde{q}$ to the measured values of the polarisation angle Φ and degree D. In this framework, the modelled polarisation angle R_Φ is represented by an incomplete third-degree polynomial of the form

$$R_\Phi(\tilde{p},\tilde{q}) = a_\Phi + b_\Phi\tilde{p}\tilde{q} + c_\Phi\tilde{q} + d_\Phi\tilde{p}^2\tilde{q} + e_\Phi\tilde{q}^3. \tag{6}$$

The constant offset a_Φ can be made zero by correspondingly defining the zero position of the orientation angle ω of the linear polarisation filter. Eq. (6) is antisymmetric in $\tilde{q}$ with respect to a_Φ. At the same time, $R_\Phi(\tilde{p},\tilde{q}) = a_\Phi = \text{const}$ for $\tilde{q} = 0$, corresponding to coplanar vectors $\vec{n}$, $\vec{s}$, and $\vec{v}$. These properties are required for geometrical symmetry reasons as long as the interaction between the incident light and the surface material can be assumed to be isotropic.

The observed polarisation degree R_D is represented in an analogous manner by an incomplete second-degree polynomial of the form

$$R_D(\tilde{p},\tilde{q}) = a_D + b_D\tilde{p} + c_D\tilde{p}^2 + d_D\tilde{q}^2. \tag{7}$$

In this case, symmetry in $\tilde{q}$ is imposed for geometrical reasons, once more due to the assumed isotropy of light-surface interaction. Fig. 3 illustrates the polarisation properties of a raw forged iron surface at a phase angle of $\alpha = 75°$ along with the polynomial fits according to Eqs. (6) and (7).

3 3D SURFACE RECONSTRUCTION USING REFLECTANCE AND POLARISATION

Well-known approaches to reflectance-based 3D surface reconstruction are *shape from shading* and *photometric stereo*, the latter term referring to the evaluation of multiple images of the surface acquired under different illumination conditions. These methods aim at determining the surface gradient field, which is then integrated in order to obtain the depth $z(u,v)$. In this section we will extend this approach by introducing polarisation information.

The reflectance function as well as polarisation angle and degree can be expressed in terms of the surface gradients $p(u,v)$ and $q(u,v)$:

$$\begin{aligned} I(u,v) &= R\left(p(u,v), q(u,v)\right) && (8) \\ \Phi(u,v) &= R_\Phi\left(p(u,v), q(u,v)\right) && (9) \\ D(u,v) &= R_D\left(p(u,v), q(u,v)\right) && (10) \end{aligned}$$

The representation of R in Eq. (8) is called *reflectance map* (Horn and Brooks, 1989). Provided that the model parameters of the reflectance and polarisation functions R, R_Φ, and R_D are known and measurements of intensity and polarisation properties are available for each image pixel, the surface gradients p and q can be obtained by solving the nonlinear system of equations (8)–(10). For this purpose we make use fo the Levenberg-Marquardt algorithm in the overdetermined case and the Powell dogleg method (Powell, 1970) otherwise. In the overdetermined case, the root of Eqs. (8)-(10) is determined in the least-mean-squares sense. The contributions from the different terms are then weighted according to the measurement errors, respectively, which we have determined to $\sigma_I = 10^{-3} I_{\text{spec}}$ with I_{spec} as the intensity of the specular reflections, $\sigma_\Phi = 0.2°$ and $\sigma_D = 0.01$. The surface profile $z(u,v)$ is derived from the resulting gradients $p(u,v)$ and $q(u,v)$ by means of numerical integration of the gradient field (Jiang and Bunke, 1997).

It is straightforward to extend this approach to photopolarimetric stereo because each light source provides an additional set of equations. Eq. (8) can only be solved, however, when the surface albedo $\rho(u,v)$ is known for each surface point. A constant albedo can be assumed in many applications. If this assumption is not valid, albedo variations will affect the accuracy of surface reconstruction.

For surfaces with unknown and non-uniform albedo it is possible to utilise two images acquired under different illumination conditions, such that Eq. (8) can be replaced by

$$\frac{I_1}{I_2} = \frac{R_1\left(p(u,v), q(u,v)\right)}{R_2\left(p(u,v), q(u,v)\right)} \tag{11}$$

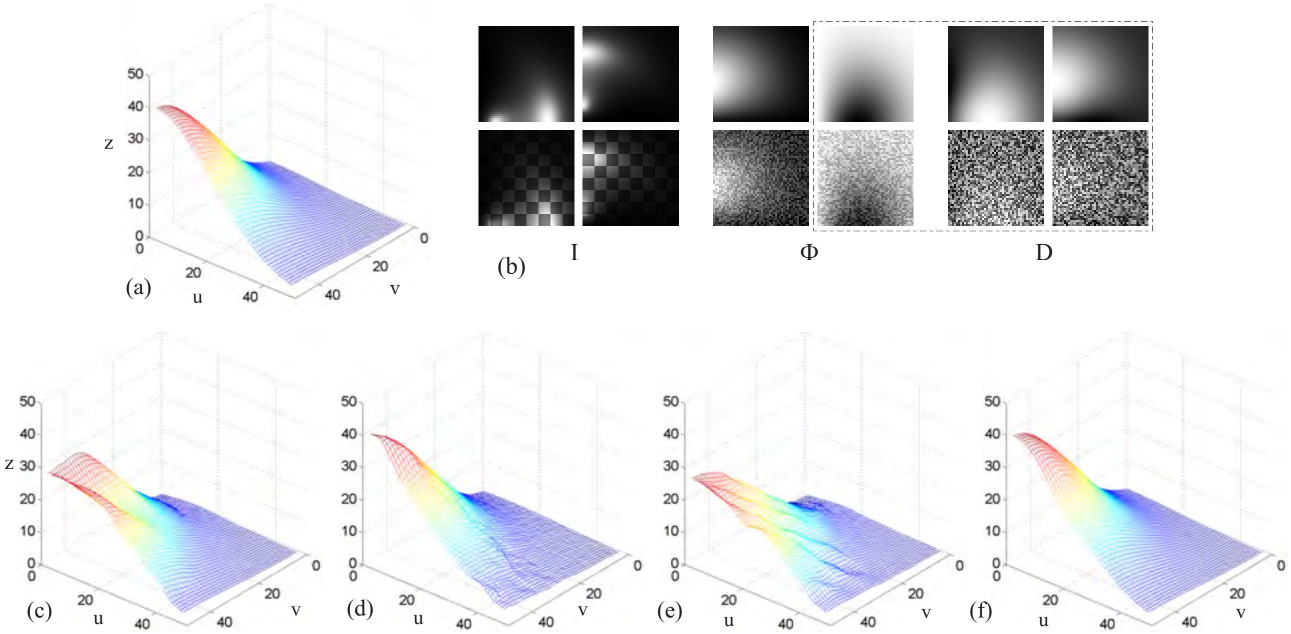


Figure 4: 3D reconstruction of a synthetically generated surface based on a photopolarimetric stereo image pair. (a) Ground truth. (b) From the left: Reflectance, polarisation angle and degree images, without and with non-uniform albedo, without and with noise, respectively (cf. Table 1). The second polarisation angle image and both polarisation degree images have been excluded from the analysis (cf. Section 4.1). Reconstruction result for noisy images of a surface with uniform albedo is shown in (c) using the albedo-dependent approach according to Eq. (8) and in (d) using the albedo-independent approach according to Eq. (11). Reconstruction results for a surface with non-uniform albedo in the noise-free case is shown in (e) for the albedo-dependent and in (f) for the albedo-independent approach.

In Eq. (11), the albedo cancels out. The quotient approach has been introduced in the context of photoclinometric analysis of planetary surfaces (McEwen, 1985) and has been integrated into the shape from shading formalism (Wöhler and Hafezi, 2005).

An advantage of the described local approach is that the 3D reconstruction result is not affected by additional constraints such as smoothness of the surface but directly yields the surface gradient field for each image pixel. A drawback, however, is the fact that due to the inherent nonlinearity of the problem, existence and uniqueness of a solution for p and q are not guaranteed for both the albedo-dependent and the albedo-independent case. But in the experiments presented in Section 4 we show that in practically relevant scenarios a reasonable solution for the surface gradient field and the resulting depth $z(u, v)$ is obtained even in the presence of noise.

4 EXPERIMENTAL RESULTS

4.1 Evaluation based on synthetic ground truth data

To examine the accuracy of 3D reconstruction, we apply the algorithm described in Section 3 to the synthetically generated surface shown in Fig. 4a. We still assume a perpendicular view on the surface along the z axis, corresponding to $\vec{v} = (0, 0, 1)^T$. The scene is illuminated by $L = 2$ light sources (one after the other) under an angle of 15° with respect to the horizontal plane at azimuth angles of $\psi^{(1)} = 0^\circ$ and $\psi^{(2)} = 90^\circ$, respectively. This setting results in identical phase angles $\alpha^{(1)} = \alpha^{(2)} = 75^\circ$ for the two light sources. The initial values for $p(u, v)$ and $q(u, v)$ must be provided relying on a-priori knowledge about the surface orientation. In the synthetic surface example, they are initialised with the value -0.5. It has been demonstrated that the initial gradients can be estimated using depth from defocus (d'Angelo and Wöhler, 2005).

The synthetic reflectance and polarisation angle images shown in Fig. 4b have been generated by means of the polynomial fits to the measured reflectance and polarisation properties presented in Figs. 2 and 3. We have used two synthetic surfaces for an evaluation of our reconstruction method, one surface with uniform albedo and one with spatially non-uniform albedo. In our experiments we have found that the behaviour of the polarisation degree of rough metallic surfaces tends to change significantly over the surface, due to local variations of the surface roughness (d'Angelo and Wöhler, 2005). In contrast, the behaviour of the polarisation angle does not show local variations over the surface. We thus decided not to make use of the polarisation degree in our practical experiments (cf. Section 4.2).

According to Fig. 3, the observed polarisation angles cover only a narrow interval. Hence, we have observed that the azimuth angle ψ must be known at an accuracy of about 0.1° if one desires to use both polarisation angle images for reconstruction, while the reflectance is less sensitive in this respect. As such accurate knowledge of ψ is difficult to obtain for practical reasons, we decided to use only one polarisation angle image.

The reconstruction results are shown in Fig. 4. The noise level amounts to 5 times the measurement errors given in Section 3. The corresponding RMS deviations from the ground truth for z, p, and q are given in Table 1. We have observed that for a significant fraction of pixels (about 25 percent) no solution of Eqs. (8)–(9) is obtained with the applied initialisation, presumably due to a small convergence radius. When Eq. (8) is replaced by Eq. (11), convergence is achieved for all pixels, leading to much higher accuracy of reconstruction. We have found experimentally that it is possible to decrease the reconstruction error obtained from Eq. (8) by decreasing the weight of the reflectance in the least-mean-squares optimisation. As seen from the RMS error of z, the quotient-based approach according to Eq. (11) yields the same re-

Table 1: Evaluation results on the synthetic ground truth example shown in Fig. 4 using both reflectance images but only one polarisation angle image.

Method	Albedo	RMS error (without noise)			RMS error (with noise)		
		z	p	q	z	p	q
I_1,I_2,Φ_1	uniform	3.2	0.20	0.18	3.2	0.20	0.19
I_1,I_2,Φ_1	non-uniform	4.1	0.25	0.24	4.1	0.26	0.24
I_1/I_2,Φ_1	uniform	0.4	0.10	0.00	0.8	0.24	0.16
I_1/I_2,Φ_1	non-uniform	0.4	0.10	0.00	0.8	0.24	0.17

Table 2: Evalutation results on synthetic ground truth data using various combinations of all available reflectance and polarisation data.

Method	Albedo	RMS error (without noise)			RMS error (with noise)		
		z	p	q	z	p	q
I_1,Φ_1	uniform	0.7	0.15	0.01	1.3	0.19	0.16
I_1,Φ_1	non-uniform	1.5	0.21	0.04	1.5	0.22	0.16
I_1,D_1	uniform	0.5	0.01	0.11	9.1	0.85	1.10
I_1,D_1	non-uniform	2.5	0.11	0.42	7.7	0.82	1.17
Φ_1,D_1	uniform	0.0	0.00	0.00	4.0	1.10	0.29
Φ_1,D_1	non-uniform	0.0	0.00	0.00	4.0	1.10	0.29
I_1,Φ_1,D_1	uniform	0.5	0.13	0.01	1.4	0.22	0.16
I_1,Φ_1,D_1	non-uniform	1.4	0.20	0.04	1.3	0.24	0.16
I_1,I_2	uniform	3.6	0.26	0.26	3.6	0.27	0.27
I_1,I_2	non-uniform	4.1	0.33	0.33	4.1	0.32	0.31
I_1,I_2,Φ_1,Φ_2	uniform	2.7	0.17	0.17	2.8	0.18	0.18
I_1,I_2,Φ_1,Φ_2	non-uniform	4.0	0.25	0.25	4.0	0.24	0.24
I_1,I_2,D_1,D_2	uniform	3.6	0.21	0.21	3.6	0.21	0.21
I_1,I_2,D_1,D_2	non-uniform	4.1	0.26	0.26	4.1	0.26	0.26
I_1,I_2,Φ_1,Φ_2,D_1,D_2	uniform	2.7	0.17	0.17	2.7	0.18	0.17
I_1,I_2,Φ_1,Φ_2,D_1,D_2	non-uniform	4.0	0.25	0.25	4.0	0.24	0.24
I_1/I_2,Φ_1,Φ_2	uniform	0.0	0.00	0.00	0.2	0.12	0.12
I_1/I_2,Φ_1,Φ_2	non-uniform	0.0	0.00	0.00	0.2	0.12	0.12
I_1/I_2,Φ_1,Φ_2,D_1,D_2	uniform	0.0	0.00	0.00	0.2	0.12	0.11
I_1/I_2,Φ_1,Φ_2,D_1,D_2	non-uniform	0.0	0.00	0.00	0.2	0.12	0.12

sults for the surfaces with uniform and non-uniform albedo, while the error increases when Eq. (8), assuming a uniform albedo, is used.

For comparison, we report in Table 2 the reconstruction accuracy obtained using various combinations of all available reflectance and polarisation data, including the polarisation degree. The values are computed both for a single set and for a pair of reflectance and polarisation images, respectively. We have found that a pair of intensity images alone is not sufficient for reasonably accurate 3D surface reconstruction. With both reflectance and polarisation angle images, the reconstruction results become virtually exact when Eq. (11) is used. Even with a single light source we obtain good reconstruction results when all available reflectance and polarisation data are used.

4.2 Application to a rough metallic surface

We will now describe the application of our photopolarimetric 3D reconstruction method to the raw forged iron surface of an automotive part. Image resolution was 0.30 mm per pixel. For each pixel, the polarisation properties are determined as described in Section 2. The 3D reconstruction result $z(u, v)$ along with the reflectance and polarisation images is shown in Fig. 5 for a flawless and a deformed part, respectively. As discussed in Section 4.1, the reconstruction is based on the quotient I_1/I_2 of the two reflectance images and one polarisation angle image. The surface gradients $p(u, v)$ and $q(u, v)$ are initialised with zero values. The difference between the two surfaces shows that some material is missing in the deformed part. This is due to a fault caused during the forging process. The offset between the two surfaces at the margin of the part amounts to 2.05 ± 0.05 mm along the surface normal, obtained by tactile measurement with a sliding calliper at the points indicated by the arrows in Fig. 5b. The 3D reconstruction yields a value of 2.1 mm (Fig. 5c), which is in good agreement. A cross-section of the same surface was measured with a laser focus profilometer and compared to the corresponding cross-section extracted from the reconstructed 3D profile (Fig. 5d). The RMS deviation amounts to 0.22 mm, corresponding to about two-thirds of a pixel.

5 SUMMARY AND CONCLUSION

In this paper we have presented an image-based method for 3D surface reconstruction relying on the simultaneous evaluation of reflectance and polarisation information for multiple images (photopolarimetric stereo). The reflectance and polarisation properties of the surface material have been obtained by means of a series of images acquired through a linear polarisation filter under different orientations. Analytic phenomenological models have been fitted to the obtained measurements, allowing for an integration of both reflectance and polarisation features into a unified local (pixel-wise) optimisation framework. The presented method has been evaluated based on a synthetically generated surface. The dependence of the accuracy of 3D reconstruction on the utilised reflectance and polarisation data is systematically examined. Furthermore we have applied our method to the difficult real-world scenario of 3D reconstruction of a surface section of a raw forged iron part. We have shown that our approach is suitable

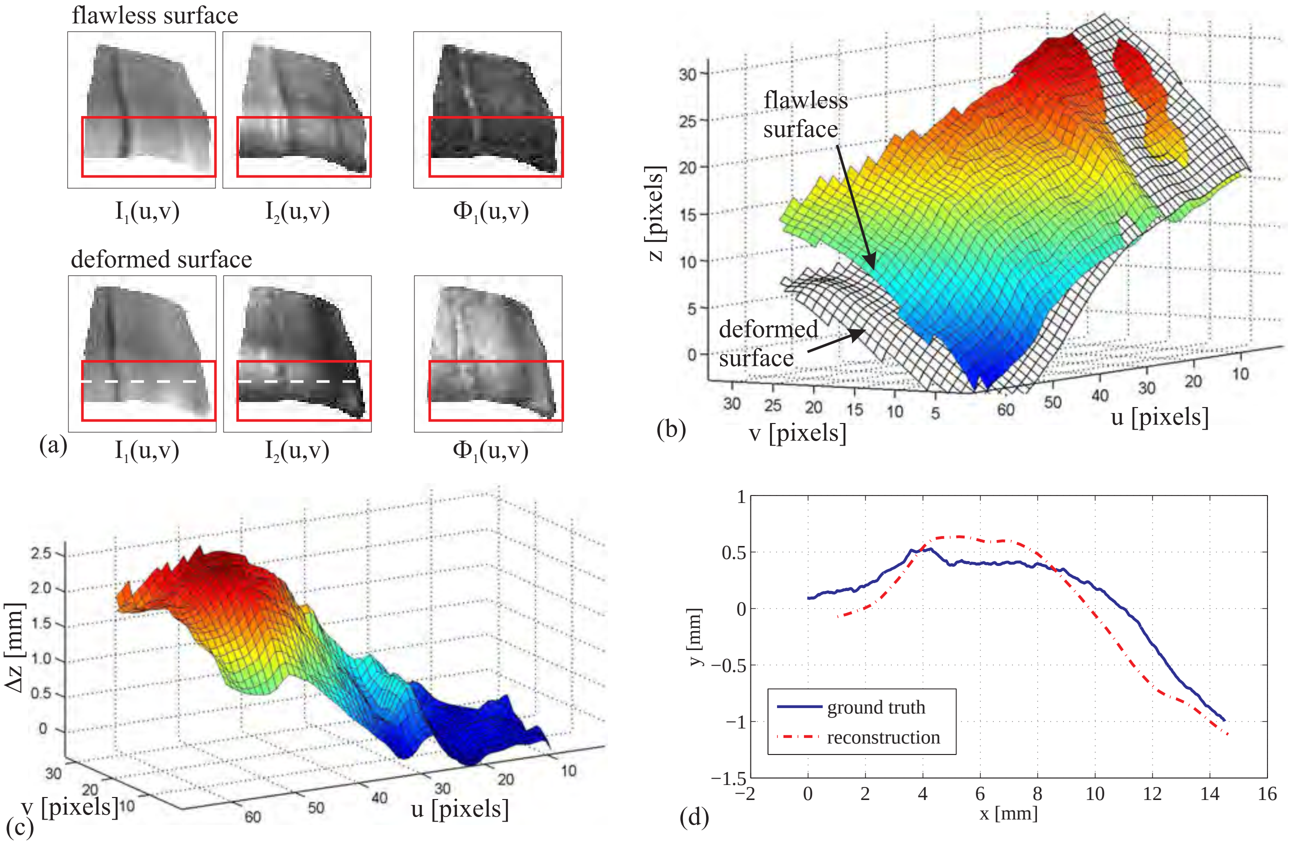


Figure 5: Application of the described 3D surface reconstruction method to a raw forged iron surface. (a) Reflectance and polarisation angle images. The red boxes indicate the reconstructed area. (b) Reconstructed 3D profiles of both parts, viewed from the upper right. (c) Difference Δz between flawless and deformed surface. (d) Comparison of the cross-section indicated by the dashed line in (a) to ground truth.

for detecting anomalies of the surface shape, thus rendering it a promising technique for optical quality inspection systems.

REFERENCES

Batlle, J., Mouaddib, E., Salvi, J., 1998. Recent progress in coded structured light as a technique to solve the correspondence problem: a survey. *Pattern Recognition*, 31(7), pp. 963-982.

D'Angelo, P., Wöhler, C., 2005. 3D Reconstruction of Metallic Surfaces by Photopolarimetric Analysis. In: H. Kalviainen et al. (Eds.), *Proc. 14th Scand. Conf. on Image Analysis*, LNCS 3540, Springer-Verlag Berlin Heidelberg, pp. 689-698.

D'Angelo, P., Wöhler, C., 2005. 3D Surface Reconstruction by Combination of Photopolarimetry and Depth from Defocus. *Pattern Recognition, Proc. of 27th DAGM Symposium*, LNCS 3663, Springer-Verlag Berlin Heidelberg, pp. 176-183.

Horn, B. K. P., Brooks, M. J., 1989. *Shape from Shading*. MIT Press, Cambridge, Massachusetts.

Horn, B. K. P., 1989. Height and Gradient from Shading. MIT technical report 1105A.
http://people.csail.mit.edu/people/bkph/AIM/AIM-1105A-TEX.pdf

Jiang, X., Bunke, H., 1997. *Dreidimensionales Computersehen*. Springer-Verlag, Berlin.

McEwen, A.S., 1985. Topography and albedo of Ius Chasma, Mars. *Proc. 16th Conf. on Lunar and Planetary Science*, pp. 528-529.

Miyazaki, D., Kagesawa, M., Ikeuchi, K., 2004 Transparent Surface Modeling from a Pair of Polarization Images. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 26(1), pp. 73-82.

Miyazaki, D., Tan, R. T., Hara, K., Ikeuchi, K., 2003. Polarization-based Inverse Rendering from a Single View. *IEEE Int. Conf. on Computer Vision*, Nice, France, vol. II, pp. 982-987.

Nayar, S. K., Ikeuchi, K., Kanade, T., 1991. Surface Reflection: Physical and Geometrical Perspectives. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 13(7), pp. 611-634.

Powell, M. J. D., 1970. A Fortran Subroutine for Solving Systems of Nonlinear Algebraic Equations," *Numerical Methods for Nonlinear Algebraic Equations*, P. Rabinowitz, ed., Ch.7.

Rahmann, S., Canterakis, N., 2001. Reconstruction of Specular Surfaces using Polarization Imaging. *Int. Conf. on Computer Vision and Pattern Recogntion*, Kauai, USA, vol. I, pp. 149-155.

Wöhler, C., Hafezi, K., 2005. A general framework for three-dimensional surface reconstruction by self-consistent fusion of shading and shadow features. *Pattern Recognition*, 38(7), pp. 965-983.

Wolff, L. B., 1991. Constraining Object Features Using a Polarization Reflectance Model. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 13(7), pp. 635-657.

COMPARATIVE STUDY OF DISPARITY ESTIMATIONS WITH MULTI-CAMERA CONFIGURATIONS IN RELATION TO DESCRIPTIVE PARAMETERS OF COMPLEX BIOLOGICAL OBJECTS

Michael Nielsen*, Hans J. Andersen, Erik Granum

Aalborg University
Laboratory of Computer Vision and Media Technology
mnielsen@cvmt.dk

KEY WORDS: 3D reconstruction, Multi-Baseline, Trinocular Stereo, Graph Cuts, Performance Evaluation, Biological Structures, Remote Sensing.

ABSTRACT

This paper aims at evaluating multi-camera configurations as a function of the descriptive parameters of complex biological objects. Multi-baseline Stereo has potential to handle projective distortion at large baselines. Being close to the observed object and the orientation of object surfaces pointing toward the camera increase the *projection distortion*. An example is 3D reconstruction of plants where the leaves can be pointing steeply toward the cameras, while, sub-leaf reconstruction needs high depth resolution, because the leaves overlap closely to each other. The paper presents a new dissimilarity measure, called Sums of Individual Sums of Squared Differences (SISSD). It takes projection distortion and changing specular highlights into account by learning the gradual changing of the feature window. The method was included in the comparative study that used realistic ray traced plant models, where the descriptive parameters of the objects could be controlled. Other configurations in the experiment were the commonly used Multi-baseline Sum of the Sums of Squared Differences (SSSD), the popular binocular graph cuts, and two trinocular correlation techniques. Comparison is in regard to leaf type, texture and orientation, proportion of occlusion and proportion of changing highlights by computing the overall-, occluded-, and highlighted- percentage of bad matching pixels ($pbmp$, $pbmp_{occ}$, and $pbmp_{high}$). The results showed a complicated relationship of trade-offs that points toward further development combining the strengths of the individual configurations.

1 INTRODUCTION

Computer vision based 3D reconstruction of close-up complex biological structures is a difficult discipline. There are various multi-camera configurations to choose from. It would be useful to learn about the performance related to descriptive parameters of the objects at hand, in order to choose the best configuration. The Descriptive parameters of the objects are *surface shape*, *surface orientation*, *presence of texture*, *proportion of changing specular highlight* and *proportion of occlusion*. The specular highlights in concern are those that changes gradually from one image to the next across the baseline. Multi-baseline Stereo has been described and tested in literature as a method for improving the handling of occlusion and ambiguity across the scan lines (Okutomi and Kanade, 1993)(Jeon et al., 2001) by using the sum of the energy measures across the camera array; e.g. Sum of Sums of Squared Difference (SSSD). Attempts have also been made at dealing with specular highlights by actively detecting specular highlights within the algorithm (Li et al., 2002) and treating them as occlusions. However, the problems related to nearby objects are overlooked as the algorithms assume that the area looks the same in all cameras. This paper presents an alternative measure that utilizes the fact that a multi baseline array consists of subsets of smaller baselines. A large baseline improves depth resolution but it also makes the correspondence more difficult (Okutomi and Kanade, 1993). Three factors increase this effect: Being close to the observed object, window correlation size, and orientation of object surfaces.

Precision agriculture is a field with rising interest in 3D computer vision, which is becoming tangible as new high dynamic range cameras and precalibrated multi-view cameras are being developed. These cameras satisfy the epipolar geometry constraints and the intrinsic- and extrinsic calibration can be skipped. Close-up 3D reconstruction of plants is an excellent example where the leaves can be pointing steeply toward the cameras and it needs high depth resolution because the leaves overlap closely to each other. Excellent depth maps has potential to aid the segmentation of individual leaves (Lee et al., 1996), if the disparity maps have trustworthy discontinuity edges. This is useful in precision agriculture for segmenting individual leaves for autonomous weed identification, fruit picking, branch thinning, and for finding sampling points on specific locations of a plant (Christensen and Jørgensen, 2003,)(Nielsen et al., 2004). The image acquisition is expected to be done from a moving platform in an outdoor environment, so reconstruction must be done from a single time slice.

In general terms plants belong to the class of objects that are: semitransparent, biological, non-rigid structures. Disparities are often non-planar and can get very *steep* toward the cameras. Textures are non-existent or highly detailed, and having more or less specular highlights. Fortunately, they are segments of smooth surfaces, but intertwining and overlapping. It is very difficult to get dense ground truth. The Vision based depth map reconstruction is usually confined to fronto-planar depth scenes, where the depth maps can be described as regions of near-equal disparities. These scenes are viewed from a distance and have small finite disparity spaces, where it is reasonable to manually acquire ground truth. As an alternative, structured light can be used. It uses multiple images so that the objects must be rigid in time (Scharstein and Szeliski, 2003).

2 METHODS AND MATERIAL

The stereo correspondence algorithms were all based on a basic Sum of Squared Difference (SSD) dissimilarity (energy) function (eq. 1). *The presented methods assumes precalibrated images satisfying epipolar geometry constraints, equal baseline, and zero rotation*.

$$E_{i,j}(x,y,d) = \sum_{(u,v)\in W(x,y)} (I_i(u,v) - I_j(u+d,v))^2 \tag{1}$$

d is the tested disparity, W is the window around (x,y), I_i is the ith image. The windows can be placed in various ways around the pixel and question, but we limited this experiment to centered windows. Adding multiple windows can improve the correspondence near disparity borders (Fusiello et al., 2000), but we wanted to keep this factor out of the experiment this time. It was shown in another experiment that five symmetric windows were optimal, ie. the center and the four diagonals (Nielsen et al., 2005).

In the classical multi baseline SSSD the Sum of Squared Difference between the reference camera and the ith camera is computed for N cameras. See equation 2.

$$S(x,y,d) = \arg\min_{d} \sum_{c=2}^{N} (E_{1,c}(x,y,\frac{d(c-1)}{N-1})) \tag{2}$$

We see that the binocular case ($N = 2$) is a special case of this equation.

2.1 Introducing SISSD

A new measure Sum of Individual Sums of Squared Differences is defined as SISSD (see equation 3). This measure was supposed to learn the graduate change in the feature window across the baseline. This could be a problem with occlusions as it would learn the feature of the occluding object, which was countered by including the weighted dissimilarity in regard to the reference camera. In the new measure we computed the Sum of Squared Difference between the $i-1$th and the ith camera, and between the 1st and the ith camera to ensure that it does not adapt to a completely different object.

$$\begin{aligned} S(x,y,d) &= \arg\min_{d} \sum_{c=2}^{N} [\alpha(E_{c-1,c}(x,y,\frac{d(c-1)}{N-1})) \\ &\quad +(1-\alpha)(E_{1,c}(x,y,\frac{d(c-1)}{N-1}))] \end{aligned} \tag{3}$$

We see that SSSD is a special case of SISSD, where $\alpha = 0.0$. Figure 1 shows an example of the case with steep object where the projection distorts the orientation of the leaf. The top shows parts of images of a five camera array. The middle plot the development of the dissimilarity (energy) across increasing baseline. It is obvious that SSSD increases exponentially, while SISSD is even less than SSD. The bottom plot shows the dissimilarity for the three measures across the scan line and prints the best match for SSD, SSSD, SISSD and Ground Truth (GT). This trait should also be an advantage in the presence of specular highlights that travel across the baseline. An example is shown in figure 2. Based on these preliminary results, a benchmark experiment was performed. The goal was to validate that SISSD performed better than SSSD on steep-leaved objects and in areas where the specular highlight state changes, and whether the reference similarity constraint could counter the occlusion problem.

2.2 Comparative Methods

The other common multi-camera alternative to the multi baseline camera array is called the right-angled trinocular L-setup (Mulligan and Daniilidis, 2002). Two different trinocular algorithms are used for comparison, trinocular minimum (T_m eq. 4) and trinocular sum (T_s eq. 5). In principle, they use two image pairs, where the second switches the disparity to the y-axis. Their baselines are equal to the largest multi baseline (Image N).

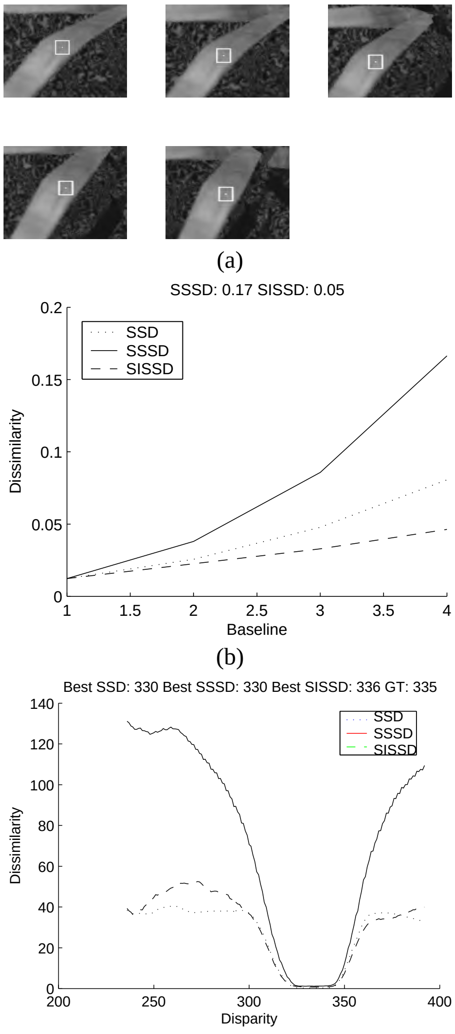

(a)
(b)
(c)

Figure 1: The case of steep leaves where projection changes orientation across the baseline. (a) five views of the location on the steep leaf. (b) The development of the dissimilarity across the baseline. (c) The dissimilarity/energy function across the scan line in the image. The best match for SSD, SSSD, SISSD ($\alpha = 1$), and ground truth (GT) is given over the graph.

$$T_m(x,y,d) = \arg\min_{d} \min(E_{1,N_x}(x,y,d), E_{1,N_y}(y,x,d)) \tag{4}$$

$$T_s(x,y,d) = \arg\min_{d}(E_{1,N_x}(x,y,d) + E_{1,N_y}(y,x,d)) \tag{5}$$

In theory the T_m should comparably be more robust to occlusions by choosing the best match in a single image pair. T_s should comparably be more certain of a match if the point is visible in all cameras by choosing the best match where both image pairs are good matches.

One of the best 3D reconstruction algorithms available uses a graph cut energy minimization, which yields similar results to the slower simulated annealing. The difference is that graph cuts preserves depth discontinuity (Kolmogorov and Zabih, 2002). It does not rely on window sizes which tend to dilate the depth regions and are sensitive to perspective distortion. The main adjustable parameter is the impact of the smoothness constraint, λ.

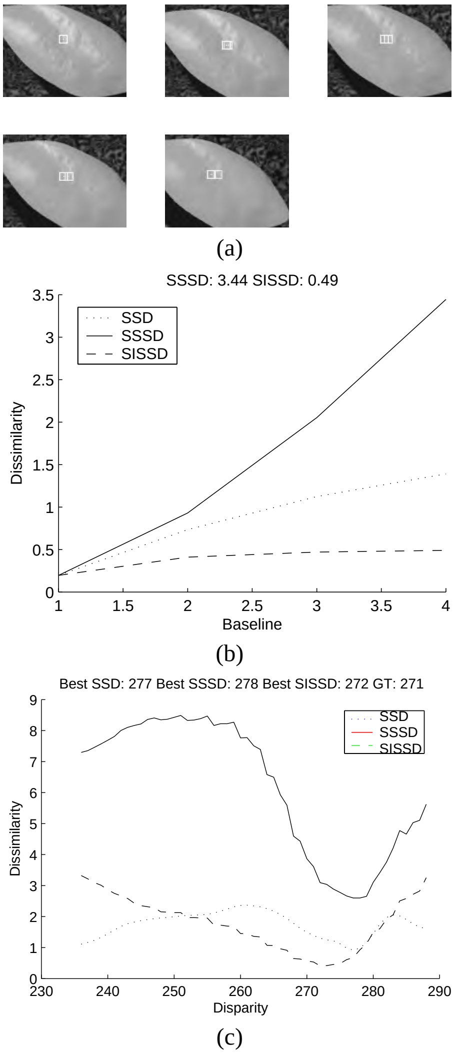


Figure 2: The case of flat leaves where the highlight changing across the baseline. The potential weakness of SISSD is that the dissimilarity difference between the correct match and its surroundings is not very pronounced. This makes the global minimum sensitive to jitter.

Since it assumes regions of equal depth, it excels at fronto-planar scenes, but may have trouble when it comes to steep leaves on plant structures. It was interesting to see how it performed in this new context. We used Kolmogorov's implementation of the graph cut algorithm (Kolmogorov and Zabih, 2002) that is referred to as *kz1*. This is only a binocular algorithm which used the 1st and the Nth camera. λ was given a small value (half of the automatic setting).

There are three common quality metrics root-mean-square, reprojection/prediction of a novel view(Szeliski and Zabih, 1999), and percentage of bad matching pixels. The latter is chosen because the focus is to generate correct disparity maps. Root-mean-square error does not ensure that the structure and discontinuities are preserved. Reprojection error does not measure the actual disparity error, but *whether the reprojection of one green pixel happen to hit a matching green pixel* in the novel view. However, in a scene full of green plants that is very likely even if the disparity is very wrong.

The estimated disparity maps d_E were compared to ground truth (d_{GT}) using the Percentage of Bad Matching Pixels metrics as in (Scharstein and Szeliski, 2002):

$$PBMP = \frac{1}{N}\sum_{(x,y)} |d_E(x,y) - d_{GT}(x,y)| > \delta \qquad (6)$$

2.3 Experimental Setup

The experimental tests were conducted in order to learn more about the algorithms in the complex context of close-up reconstruction of complex structures. Hence, near-photo realistic ray traced scenes of plants were used in order to control the scene parameters and get valid ground truth disparity maps, occlusion masks, and highlight masks. The scenes had natural outdoor lighting and focal blur, which is a natural problem with plants with steep leaves. Blur is unavoidable, because the aperture cannot be very small and the shutter must be fast when capturing images from a moving platform and the plants are waving in the wind.

Two main classes of plants, long leaf (grass-like, e.g. cereal) and broad leaf (e.g. beet and tomato) were generated. This relates to *surface shape*. For each of these there were plants with steep leaves and flat leaves, respectively. This relates to *surface orientation*. Steep leaves compared to flat leaves have less highlight, more occlusion, and vice versa. A natural case with two grassy plants with flat and steep leaves and a lot of occlusion were used, too. Each scene was generated with textured (spotted) and no texture (glossy), both having bump maps. This relates to *presence of texture*. Finally, all images very generated with and without specularity. This served two purposes; 1. it was required to find the highlight masks (where highlights exist in one frame and not the other), and 2. in order to test overall performance of the algorithms and the same geometrical structure with and without the presence of highlights. There were 18 image sets in total. See figure 3 for an example with ground truth.

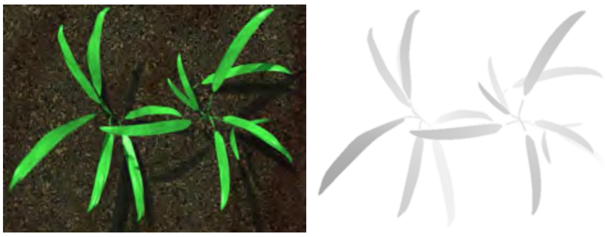


Figure 3: A natural case, where two grass-like plants are close together and leaves are occluded. The proportion of occluded pixels is 5% and the proportion of changing highlights are 5%.

3 RESULTS AND DISCUSSION

The overall results are shown in table 1. It is the mean and spread of performance over all plant types. Note that the ground truth maps were calculated in floating points as to represent the (scaled) inverse of the real height. The disparity maps were integer pixels. If the ground truth had been rounded, the values would have been 10-20% lower. $Multi_{3cam}$ used the same cameras as $Multi_{5cam}$, but skipped camera 2 and 4.

The table shows that having those two extra cameras in between the three cameras did improve the result by 11% in average for all pixels, 8% for highlighted pixels, and 8% for occluded pixels. Meanwhile, their spread was approximately equal or slightly narrower (for occluded pixels). The significance of 8.9% versus 8.2% is up to the application to decide. The development within

Table 1: Comparison of Stereo setups. Mean PBMP (%) and their standard deviations calculated from all pixels (all), pixels with different specularity state (high), and occluded pixels (occ).

Stereo Setup	All	High	Occ
$Multi_3SSSD$	8.9(5.9)	22.1(14.6)	50.3(30.9)
$Multi_3\alpha0.25$	8.9(5.6)	20.9(13.7)	55.4(28.7)
$Multi_3\alpha0.50$	9.9(5.6)	20.6(12.1)	64.6(23.8)
$Multi_3\alpha0.75$	13.5(6.6)	23.0(12.2)	69.1(24.0)
$Multi_5SSSD$	8.3(5.5)	20.3(13.9)	46.1(28.3)
$Multi_5\alpha0.25$	8.2(5.3)	19.4(13.3)	49.9(24.5)
$Multi_5\alpha0.50$	8.8(5.4)	19.1(12.7)	55.3(22.5)
$Multi_5\alpha0.75$	11.6(6.0)	21.0(12.5)	69.1(20.4)
$GraphCut$	14.6(8.7)	19.6(16.3)	73.9(24.3)
$TrinoMin$	10.2(6.5)	23.1(12.5)	30.6(22.3)
$TrinoSum$	9.8(6.9)	23.6(15.8)	40.3(25.0)

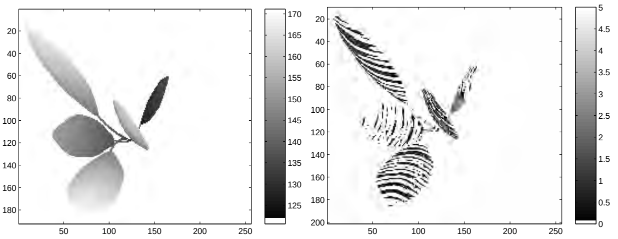


Figure 4: [Left] Ground truth and [Right] Graph Cuts Log(disparity error) for steep spotted broad leaf without highlights. The banding characteristics were caused by the attempt to impose fronto-planar regions on the steep leaves.

$multi_5$ by increasing $alpha$ was devastating for occluded pixels by 50%, while overall and highlight pixels reach a local minima between $\alpha = 0.25$ and $\alpha = 0.5$. The benefit was rather small, though; 1% for all pixels and 5% for highlight pixels. The SISSD measure may be a improvement when using larger window sizes, which tend to be the case when using real images. The trinocular measures did well and they excel at occluded pixels, especially T_m. Graph cuts did the worst, except at correcting highlight pixels by smoothing those areas. Figure 4 shows why graph cuts did not do very well. The disparity map was banded, ie. staircase shaped, instead of smooth.

Figure 5 shows the errors from the multi-baseline reconstruction of the same plant. The errors were more recognizable as noisy jitter, which could be removed by an energy minimizing sloped smooth surface technique.

Figure 6 shows the errors from trinocular results for the same plant. The very steep leaf in the middle and the one to the right of it are difficult for all the algorithms except trinocular minimum (T_m). It is so steep that it is almost a self-occlusion. In the second camera the leaf would be extended along orientation of the baseline, thus occluding the other leaf. T_m simply reconstructed it from the Y direction. The lesson is that it is not only the orientation toward the camera that affects the result, but if the orientation of a leaf aligns with the baseline it can be difficult to reconstruct it. This is especially a problem with textureless grass-like leaves that aligns with the baseline (Nielsen et al., 2004). In comparison, SISSD was able to reconstruct the steep leaf nearly as good, but the leaf to the right of it was as bad as Trinocular sum (T_s).

Figure 7 plots the all-pixel results grouped by descriptive object parameters, i.e. leaf shape, leaf orientation (flat or steep leaves), texture, and highlights and occlusion. Horizontal axis is the setup: M 0.0 (SSSD), M 0.25 (SISSD $\alpha = 0.25$), M 0.5, M 0.75, Binocular Graph Cut, Trinocular Minimum T_m, and Trinocular sum T_s. The vertical axis is the mean pbmp for window sizes ranging from 4-12. The same goes for figures 8 and 9 that show the pbmp of highlight pixels and occlusion pixels, respectively.

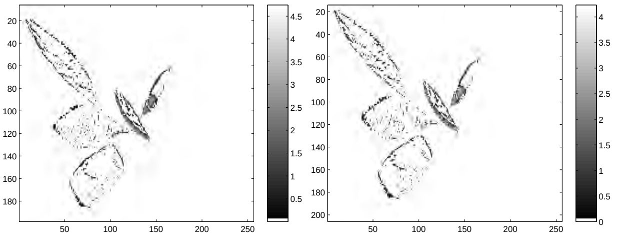


Figure 5: [Left] Log(disparity error) Multi-baseline SSSD and [Right] SISSD $\alpha = 0.5$. These results did not have any banding, but the difference between the SSSD and SISSD was very small. The result would be excellent if it were combined with a slope- and discontinuity preserving graph cut minimization.

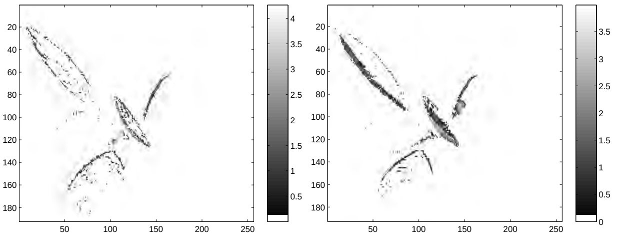


Figure 6: [Left] Log(disparity error) trinocular minimum (T_m) and [Right] trinocular sum (T_s).

Figure 7 plot (a)(plants without specular highlights) clearly pins down the sources of error for reconstruction in general. The flat-leafed plants (since they had no specular highlights on this plot) all score very well. The errors were large when the leaves were steep or occluding (the model called *two grassy* is 5% occluded in comparison to the steep broad leaf which is only 1%).

The interesting aspect on plot (a) on figure 7 is that it was the steep leaves that best improved slightly from SISSD, while the flat leaves are reconstructed best through SSSD. However, taking a look at plot (b) reveals that when there were highlight on those flat leaves, SISSD was an improvement, too, especially for broad leaf plants.

Note also the fact that the steep leaves were troublesome for graph cuts on plot (a) and (c), especially the glossy steep broad leaf, which was easier for the others compared to grassy plants. Plot (a) to (d) shows consistently that T_s reconstructed grass-like plants better than T_m, but T_m reconstructed broad leaf plant best. This trend is revisited in figure 8.

Figure 7 Plot (d) shows that in the more natural case, SSSD and T_s were best, even though T_m was best in most occluded parts (figure 9 plot (a) and (b)). Maybe the algorithm could dynamically choose T_m by detecting occlusion with left-right consistency (Fusiello et al., 2000).

Figure 8 plot (a) and (b) shows the subtle strength of SISSD in the highlighted areas. The flat glossy broad leaf was the most difficult to reconstruct. Note that this is the plant type that was 50% highlighted, and there were no texture other than shading and bumps to correlate. The graph cut algorithm were particularly bad in this case, because it created non existant surfaces in over the plant from the errors of the highlights.

4 CONCLUSIONS

The relationship between the performances of the algorithms and the descriptive parameters of the plant objects were investigated. A new multi-baseline Sum of Squared Difference based correlation was defined (SISSD) in order to minimize the effect of perspective distortion within the windows. The results showed that there was a relationship between the performance and the descriptive parameters of the objects. However, SISSD was only a marginal improvement on images with steep leaves (slopes), but more so in the presence of highlights. It was mainly an improvement at the actual highlight areas, especially on shiny broad leaf plants. On the other hand SSSD was better at matching the occluded areas. The best algorithm for occluded areas was the trinocular T_m algorithm. Binocular Graph cuts were not able to reconstruct the slopes in steep leaves, but the smoothness optimization seemed to smoothen over the errors from highlights, when the highlight areas were not too large. The results showed a complicated relationship of trade-offs that points toward further development combining the strengths of the individual configurations.

4.1 Perspectives on future work

An improvement to the SISSD measure could be to have α depend on the distance from reference image. Another interesting aspect would be to place the 5 cameras in a trinocular setup. The five cameras would then complete two systems of three-camera multi-baseline systems in each direction.

Furthermore, a multi-baseline or trinocular algorithm in combination with graph cuts would be interesting to pursue, and to improve its ability to reconstruct steep slopes. There are other works on these aspects to pay special attention to (Buehler et al., 2002)(Lin and Tomasi, 2004). Buehler's trinocular algorithm does not handle the situation where occlusion only exist in one camera pair. This was the strength of the trinocular minimum algorithm in this paper. Lin and Tomasi's algorithm for sloped surfaces relies too strongly on large smooth surfaces. This may be a problem for natural leaves that can be curled and there might only be small segments showing of each leaf, while the surface boundaries are only vaguely defined by intensity edges (sometimes not at all).

The final step is to create a mesh that is able to treat intertwining and overlapping leaves as individual surfaces.

4.2 Acknowledgments

The authors would like to thank the Danish Technical Research Council, the Danish Agricultural and Veterinary Research Council and the Danish Ministry of Food, Agriculture and Fisheries for funding.

REFERENCES

Buehler, C., Gortler, S. J., Cohen, M. F. and McMillan, L., 2002. Minimal surfaces for stereo. In: ECCV (3), pp. 885–899.

Christensen, L. K. and Jørgensen, R. N., 2003. Spatial reflectance at sub-leaf scale discriminating NPK stress characteristics in barley using multiway regression (N-PLS). In: 2003 ASAE Annual International Meeting, Las Vegas, Nevada, USA, July 27-30, Paper no. 031138.

Fusiello, A., Roberto, V. and Verri, A., 2000. Symmetric stereo with multiple windowing. International Journal of Pattern recognition and Artifical Intellingence 14(8), pp. 1053–1066.

Jeon, J., Kim, K., Kim, C. and Ho, Y., 2001. Robust stereo matching algorithm using multiple-baseline cameras. In: IEEE Pacific Rim Conference on Communications, Computers and signal Processing, Vol. 1, pp. 263–266.

Kolmogorov, V. and Zabih, R., 2002. Multi-camera scene reconstruction via graph cuts. In: IEEE European Conference on Computer Vision.

Lee, W. S., Slaughter, D. C. and Giles., D. K., 1996. Development of a machine vision system for weed control using precision chemical application. In: International Conference on Agricultural Machinery Engineering '96, Seoul, Korea., pp. 802–811.

Li, Y., Lin, S., Lu, H., Kang, S. and Shum, H.-Y., 2002. Multibaseline stereo in the presence of specular reflections. In: IEEE Intl Conf. on Pattern Recognition, Vol. 3, pp. 573–576.

Lin, M. and Tomasi, C., 2004. Surfaces with occlusions from layered stereo. In: IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI), Vol. 26(8), pp. 1073–1078.

Mulligan, J. and Daniilidis, K., 2002. Trinocular stereo: A realtime algorithm and its evaluation. International Journal for Computer Vision 47, pp. 51–61.

Nielsen, M., Andersen, H. J., Slaughter, D. C. and Granum, E., 2005. Ground truth evaluation of 3d computer vision on nonrigid biological structures. In: J. Stafford (ed.), Precision Agriculture 05, Wageningen Academic Publishers, The Netherlands, pp. 549–556.

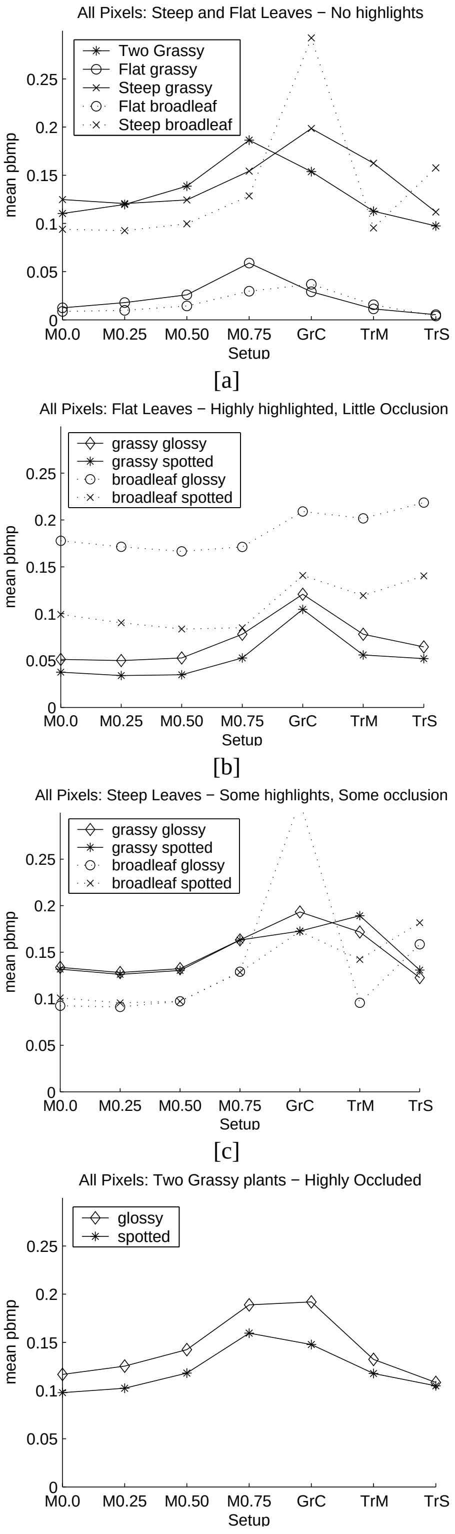
Nielsen, M., Christensen, L. K. and Andersen, H. J., 2004. Subleaf scale remote sensor for npk discrimination using stereo vision. In: Engineering the Future, International Conference on Agricultural Engineering, Leuven, Belgium, 12-14 September, Session 10, no. 327.

Okutomi, M. and Kanade, T., 1993. A multi-baseline stereo. In: IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 15(4), p. 353363.

Scharstein, D. and Szeliski, R., 2002. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. International Journal of Computer Vision 47(1/2/3), pp. 7–42.

Scharstein, D. and Szeliski, R., 2003. High-accuracy stereo depth maps using structured light. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2003), Madison, WI, USA, pp. 195–202.

Szeliski, R. and Zabih, R., 1999. An experimental comparison of stereo algorithms. In: Springer, International Workshop on Vision Algorithms, Corfu, Greece, pp. 1–19.



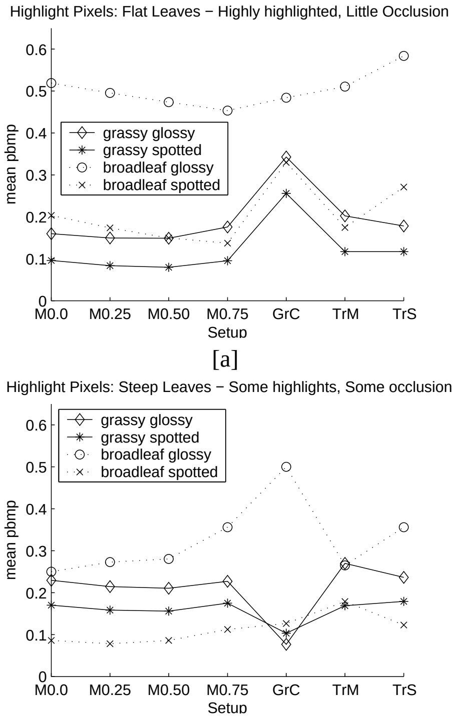
[a]

[b]

[c]

[d]

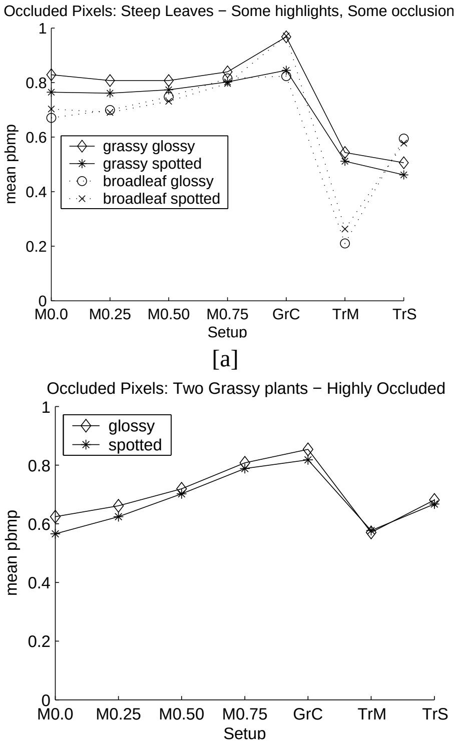
Figure 7: PBMP from all pixels results by object type and leaf orientation. The worst case occlusion is the *Two Grassy Plants* model being 5% occluded. The worst case of highlights were the flat grass-like and flat broad-leaf. 20% of their area suffered from changing specular highlights.



[a]

[b]

Figure 8: PBMP from specular changing highlight pixels results by object type and leaf orientation. SISSD (M0.25-M0.75) improves performance.



[a]

[b]

Figure 9: PBMP from occluded pixels results by object type. Trinocular minimum T_m is the best algorithm for occluded areas.

AUTHOR INDEX