The University of Western Australia

Technical Report 95/6

July 1995

Surface Reconstruction based on Visual Information

Reinhard Klette, Andreas Koschan,
Karsten Schlüns, and Volker Rodehorst

Computer Vision Group, FR 3-11
Computer Science Department, Berlin Technical University, D-10587 Berlin, Germany
http://www.cs.tu-berlin.de/~cvworld
{klette,koschan,karsten,vr}@cs.tu-berlin.de

Abstract

The computation of surface data based on visual information is an important sub-component in the computer-graphical surface reconstruction of solids and in the control of 3-D environments. Different methodologies can be used for that as, e.g., static stereo, shape from motion, shape from shading, photometric stereo, or structured lightening. There exist different basic approaches in literature often based on simplifying assumptions. However, it is well known that such assumptions may not be true if surface reconstruction is applied under practical circumstances. In this paper, several problems are mentioned which are related to practical applications of surface reconstruction approaches following the methodologies of static stereo, shape from motion, and photometric stereo. We present specific solutions to cope with these problems, or the solution state what was reachable in our work. Some problems are ill-posed and limitations of approaches have to be accepted. As a second contribution of this paper, we discuss the evaluation problem of surface reconstruction algorithms. It is important to answer such questions as 1) For what kind of surfaces and 3-D objects an algorithm behaves either well or bad? 2) How accurate are the reconstruction results of an algorithm under specified circumstances? What measure can be used to evaluate reconstruction accuracy? 3) How to compare reconstruction results following different methodologies? 4) What algorithm can be suggested for a specific application project? and so on. So far we present some proposals and first quantitative or qualitative results for answering such questions. In our opinion a methodology for evaluating surface reconstruction algorithms is still at its beginning. However a critical evaluation of potential methods in project applications is helpful in selecting the appropriate algorithm.

Keywords

Static stereo, shape from motion, photometric stereo, performance evaluation.

CR categories

I.4, I.2, F.2

Communicated by

Dr Chi-Ping Tsang

Contents

1 INTRODUCTION

This report is about work that was stimulated by applications of several surface reconstruction algorithms. Two typical application situations are assumed. First, *computer-graphical surface reconstructions of solids*, e.g., in building architecture, medical surgery, production and manufacturing etc., need precise 3-D surface data. Computing time demands are often relaxed in this application. Second, the *control of 3-D environments*, e.g., in robotics or active vision, also incorporates tasks of (rough) obstacle detection, distance estimations or motion control. Here, a few range data or rough drafts of object surfaces can be sufficient. However, the computing time has to be minimized to ensure on-line processing.

The report presents some refinements of elementary approaches in static stereo, shape from motion and photometric stereo. A comparative performance analysis is initiated for evaluating different surface reconstruction results.

1.1 Aims and Problems

The results of surface reconstruction algorithms are either dense according to the spatial resolution of the image (dense range data, dense motion field, dense gradient data etc. - at all pixels of an image, or in a few image segments), or they are isolated measurements, i.e. sparsely distributed. *Reconstructed surface patches* are approximations of 3-D object faces containing absolute or relative depth information. They can be accurate or very rough - this is a matter of reconstruction quality. Depth or surface representation methods are used for these patches as known in computer graphics, e.g. a 3-D point cluster representation for depth values, or the floating horizon method for surface representation. *Isolated measurements* of 3-D point positions are approximate locations of specific surface points. Reconstructed surface patches and isolated measurements characterize the two typical aims of surface reconstruction algorithms. Specific applications are characterized by these aims and accuracy demands, cp. Tab. 1.1.

application	aim	computational speed
computer-graphical surface reconstructions of solids	reconstructed surface patches, isolated measurements, high accuracy demands	typical: off-line processing also possible: on-line processing
control of 3-D environments	reconstructed surface patches, isolated measurements, lower accuracy demands	typical: on-line processing also possible: off-line processing

Table 1.1: Aims in different surface reconstruction fields.

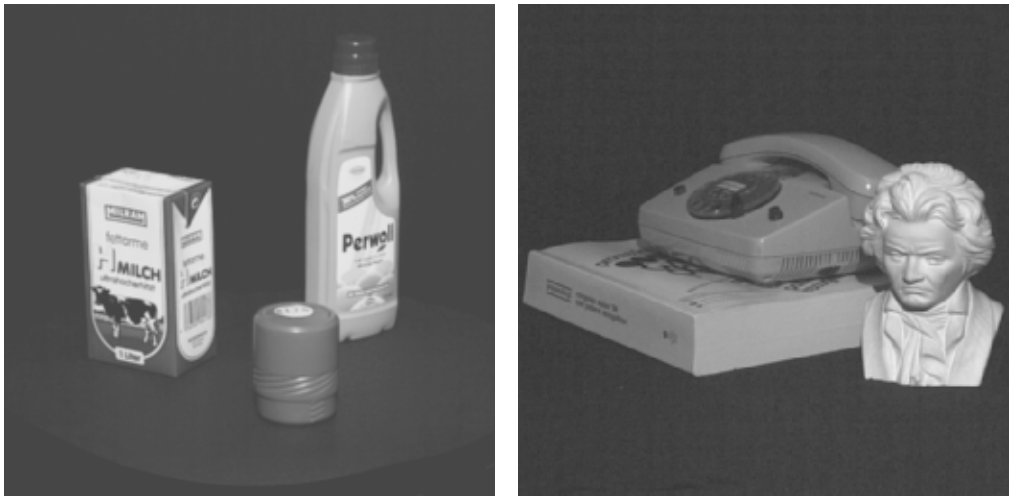


Figure 1.1: Typical input pictures of the scene spaces PACKAGE (box, bottle etc.) and BEETHOVEN (plaster statue, alone or in a desk environment).

Two scene spaces PACKAGE and BEETHOVEN are used throughout the paper for comparing results of different algorithms. Typical input images of these scene spaces are shown in Fig. 1.1. For image acquisition we have used a 3CCD-Color-Camera DXC730P from Sony. Gain control and gamma correction were turned off. Automatic white balancing was used. The digitization process was done by a DATACELL s2200-frame grabber in a Sun IPX workstation. The focal length used with static stereo (Section 3) and dynamic stereo (Section 4) was 23 mm. The focal length used with the shading based approaches (Section 5) was ca. 60 mm. The scenes were illuminated by one or several slide projectors (dynamic stereo and shading based approaches) or fluorescent ceiling lamps (static stereo).

The objects in these experiments are considered under the assumptions of computer-graphical surface representations and of control of 3-D environments. Three approaches were considered in these experiments

- *static stereo analysis* for color cameras (with studying parallel realizations of correspondence calculations to ensure on-line processing),
- *motion analysis* with extensive testing and modifications of optical flow computations, and a shape reconstruction algorithm for a rotating disc based on Tsai's camera calibration (*dynamic stereo analysis*),
- *photometric stereo analysis* for polyhedral objects and for different reflection assumptions, where different albedo values can be present in a scene.

We also started with experiments following a forth approach,

- *light plane projections* and a rotating disc, based on Tsai's camera calibration.

The experimental setting is still under development so that this forth approach is not yet included in the discussion in this report. However some comments in the Conclusions also reflect experimental results with this structured lightening method.

In literature the mentioned approaches are normally based on simplifying assumptions [12]. However, it is well known that such assumptions may not be true if surface reconstruction is applied under practical circumstances. For example, the following problems can arise,

- *evaluation of camera calibration method*: For a selected calibration method, the mapping of 3-D scene points onto picture points, and of picture points into the scene should be possible in both directions.
- *selection of correspondence method*: What method should be chosen from all the several hundreds of published methods of calculating corresponding pixels in static or dynamic stereo images?
- *on-line static stereo* : What speed-up is possible if parallel implementations are used instead of serial one's for the chosen correspondence methods?
- *selection of optical flow method*: How to select an appropriate optical flow method?
- *existence of optical flow method*: Is there any optical flow method which satisfies the demands of shape from motion?
- *depth from point correspondences*: How camera calibration results can be used in different situations, e.g. objects on a rotating disc?
- *extension to non-static scenes*: Is reconstruction still possible if rigid scene objects can move with a certain maximum speed? How about non-rigid objects?
- *using more general reflection models*: What reflectance methods can be used if Lambertian reflection can not be assumed? How can an image representing non-Lambertian surfaces be processed to meet the requirements of methods assuming Lambertian reflection?
- *treatment of shadow regions*: How difficult is the treatment of shadow regions if a single point-light source is assumed?
- *consideration of interreflections*: How interreflection models can be used to improve reconstruction methods?
- *inaccurate illumination parameter estimation*: How to deal with inaccuracy in illumination calibration?
- *reconstruction of polyhedral objects*: What methods can be used to reconstruct planar faces?

We mention that certain problems can be ill-posed. In such cases it has to be accepted that no solution exists if no further information is available for transforming the ill-posed problem into a well-posed one.

1.2 Performance Characterization

A general methodology of quantitative algorithm evaluation is described in [6], and used for the evaluation of thinning algorithms. Quantitative evaluations of algorithms are still a rare case in the computer vision literature. As an other example, such quantitative studies were performed in [1, 16, 17] for differential optical flow algorithms. In this report, the scene space PACKAGE is used to describe a first step into the direction of quantitative evaluations of surface reconstruction methods. So far typical comparisons between different methods are based on qualitative evaluations, and this is demonstrated for different reconstruction algorithms in the second scene space BEETHOVEN.

A large number of papers proposing surface reconstruction algorithms have been published in literature. Some performance characterizations should be available to evaluate such an algorithm and to support comparisons between different surface reconstruction algorithms. Such an *evaluation scheme* combines several entries, cp. Tab. 2 for an example. In this report several surface reconstruction algorithms are discussed. However a complete filling of the proposed evaluation scheme is very difficult. We discuss possible entries of Tab. 1.2:

Algorithm XYZ

reconstructed surface patches	**accuracy**	**isolated measurements**	**accuracy**	**computing time**
(A) motion data, gradient data, range data etc.	**(B)** conditions, error measure etc.	**(C)** point location stability	**(D)** conditions, error measure etc.	**(E)** a) serial b) parallel

Table 1.2: Tabular representation of an evaluation scheme of surface reconstruction algorithms.

(A+C) Surface reconstruction algorithms deliver depth values (range data), correspondence data (motion vectors, static stereo correspondences) or surface gradients. In the latter two cases absolute or relative depth information has to be computed from the correspondences (*depth from correspondence*) or from the gradients (*depth from gradients*). Normally, isolated gradient measurements are without practical value, but isolated motion data or range data can be. Under **(A)** it can be specified what data are generated by the algorithm to support the reconstruction of surface patches, and how these data are transformed into depth or surface representations. A criterion for evaluating the computed reconstruction data should also be included here. For example, motion vector computation methods typically lead to dense motion fields, but only those vectors should be selected which are good representations of the viewed 3-D motion. Analogously, under

(C) the kind of computed data is specified, and a method for choosing pixel positions of isolated measurements. Some analysis results about the stability of these point locations would be desirable. For example, a static stereo algorithm delivers several point correspondences, but only some of them allow accurate depth measurements.

(B+D) Accuracy is a relation between *ground truth* and calculated result. For surface reconstruction algorithms, the geometrical locations of 3-D scene object surfaces act as ground truth, and the reconstructed surface patches or isolated measurements are the calculated result. If synthetic objects are used, then the ground truth is known. Real objects are used in the scene spaces PACKAGE and BEETHOVEN. For some objects, as the box in scenes of PACKAGE, the size is available, see Fig. 1.2. In general, in **(B+D)** the population of pictured 3-D objects should be characterized, the image acquisition conditions, the *error criterion* for comparing calculated results with the ground truth and finally the evaluation results based on this error criterion. A quantitative evaluation allows the use of *error criterion functions*. For example, the average of all deviations in reconstructed sizes of the box in PACKAGE scenes, cp. Fig. 1.2 for the actual sizes, is such a function. If such functions are not available then the qualitative visual comparison of reconstructed surfaces with the given 3-D objects is used as a qualitative error criterion.

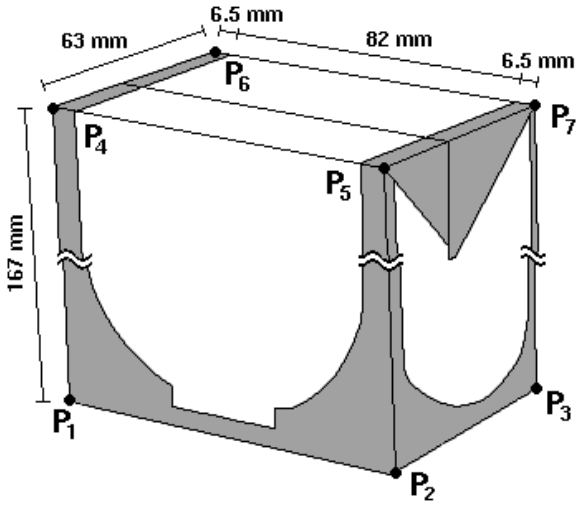


Figure 1.2: Size of the box in PACKAGE scenes as shown in Fig. 1.1.

(E) Additionally to the reconstruction results also the computing time behavior is of interest in relation to the implementation environment, e.g. whether only serial computations or also a parallel processor was used.

A difficult problem arises if the geometrical object model, i.e. the ground truth is not available as it is the case for the plaster statue. However, even if the geometrical surface data are available then still the following problem remains,

- *3-D surface error measure*: How to measure geometrical differences between two 3-D surfaces representing partial views of the same 3-D object?

The complexity of this problem increases if only relative depth values can be reconstructed, i.e. the reconstructed surface patches are scaled with some variable.

The scene space PACKAGE allows evaluations in the sense of categories **(C+D)** in Tab. 1.2, and the scene space BEETHOVEN in the sense of **(A+B)**.

1.3 Contents of the Report

This report describes solutions and performance evaluations for three different stereo analysis approaches. Chapter 3 is devoted to static stereo analysis, Chapter 4
discusses topics in dynamic stereo analysis, and Chapter 5 informs about progress in photometric stereo analysis. Chapter 2 specifies the used camera model and reports about camera calibration results.

For the evaluation of stereo analysis results several experiments are sketched. An *experiment* is described by specifying its input data and the available ground truth, by defining an error criterion, by specifying the selected algorithms which are used, and by a discussion of the experimental results and the conclusions. Normally, the experiments are characterized by input data, ground truth, and error criterion. The further topics are discussed in the context of these experimental settings.

This report can be read as an information about the current state in stereo analysis what could be achieved in our group at the Berlin Technical University. It is also an invitation to other groups for defining joint projects in performance evaluation. The obtained performance evaluations are still far away from the evaluating data sheets which are given in [6] for thinning algorithms. However, we propose quantitative comparisons using objects as the box in the PACKAGE scene space, and qualitative comparisons using plaster statues as the one of Beethoven for further evaluations.

Chapter 3 also contains some material about the parallel implementation of static stereo analysis approaches. This is given in this report with respect to the general importance of parallel implementations for shape reconstruction algorithms. All the methods discussed in Sections 3, 4 and 5 are inherently suitable for parallel implementation because of they are focusing on local computations.

2 PROJECTION MODEL AND CALIBRATION

Camera calibration is of basic importance for reconstruction approaches [18, 30, 47]. This Section describes a solution of the *evaluation of camera calibration method* problem, see Subsection 1.1. First, the camera geometry is briefly introduced. Then the calibration method is cited, and its evaluation is explained.

2.1 Coordinate Systems

The following (left handed) coordinate systems are used to model the relations between scene space objects and projected images, see Fig. 2.1:

(X_w, Y_w, Z_w) denote the 3-D coordinates of object surface points **P** in the *world coordinate system* ,

(X_c, Y_c, Z_c) denote the 3-D coordinates of **P** in the *camera coordinate system* ,

f is the distance between the image plane and the projection center *(focal length)*,

(x_u, y_u) are *non-distorted* image coordinates of (X_c, Y_c, Z_c) assuming an ideal pinhole camera,

(x_d, y_d) are *distorted* image coordinates, differing from (x_u, y_u) by radial lens distortion, and

(x_f, y_f) are *device-dependent* coordinates of (x_d, y_d) in the digitized image (not illustrated in Fig. 2.1).

The Z-axis Z_c of the camera coordinate system coincides with the optical axis, and it is pointing into the scene space.

If only relations between camera coordinates and non-distorted image coordinates are discussed then world and camera coordinates are assumed to be identical. For simplification, (x, y) and (X, Y, Z) are used in this case, without indices u , w or c.

All coordinates and parameters are measured at the same scale, e.g. μm. The only exception are discrete coordinates (x_f, y_f) for the digitized image which are given in (sub-) pixels.

The 3-D camera coordinates (X_c, Y_c, Z_c) are transformed in ideal, non-distorted image coordinates (x_u, y_u) by perspective projection. According to the pinhole camera model it holds

$$x_u = \frac{f \cdot X_c}{Z_c} \quad \text{and} \quad y_u = \frac{f \cdot Y_c}{Z_c}$$

in the *camera centered coordinate system* as shown in Fig. 2.1 where the focal point of the camera, i.e. the projection center, coincides with the origin of the camera-centered coordinate system. If parallel projection is assumed then it holds

$$x_u = X_c \quad \text{and} \quad y_u = Y_c \ .$$

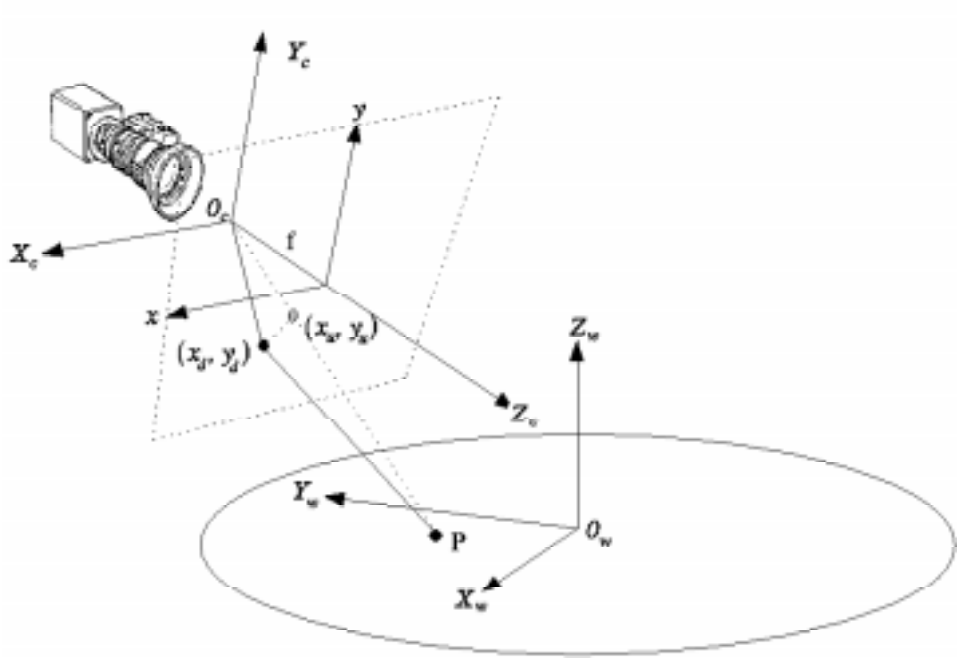


Figure 1.1: Camera geometry with perspective projection: world coordinates $X_w Y_w Z_w$, camera coordinates $X_c Y_c Z_c$, ideally projected coordinates $x_u y_u$ and distorted image coordinates $x_d y_d$ where radial lens distortion is modeled. The circular area illustrates a rotating disc as used for controlled motion in Section 4.

Central projection is assumed in Sections 3 and 4, and parallel projection in Section 5.

2.2 Camera Calibration

For camera calibration, internal camera parameters as well as geometric relations between camera coordinates and world coordinates have to be pre-calculated, and these data have essential influence on the accuracy of surface reconstruction results. The method of R.Y. Tsai [46] is used. For solving the *evaluation of camera calibration method* problem, an extension of this method was necessary.

Four steps are considered in this calibration procedure if an object surface point $\mathbf{P} = (X_w, Y_w, Z_w)$ is mapped on device dependent coordinates (x_f, y_f):

(1) For the affine transform from world coordinates (X_w, Y_w, Z_w) into camera coordinates (X_c, Y_c, Z_c), a rotation matrix **R** and a translation vector **T** have to be calibrated.

(2) For transforming the 3-D camera coordinates (X_c, Y_c, Z_c) in ideal, non-distorted image coordinates (x_u, y_u) by perspective projection, the focal length f has to be calibrated.

(3) For the calculation of non-distorted image coordinates (x_u, y_u) from real, distorted image coordinates (x_d, y_d) the calibration method as proposed by [46] had to be extended. The equations

$$x_d + D_x = x_u \,, \qquad y_d + D_y = y_u$$

are based on the following abbreviations (for radial distortion):

$$D_x = x_d \cdot \left(\kappa_1 r^2 + \kappa_2 r^4\right) \text{ and } D_y = y_d \cdot \left(\kappa_1 r^2 + \kappa_2 r^4\right), \text{ with } r = \sqrt{x_d^2 + y_d^2}\,.$$

The *distortion coefficients* κ_1 and κ_2 have to be calibrated.

These equations can be used for the restoration of images if the values of (x_d, y_d) are known. However if the calibration result should be evaluated then also the transformation of non-distorted into distorted coordinates is of interest. This direction is not described in [46]. The application of numeric methods for solving the resultant non-linear equation system leads to non-acceptable time-inefficiency, e.g. if a full image has to be transformed. Therefore ideal image points (x_u, y_u) can be distorted using the following computational fast approximation scheme:

$$x_{d_i} = \frac{x_u}{1 + \kappa_1 r_{i-1}^2 + \kappa_2 r_{i-1}^4} \text{ and } y_{d_i} = \frac{y_u}{1 + \kappa_1 r_{i-1}^2 + \kappa_2 r_{i-1}^4}, \text{ with } r_i = \sqrt{x_{d_i}^2 + y_{d_i}^2}$$

for $i = 1, \ldots, n$. The initial value is $r_0 = \sqrt{x_u^2 + y_u^2}$. Improved radii r_i are calculated during subsequent iterations.[1]

(4) For transforming non-distorted image coordinates (x_u, y_u) into device dependent image coordinates (x_f, y_f), several parameters have to be calibrated as described in [46].

Using this extension of Tsai's method, the accuracy of calibration results was evaluated. Different real and synthetic calibration objects (planes with calibration points in the latter case) were used for selecting a simple, but sufficient calibration object, cp. [19]. Using synthetic objects it was possible to prove statistically that near-optimum accuracy is achievable already by three calibration planes, each with about 20 calibration points (e.g. an "open cube"). With real calibration objects, the following experiment was performed for measuring the accuracy.

Input data, ground truth and error criterion: Different real calibration objects are used in this experiment. Grids of calibration points are used on these 3-D objects. They are projected into the image plane. The known 3-D positions of the calibration points and the measured positions of the projected points are used as input of the calibration procedure. Then the calibration results are used to transform the 3-D calibration points into test points (labeled with crosses in Fig. 2.1) of the image plane.

---

[1] About eight iterations were sufficient.

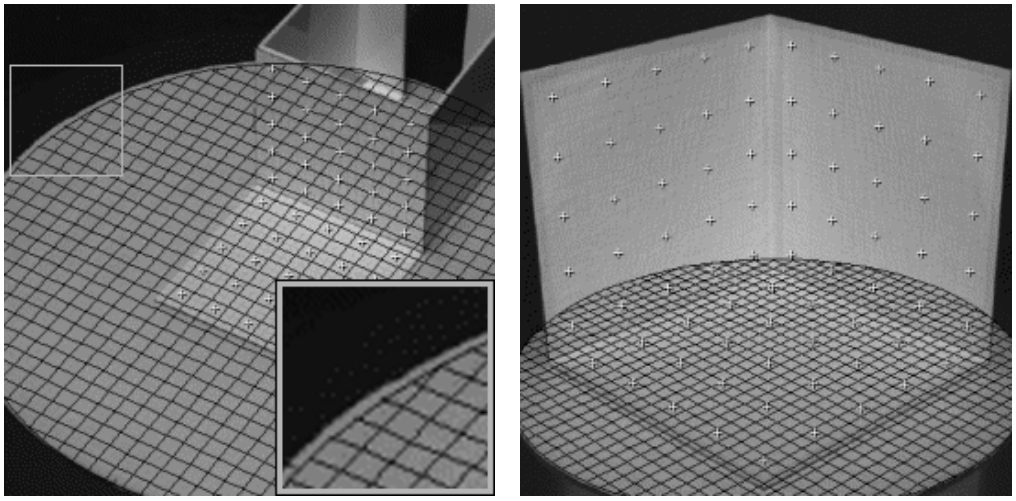


Figure 2.1: Calibration results for two different calibration objects: two planes each with 25 calibration points, the object is smaller than the image (left), and three planes each with 25 calibration points, the object covers the whole image (right). The calibration result is used to project the known calibration points into the image plane (labeled with crosses) as well as the known disc (labeled with a grid). In the left picture, a subpicture is enlarged showing the error at the border of the disc.

The deviations between the measured positions and the test points are used as quantitative error criterion. The mean square error (MSE) of these deviations is used as quantitative error criterion function. The calibration objects are placed on a disc and the geometric relation between disc (i.e. disc border) and calibration object is known. The calibration results are also used to project the disc into the image plane (labeled with a grid in Fig. 2.1). The deviation between this projected disc and the visible border of the disc is also used as a qualitative error criterion based on visual inspection.

The calculated MSE was larger in situations as shown in the left picture in comparison to the situation shown in the right picture of Fig. 2.1. The qualitative criterion did also lead to the conclusion that the calibration object should cover the whole image. The statistical results based on synthetic data (i.e. optimum calibration is nearly achievable with 3 planes, each with 20 calibration points or more) are also supported by this experiment with real calibration objects.

As a further experiment, the robustness with respect to errors in detecting calibration point locations in the image was studied using the following experiment.

Input data and ground truth: Points on calibration planes in 3-D space and projected images of these points are generated for an assumed central projection. The calibration is performed assuming Gaussian noise for these

calibration points in 3-D space, then these points are projected into the image plane, and again the projected image points are effected by Gaussian noise, with standard deviation 0.1 in both cases, i.e. 0.1 mm for world coordinates and 1/10 pixel for the projected image coordinates. The values of camera parameters, as distortion coefficients and focal length, are assumed for this ideal projection of 3-D points into the discrete image. Furthermore, the rotation matrix **R** and the translation vector **T** are known describing the coordinate transformation between the world coordinate system and the camera coordinate system.

Error criterion function: The ideal camera and transformation parameters and the calibrated parameters can be compared using different functions. Tab. 2.1 shows all the absolute differences in values.

parameters		**assumed values**	**calibrated values**
translation (in mm)	T_x	80.0	80.000
	T_y	100.0	100.011
	T_z	2000.0	1998.276
rotation (in deg)	yaw	60.0	59.994
	pitch	20.0	20.003
	roll	-5.0	-5.003
focal length (in mm)	f	8.0	7.998
distortion coefficients	κ_1	$-1.0 \cdot 10^{-5}$	$-1.3 \cdot 10^{-7}$
(in 1 / mm^2)	κ_2	$-1.0 \cdot 10^{-5}$	$-3.1 \cdot 10^{-6}$

Table 2.1: Results based on camera calibration with synthetic data assuming Gaussian errors for calibration points (before and after projection).

A synthetic calibration situation and the obtained values in this experiment are shown in Tab. 2.1. In practice it is not straightforward to detect calibration points within a digital picture at subpixel accuracy. A calibration point is projected onto a "point segment", and the centroid of these segments can be calculated, e.g., by weighted moments. A deviation of 1/10 pixel of detected point positions in comparison to the true point positions is used as theoretical estimate in this experiment. In the second phase of this experiment, the points previously used as calibration points are considered as surface points of a moving object in the scene space and their position is calculated based on the previously calculated calibration results. The experiment proves that a deviation of 1/10 in calibration point locations seems to be an acceptable error in point location detection. This experiment can be adjusted to the actual point location results of a certain calibration situation, e.g. if 1/10 pixel is not the appropriate error estimate.

3 STATIC STEREO ANALYSIS

The key problem in stereo analysis is how to find the corresponding points in the left and in the right image, referred to as the correspondence problem. Whenever the corresponding points are found, the depth can be computed by triangulation. Stereo techniques can be distinguished by either matching edges and producing sparse depth maps or by matching all pixels in the entire images and producing dense depth maps (see [24] for an overview of stereo techniques). The objective of the application always effects the decision whether the preference is given to dense stereo correspondence or to edge-based correspondence. Therefore, we took both types of approaches into consideration to find suitable solutions to the correspondence problem in stereo. A very efficient method to obtain dense stereo correspondence is presented in Subsection 3.2. In Subsection 3.3 we present the edge-based approach that we obtained the best matching results with so far.

Although good results can be achieved with stereo techniques, the excessively long computation time needed to match stereo images is still the main obstacle on the way to their practical applications. Computational fast stereo techniques are required for real-time applications. General purpose computers are not fast enough to meet real-time requirements because of the algorithmic complexity of stereo vision techniques. Consequently, the use of parallel algorithms and/or special hardware is inevitable to reach real-time execution. This Section discusses the *selection of correspondence method* problem and the *on-line static stereo* problem, see Subsection 1.1.

3.1 The Problem To Select A Correspondence Method

Worldwide, many research activities are known dealing with stereo vision. Nevertheless, there still does not exist a standardized way for the evaluation of the algorithms. The known methods differ in their solution to the correspondence problem as well as in the selection of constraints assumed for the visible objects in the scene. Moreover, nearly all publications exclusively present their own solution without comparing it to the results of other methods, and the methods are applied to rather different tasks (e.g., mobile robots, photogrammetry, stereo microscopy, etc.). The large number of distinguishable features in the solutions aggravates a direct comparison of the methods and it is nearly impossible to evaluate the suitability of a method for a selected application.

The considerations mentioned above encourage the necessity to create an experimental tool for the methodical investigation of computer vision methods. Another difficulty occurring with the evaluation of stereo methods is resulting from the direct interdependence between the single processing steps. It is not suitable to evaluate a selected single processing step (e.g. a stereo matching algorithm) without taking into consideration the interdependence of all processing

steps. Matching could be done easily if for example the correspondence search is reduced to characteristic features like junctions in the image. Although, the matching problem can be solved perfectly for those features, the obtained results are not suitable for a consecutive surface reconstruction process. Thus, the whole stereo vision cycle has to be examined and has to be assessed. Of course, the entire problem has to be divided into solvable subproblems, but the general view can not be ignored. The main emphasis of the investigations presented in [22] was on stereo matching because this processing step has a great influence on the quality of the computed results. In addition, the entire correlation and the mutual dependence of the processing steps within a stereo system were taken into consideration. The experiment can be described as follows:

Input data and ground truth: Correspondence methods are evaluated with regard to their suitability within a stereo system for the automatic registration of the geometry of an a-priori unknown 3-D object in near distance to the cameras (approximately 1 m). All methods are applied to a series of indoor images (see [22] for further details). Gray value or color images can be chosen.

Error criterion: Qualitative evaluations of calculated correspondences are used. The following Subsection contains representation examples of the computed 3-D data. Such visualizations were used for visual comparisons of the different methods.

Algorithms: We investigated 73 different stereo methods and implemented eight selected and two new matching techniques [22]. The ten methods were selected regarding their methodical distinction in solving the correspondence problem (area-based and feature-based, binocular and trinocular, statistical and physiology-based, etc.).

Results and conclusions: As a result of our comparison we found an approach based on disparity histograms [43] to be very suitable for edge-based correspondence. Furthermore, a technique based on Block Matching which was developed in our group [21] was found to be rather efficient and of high quality for obtaining dense stereo correspondence. Both techniques are detailed in the following subsection. In addition, we found that the quality of the matching results always improves when color information is used instead of gray value information. This holds for edge-based techniques and for dense techniques [25].

3.2 Dense Stereo Correspondence Employing Color Information

The computation of dense disparity maps defined for every pixel in the whole image is essential for a successful reconstruction of complex surfaces. Unfortunately, most of the existing dense stereo techniques are very time consuming (see e.g. [10, 35]). As mentioned above, we developed an efficient technique for obtaining dense depth maps based on Block Matching [21]. Furthermore, we extended the Block Matching technique to color images. Four different color

models ($RGB, XYZ, I_1I_2I_3, HSI$) and three different color measures have been investigated with regard to their suitability for stereo matching [23]. As a result the $I_1I_2I_3$ color space suggested by [34] provides the best information for stereo matching. The three coordinates are defined by

$$I_1 = \frac{R+G+B}{3}, \quad I_2 = \frac{R-B}{2}, \quad \text{and} \quad I_3 = \frac{2G-R-B}{4} .$$

The selection of the color measure had no significant influence on the results in this investigation. In addition, we applied the algorithm to gray value and color images to compare the performance. The precision of the matching results always improved by 20 to 25% when using color information instead of gray value information. Thus, high quality matching results can easily be obtained with Block Matching employing color information.

The principle of Block Matching is based on a similarity check between two equally sized blocks in the left and the right image E_L and E_R (area-based stereo). In general these blocks are $m \times n$-matrices, we assume $m = n = 2k+1$ for simplicity. The mean square error MSE between the pixel values inside the respective blocks defines a measure for the similarity of two blocks. This function MSE is defined for gray value images as follows:

$$MSE(x,y,\Delta) = \frac{1}{n^2} \sum_{i=-k}^{k} \sum_{j=-k}^{k} \left| E_R(x+i,y+j) - E_L(x+i+\Delta,y+j) \right|^2 ,$$

where Δ is an offset describing the difference $(x_R - x_L)$ between the column positions in the left and in the right image. This formula can easily be extended to color images when employing a color measure. As mentioned above, the selection of the color measure has no significant influence on the quality of the matching results. Therefore, we use the square of the Euclidean distance denoted as $dist_c$. For two colors $\mathbf{c}_1 = (R_1, G_1, B_1)$ and $\mathbf{c}_2 = (R_2, G_2, B_2)$ in the RGB color cube the measure $dist_c$ is defined as follows:

$$dist_c(\mathbf{c}_1, \mathbf{c}_2) = |R_1 - R_2|^2 + |G_1 - G_2|^2 + |B_1 - B_2|^2 .$$

The left color image $\mathbf{C}_L$ and the right image $\mathbf{C}_R$ can be represented in the RGB color space, e.g. $\mathbf{C}_L(x,y) = (R_L(x,y), G_L(x,y), B_L(x,y))$. MSE is replaced by

$$MSE_{color}(x,y,\Delta) = \frac{1}{n^2} \sum_{i=-k}^{k} \sum_{j=-k}^{k} dist_c(\mathbf{C}_R(x+i,y+j), \mathbf{C}_L(x+i+\Delta,y+j)) .$$

The block of size $n \times n$ is shifted pixel by pixel inside the search area in the right image. The disparity D between the blocks in both images is defined by the distance between those positions of the blocks (i.e. the difference in the columns) showing the minimum MSE or MSE_{color} value for both images. The search area can be limited in horizontal direction by a predefined maximum disparity D_{max}. A dense disparity map is generated when applying this pixel selection

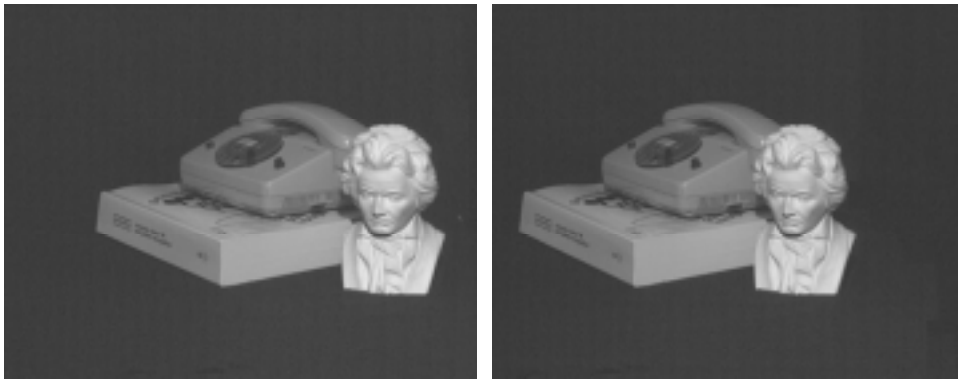


Figure 3.1: Gray value reproduction of original left and right color stereo image of the BEETHOVEN scene space (in PAL resolution 752×566 pixel).

technique to every pixel in the image. Afterwards, the pixel disparities are transformed with a median filter to avoid outliers.

The algorithm sketched above was applied to several test images representing scenes of different complexities. Here, we present results for scenes of the scene spaces BEETHOVEN and PACKAGE. An original stereo image of the scene space BEETHOVEN is shown in Fig. 3.1. The dense depth map obtained for this image with chromatic Block Matching is shown in Fig. 3.2 at the left, and the reconstructed scene is shown in Fig. 3.2 at the right. A dense depth map obtained when applying chromatic Block Matching to a PACKAGE stereo image is shown in Fig. 3.4 at the left. For further details see [23].

3.3 Edge-Based Stereo Correspondence

Dense depth maps are not always required for every application, and their computation is time consuming. Often, the computation of distances between the camera system and the objects in the scene is the exclusive aim of the stereo task. Thus, the correspondence search in stereo images can be reduced to a matching of a few image positions, e.g. characterized by edges. Edge-based stereo techniques have the advantage of being less sensitive to photometric variations. In an earlier investigation [22], we found that high quality edge matching results are obtained when a feature-based technique suggested in [43] is applied to the stereo images. The main idea of this binocular approach is based on disparity histograms showing the distribution of disparity values in the neighborhood of matching candidates in multiple resolutions. A standard stereo geometry is used to reduce the search space to horizontal lines.

Edges are extracted in both (intensity) stereo images applying the Marr-Hildreth or LoG operator, respectively, in three resolutions (σ_1 = 1.41, σ_2 = 3.18, and σ_3 = 6.01). Zero-crossings in the LoG filtered images constitute the

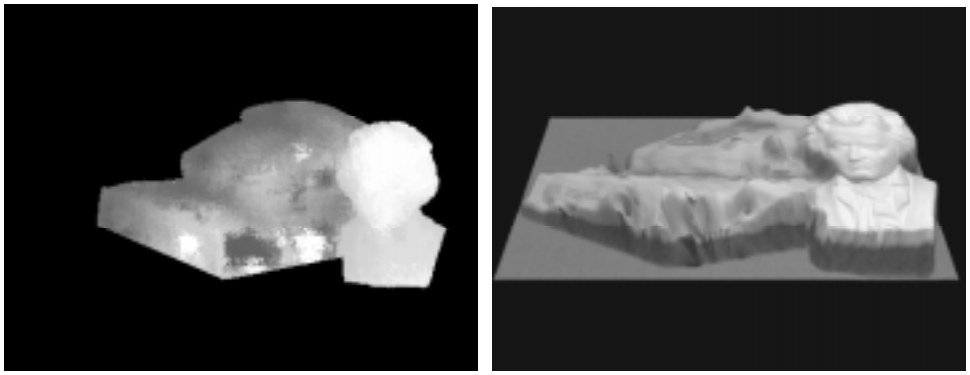


Figure 3.2: Gray coded dense depth map obtained with chromatic Block Matching (left) and gray value reproduction of the reconstructed BEETHOVEN scene using this dense depth map (right).

features in the succeeding matching process. The basic idea of the edge-based stereo approach is explained in the following, cp. also [43]. A zero-crossing is defined as a two-dimensional unit vector $\mathbf{e}(x,y)$ pointing into one of the two directions of the zero-crossing curve. A pair of zero-crossings, one in the right image and one in the left image, is regarded as a (possible) *matching pair* if the difference between the directions of the zero-crossings is less than 30 degrees. This is represented by matching functions M_R and M_L for the right and the left image, respectively: if $\mathbf{e}_R(x,y)$ and $\mathbf{e}_L(x+D,y)$ form a matching pair, then $M_R(x,y,D)=1$ and $M_L(x+D,y,D)=1$. Otherwise, let be $M_R(x,y,D)=0$ and $M_L(x+D,y,D)=0$, where D denotes a disparity.

First, the global disparity histogram GDH is determined to find approximate disparity intervals. The GDH represents the distribution of candidate disparities (including true and false matches) in the whole image. It is defined for the right image as

$$GDH_R(D) = \frac{\sum_{(x,y)\in\mathbf{R}} M_R(x,y,D)}{\sum_{(x,y)\in\mathbf{R}} \left\|\mathbf{e}_R(x,y)\right\|} \ ,$$

where $\mathbf{R}$ is the whole image raster. The function GDH_R suffices to estimate the disparity distribution. Based on the global disparity histogram, a candidate disparity interval I_α is determined in the following equation,

$$I_\alpha = \{\, D: \quad GDH_R(D) > \alpha H \,\} \ ,$$

where H is the peak value of $GDH_R(D)$ and α is a constant with $0 < \alpha < 1$. Local disparity candidates are estimated using local disparity histograms LDH. A local disparity histogram shows the disparity distribution of true and false matches

within the window $\mathbf{W}_\sigma$ of size $N_\sigma \times N_\sigma$ around a zero-crossing point, where $N_\sigma = \sqrt{2}\pi\sigma$. The local disparity histogram for a zero-crossing at (x,y) in the image is defined as

$$LDH_X(x,y,D) = \frac{\sum_{(x,y)\in\mathbf{W}} M_X(x,y,D)}{\sum_{(x,y)\in\mathbf{W}} \|\mathbf{e}_X(x,y)\|}$$

with $D \in I_\alpha$. Local disparity histograms are determined for the left and the right image ($X = L$ or $X = R$). Once local disparity histograms of all channels are computed, a best channel is selected for each window based on the first and the second largest peaks in the local disparity histograms. If the difference between the two peaks is the largest, the channel is selected. A function F is defined to check the reliability of the selected channel. $F_X(x,y,D_{max})$ is the difference between the largest peaks of the best channel in the window around position (x,y) in image $X = L$ or $X = R$, where D_{max} is the disparity showing the maximum peak. Matching is established if the values of the F functions are large and the difference between the disparity values in the right and left images is small.

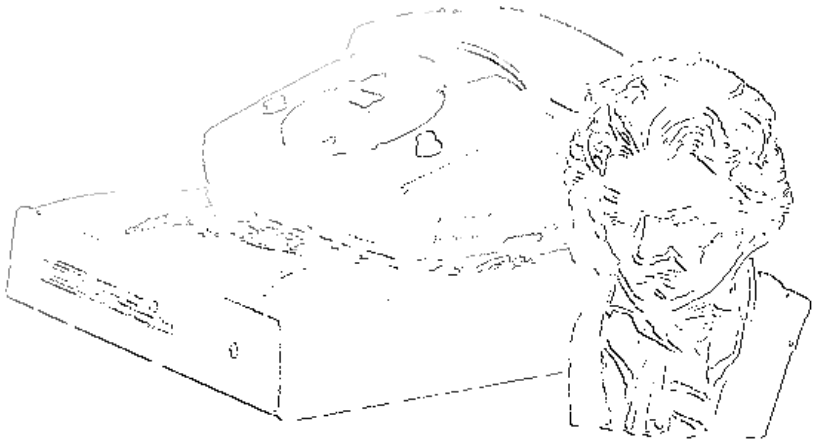


Figure 3.3: Gray value representation of the depth map obtained when applying the edge-based approach to the BEETHOVEN stereo image shown in Fig. 3.1.

Once the most likely disparity D_* is obtained in $\mathbf{W}_\sigma$, disparities of all zero-crossing points in $\mathbf{W}_\sigma$ and those of "finer channels" with smaller windows $\mathbf{W}_\omega$, where $\omega < \sigma$, are obtained by searching for the optimum disparity being the closest to D_*. For further details see [43]. Whenever a pair of zero-crossings is matched, they are removed from the sets of zero-crossings to reduce the number of remaining candidates. After trying to establish matches for all zero-crossings

inside the window $\mathbf{W}_\sigma$ and the windows $\mathbf{W}_\omega$ of finer resolutions, the algorithm (starting with global disparity histograms) is applied to the reduced feature lists. The matching process terminates if no new matches can be established.

The results are visualized in Fig. 3.3 obtained when applying this sketched algorithm to the stereo pair of BEETHOVEN images in Fig. 3.1. Furthermore, we compared the results obtained when applying the dense and the edge-based technique to PACKAGE stereo images showing the box characterized in Fig. 1.2. Results are shown in Fig. 3.4 and in Tabs. 6.1 and 6.2.

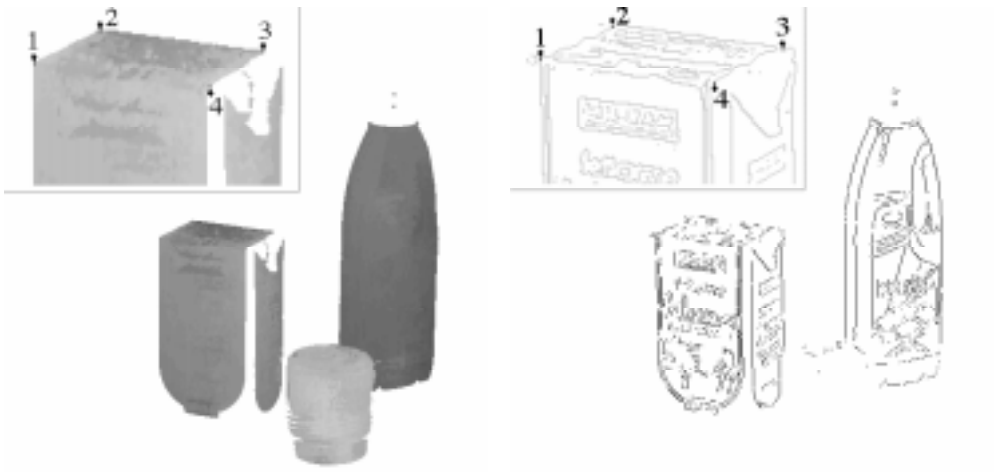


Figure 3.4: Depth values obtained to an PACKAGE stereo image when applying chromatic Block Matching (left), and points where the edge-based approach has matched zero-crossings (right). The camera was about 1.80 m away from these objects, and the baseline distance between both cameras was 11.7 cm.

3.4 Towards On-Line Static Stereo Analysis Using Parallel Algorithms

During the last years some hardware solutions to stereo analysis were already implemented. Neural networks and transputers are, for example, successfully employed for stereo [29, 31]. Parallel stereo algorithms were presented for the TMC Connection Machine [3] or special hardware [42]. None of these implementations produces dense depth maps. As mentioned before, we found the Block Matching technique using color information to be very suitable for dense stereo matching. Several ways exist to develop parallel algorithms for chromatic Block Matching. Currently, we divide both images into several segments and we compute *MSE* values inside every segment in parallel. For example, both color images can be divided into 8 segments. Now, *MSE* values can be computed in parallel for every segment using 8 processing units (PUs). An illustration of this procedure is given

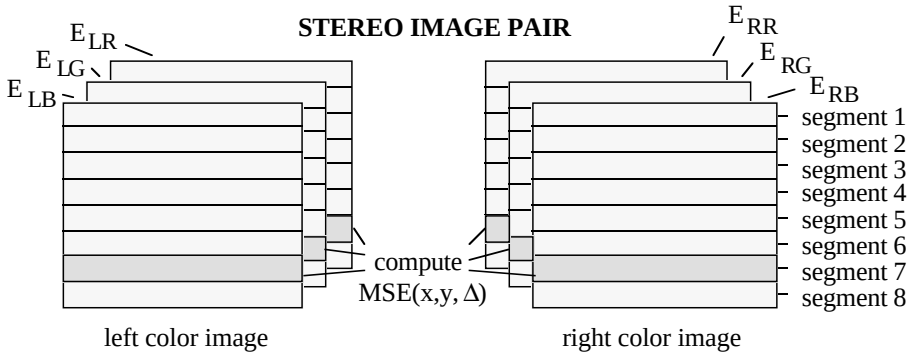


Figure 3.5: Illustration of the parallel Block Matching algorithm showing by example the computation of the *MSE* values in the segment number 7 (shadowed area) in the three color components in both color images.

```
BEGIN
    PARALLEL DO (in PU_i , 1 ≤ i ≤ 2)        { transform the left and right image i
        ConvertRGBtoI1I2I3_i ()                from RGB to I1I2I3 color space }
    END PARALLEL

    PARALLEL DO (in PU_S , 1 ≤ s ≤ 70)
        FOR d = 2 TO d_max DO                { search for corresponding blocks in
            BlockMatching_S (d)                horizontal segments by minimizing
        END FOR                                the MSE }
    END PARALLEL

    PARALLEL DO (in PU_S , 1 ≤ s ≤ 70)       { filter the block disparity image
        BlockMedian_S ()                       with a median approximation in
    END PARALLEL                             horizontal and vertical segments }

    PARALLEL DO (in PU_S , 1 ≤ s ≤ 70)       { compute pixel disparities from
        SelectPixel_S ()                       block correspondences }
    END PARALLEL

    PARALLEL DO (in PU_S , 1 ≤ s ≤ 70)       { apply the median approximation
        PixelMedian_S ()                       to the pixel of the disparity image
    END PARALLEL                             in horizontal and vertical segments}
END
```

Figure 3.6: Parallel Algorithm for Chromatic Block Matching (with up to 70 PUs).

in Fig. 3.5. In principle, both images can be divided into many segments (e.g., 70 segments for PAL resolution). Utilizing an individual processing unit for every segment speeds up the matching process.

As mentioned above, we found a slight improvement in the quality of the matching results when employing the $I_1I_2I_3$ color space instead of the *RGB* color

```
BEGIN
    PARALLEL DO (in PU_iC, 1 ≤ i ≤ 2, 1 ≤ c ≤ 3)      { search zero-crossings in
        FeatureExtraction_iC ()                          left and right image i ,
    END PARALLEL                                         for each channel c }
    DO
        PARALLEL DO (in PU_C, 1 ≤ c ≤ 3)              { compute global disparity
            ComputeGDH_C ()                              histogram and candidate
            Compute I_cα ()                              disparity interval for all
        END PARALLEL                                     channels independent }
        FOR (each feature in channel 0) DO
            PARALLEL DO (in PU_C, 1 ≤ c ≤ 3)          { for each channel c, calcu-
                ComputeLDH_C ()                          late local disparity histo-
                ComputeFXY_C ();                         gram, determine existence
            END PARALLEL                                 and magnitude of a peak }
            c ← SelectBestChannel ();
            IF (TestReliabilty (c) = OK) THEN
                MatchAndDeletePair (c);               { try to match the features in
            END IF                                       c and all channels with
        END FOR                                          finer resolution }
    WHILE (new features were matched)
END
```

Figure 3.7: Parallel algorithm of the edge-based stereo approach.

space. Image data have to be transformed from *RGB* to $I_1I_2I_3$ when this color space is used. Nevertheless, the principle of dividing a color image into several segments holds for every tristimulus color cube. A variant of the median filter, the separable *median of medians* [33], was implemented to accelerate image smoothing. Furthermore, we implemented pixel selection for every segment in parallel. The resulting parallel algorithm for chromatic Block Matching is outlined in Fig. 3.6.

Furthermore, we developed parallel algorithms for the edge-based stereo approach based on disparity histograms. We do not concentrate on the parallel implementation of the Marr-Hildreth operator since [45] presents a hardware implementation. In our parallel implementation, we detect edges in the left and in the right image in three resolutions in parallel. Afterwards, the global disparity histogram, the candidate disparity intervals, and the local disparity histograms are determined in parallel for the three resolutions. Parallel algorithms for computing the global disparity histogram and local disparity histograms are presented in [26]. The resulting parallel algorithm is outlined in Fig. 3.7.

We implemented the dense and the edge-based techniques on several different machines and we applied the algorithms to several different test images

to evaluate the time efficiency. In Tab. 3.1 and Tab. 3.2 some computing time examples are given using IRIX Power C on a SGI Power Challenge with twelve R8000 processors (75 MHz). The algorithms were applied to the BEETHOVEN stereo image of Fig. 3.1 in PAL resolution employing one, three, six or ten processing units (PUs).

computing time [in sec]	1 PU	3 PUs	6 PUs	10 PUs
conversion RGB to $I_1I_2I_3$	0.17	0.17	0.17	0.17
left				
right	0.17	0.17	0.17	0.17
estimating block disparities	26.32	9.16	4.77	3.05
block median	0.01	0.01	0.01	0.01
pixel selection	4.28	1.51	0.81	0.51
pixel median	0.36	0.13	0.08	0.07
total	31.30	11.16	6.01	3.98

Table 3.1: Computing time when applying chromatic Block Matching to the BEETHOVEN stereo image of Fig. 3.1 (752 x 566 pixels).

As a result, a very good acceleration was found for the parallel chromatic Block Matching algorithm. In general the computing time could be reduced up to the factor eight when employing ten processing units instead of one. Although color information was employed, the whole Block Matching algorithm consumes less than 4 seconds computing time when ten processing units are used in parallel. These results encourage an implementation on a highly parallel architecture.

computing time [in sec]	1 PU	3 PUs	6 PUs	10 PUs
feature extraction left	2.95	3.17	3.24	3.30
channel 0				
channel 1	5.78	6.03	6.14	6.21
channel 2	16.23	16.52	16.69	16.71
right channel 0	2.92	2.94	3.32	3.29
channel 1	5.80	5.82	6.10	6.19
channel 2	16.24	16.37	16.73	16.80
Subtotal	49.96	22.53	16.78	16.97
edge matching	62.40	31.29	32.28	33.67
total	112.36	53.82	49.06	50.64

Table 3.2: Consumed computing time when applying the edge-based approach to the image BEETHOVEN of Fig. 3.1 (752 x 566 pixels).

Opposed to this, our parallel algorithm for edge-based stereo is not very suitable for high parallelism. The best acceleration was achieved with three PUs because the results were computed in parallel for the three resolution channels. Further results can be found in [26].

4 MOTION ANALYSIS

Shape from motion is often cited as one of the basic computer vision approaches [8], and even some books in computer vision are focusing on that issue, e.g. [11, 28]. Rigid objects are projected into the image plane assuming a certain camera model. For a time sequence of such projections, motion vectors have to be computed. Based on these vectors, certain shape values of the objects can be determined leading to a (partial) 3-D representation of the projected objects [13, 14].

This Section contains contributions to the *selection of optical flow method* problem, the *existence of optical flow method* problem, and the *depth from point correspondences* problem, as specified in Subsection 1.1.

4.1 Calculation of Optical Flow Fields

The computation of high-accuracy dense motion vector fields is the most critical issue of surface from motion, cp. [1, 16, 17] for evaluations of optical flow algorithms. Approaches for solving this task can be classified as point-based differential methods, as region based matching methods, as contour-based methods, or as energy-based methods.

Point-based differential methods use spatial and time gradients of the image irradiance function. The point-based differential methods are favorite candidates for surface reconstruction approaches since complete and dense motion fields can be computed. However, the question is how accurate are these motion fields. For choosing an optical flow algorithm, the experiment specifications were as follows:

Test images and ground truth: Synthetic textured images were generated (periodic or autoregressive pseudo-random patterns) and used for simulating translations and rotations. In an experiment (A), motions of homogenous textured planes were used to calculate a sequence of images. In experiment (B), independent motions of a textured circle and of a differently textured background plane were used for computing image sequences. In both experiments the translation and/or rotation parameters were available as ground truth, i.e. the motion vectors $\mathbf{u}_* = (u_*, v_*)$ were known in all pixel positions in sub-pixel accuracy.

Error criterion function: Assume that an optical flow algorithm computes a dense motion field $\mathbf{u} = (u, v)$. The normalized sum of all relative errors (SRE) between $\mathbf{u}_*$ and $\mathbf{u}$ is defined as follows,

$$SRE(\mathbf{u}_*, \mathbf{u}) = \frac{1}{\#\,\text{pixels}} \cdot \sum_{\text{all pixels}} \frac{\sqrt{(u-u_*)^2 + (v-v_*)^2}}{\sqrt{u_*^2 + v_*^2}} = \frac{1}{\#\,\text{pixels}} \cdot \sum_{\text{all pixels}} \frac{\|\mathbf{u} - \mathbf{u}_*\|}{\|\mathbf{u}_*\|}.$$

This error criterion function is used in experiments (A) and (B). A different measure was used in [1], namely the normalized sum of all angular errors (*SAE*)

$$SAE(\mathbf{u}_*, \mathbf{u}) = \frac{1}{\#\,\text{pixels}} \cdot \sum_{\text{all pixels}} arccos\left(\frac{(\mathbf{u}_*, \delta t) \cdot (\mathbf{u}, \delta t)}{\|(\mathbf{u}_*, \delta t)\| \cdot \|(\mathbf{u}, \delta t)\|} \right),$$

with time δt between two images of the sequence. The *SRE* function was used in our experiments since it was more adequate to small (and large) motions, and since point-based differential methods can only detect small motions.

For experiment (A), some typical results are illustrated in Fig. 4.1. Surprisingly (also to the reader?), the original Horn/Schunck-method was quite tolerant to the different textures. The Nagel-method which uses the unknown 3-D positions of surface points is of theoretical interest only, and did not show any essential improvement in comparison to the Horn/Schunck-method. In general, all these differential methods fail if there is not sufficient "diversity" in the used texture.

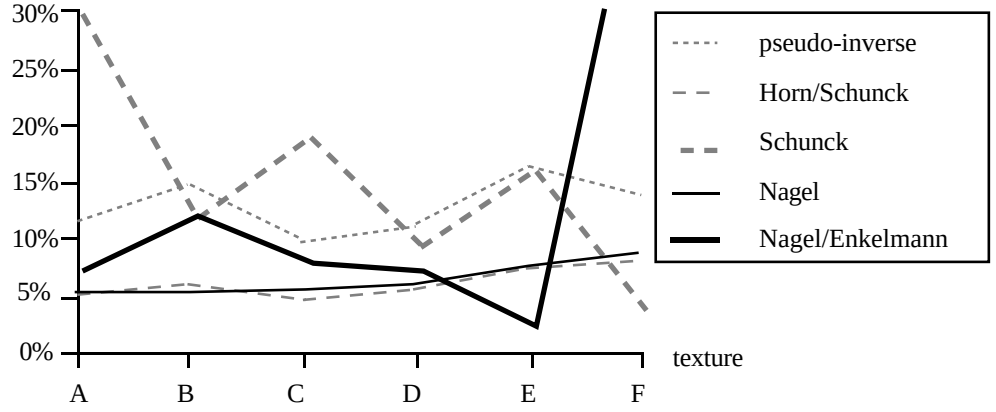


Figure 4.1: Examples of *SRE* values in experiment (A), for six differently textured planes where a small (constant) translation of these planes was simulated.

For experiment (B), some typical results are shown in Fig. 4.2. Here, for series of differently textured circles moving on textured planar backgrounds (inverse motions were used for generating "simple" motion boundaries), the errors did increase in comparison to experiment (A). Even after improving the initialization of the iterative methods by using a result of a non-iterative pseudo-inverse method for initialization, the *SRE* error was about 5% at best, where the Nagel/Enkelmann-method did behave best for this two-component motion. Note that such homogeneous textured circles on a homogeneous textured background represent a simple input for such a motion detection algorithm. The iterative Schunck-method was even more worse then the used non-iterative pseudo-inverse method!

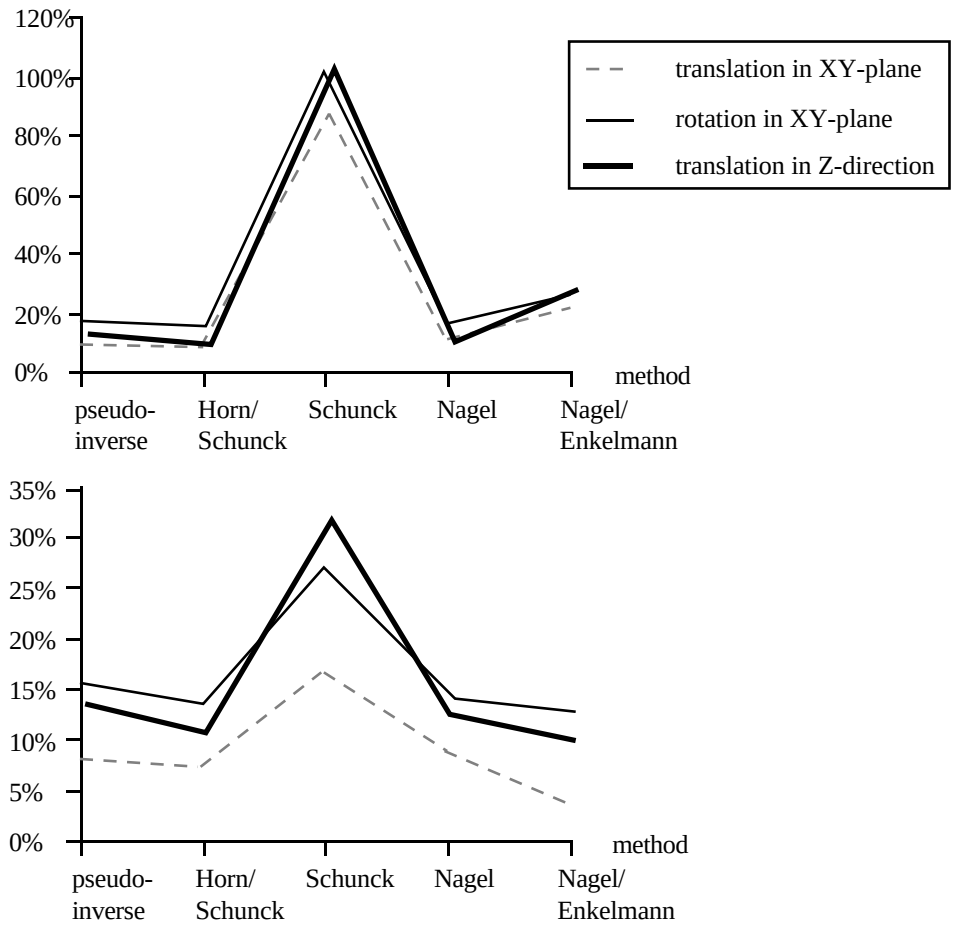


Figure 4.2: Averaged *SRE* values in experiment (B) for repeated motions of textured circles on differently textured backgrounds. Above: the start value $\mathbf{u}_0 = (0,0)$ is used for the iterative methods. Below: results of the non-iterative pseudo-inverse method are used as start values for the iterative methods (below a finer error scale is used).

Later on, also further motion detection algorithms, not listed in Fig. 4.1 or Fig. 4.2, were evaluated [17] using the source code of [1].[2] Because real objects often do not have "nicely textured surfaces", the experimental results were even more worse! Thus, for the following results on shape from motion, erroneous input data have to be taken in mind if images of real moving objects have to be analyzed.

---

[2] Programs obtained via ftp and own programs (as far as implemented), both following the same approach, often did lead to slightly different results. For us at least this did prove that theory, algorithms and implementations are different issues.

Based on a-priori knowledge about object motion, correspondence based motion detection can be constrained for obtaining improved results. Assume that objects are placed on a rotating disc, cp. Fig. 1.1, i.e. the radius of the disc and a fixed rotation axis are given (for calculating the axis, Tsai's camera calibration as discussed in Section 2 can be used). For this case of rotational motion [19], for correspondence calculation the epipolar constraint of static stereo can be adopted. In this specific (partially calibrated) case, optical flow vectors connecting corresponding image points can be used to calculate depth without going via shape.

4.2 Depth from Correspondence

By using dynamic stereo based on the rotating disc, for corresponding points in consecutive images depth can be computed directly without going via shape. Assume that during image acquisition of an object placed on the rotating disc, projections of points $\mathbf{C}_1$ and $\mathbf{C}_2$ in the image plane of the camera-centered coordinate system are given for the same visible surface point $\mathbf{W}$ in the world coordinate system, at consecutive time slots t and $t+1$. The task consists in calculating the coordinates of $\mathbf{W}$, where the Z-coordinate of $\mathbf{W}$ in the camera-centered coordinate system is identified with *depth*.

At first, assume that the rotation speed can be controlled, i.e. the rotation angle between time t and $t+1$ is known. Based on the calibration results, the defined task can be solved. It holds

$$\mathbf{RW} + \mathbf{T} = \mathbf{C}_1$$

where $\mathbf{R}$ denotes the calibrated 3×3 rotation matrix, and $\mathbf{T}$ denotes the calibrated translation vector. For the rotation $\mathbf{R}_\Delta$ of the disc between t and $t+1$, it holds

$$\mathbf{R}\mathbf{R}_\Delta\mathbf{W} + \mathbf{T} = \mathbf{C}_2 \ .$$

For the calibrated focal length f and the ideal image points $\left(x_{\mathbf{P}i}, y_{\mathbf{P}_i}\right)$ at time $t_1 = t$ and $t t_2 = t+1$, it holds that

$$x_{\mathbf{P}i} = \frac{f \cdot X_{\mathbf{C}_i}}{Z_{\mathbf{C}_i}} \ , \text{ and } \quad y_{\mathbf{P}i} = \frac{f \cdot Y_{\mathbf{C}_i}}{Z_{\mathbf{C}_i}} \ .$$

For the calibrated distortion coefficients κ_1 and κ_2, based on measurements of the distorted image coordinates, at first the ideal image points $\mathbf{P}_i$ can be computed, and secondly these ideal points can be used for determining points $\mathbf{C}_i$ in the camera-centered coordinate system,

$$\mathbf{C}_i = \begin{pmatrix} X_{\mathbf{C}_i} \\ Y_{\mathbf{C}_i} \\ Z_{\mathbf{C}_i} \end{pmatrix} = \begin{pmatrix} \frac{x_{\mathbf{P}_i} Z_{\mathbf{C}_i}}{f} \\ \frac{y_{\mathbf{P}_i} Z_{\mathbf{C}_i}}{f} \\ Z_{\mathbf{C}_i} \end{pmatrix} = Z_{\mathbf{C}_i} \begin{pmatrix} \frac{x_{\mathbf{P}_i}}{f} \\ \frac{y_{\mathbf{P}_i}}{f} \\ 1 \end{pmatrix} = Z_{\mathbf{C}_i} \mathbf{E}_i \ ,$$

by solving the following two equation systems. For abbreviation, let $Z_1 = Z_{\mathbf{C}_1}$, and $Z_2 = Z_{\mathbf{C}_2}$. For $\mathbf{E}_i$ as defined above it holds (cp. Fig. 4.3 at the left)

$$\mathbf{R}^T \left(Z_1 \mathbf{E}_1 - \mathbf{T} \right) = \left(\mathbf{R}\mathbf{R}_\Delta \right)^T \left(Z_2 \mathbf{E}_2 - \mathbf{T} \right)$$

There are three equations and two unknowns. In fact, the disc rotation angle φ_Δ can be taken as third unknown. But, the equation system "looses its linearity" if considered also for unknown φ_Δ.

For abbreviation, let $\mathbf{a} = \left(a_X, a_Y, a_Z, \right) = \mathbf{R}^T \mathbf{E}_1$, $\mathbf{b} = \left(b_X, b_Y, b_Z, \right) = \mathbf{R}^T \mathbf{T}$, and $\mathbf{c} = \left(c_X, c_Y, c_Z, \right) = \mathbf{R}^T \mathbf{E}_2$. Then it holds (cp. Fig. 4.3 at the right)

$$\varphi_\Delta = 2 \ arctan \left(\frac{c_2 \left(a_Y b_Z - b_Y a_Z \right) + c_1 \left(a_X b_Z - b_X a_Z \right)}{c_2 \left(a_X b_Z + b_X a_Z \right) - c_1 \left(a_Y b_Z + b_Y a_Z \right)} \right) ,$$

$$Z_1 = \frac{b_Z \left(c_X - c_X \ cos\left(\varphi_\Delta \right) + c_Y \ sin\left(\varphi_\Delta \right) \right)}{a_X b_Z - a_Z b_X \ cos\left(\varphi_\Delta \right) + a_Z b_Y \ sin\left(\varphi_\Delta \right)} ,$$

$$Z_2 = Z_1 \cdot \frac{a_Z - c_Z}{b_Z - c_Z} \ .$$

Despite of that the calibration results are very accurate, the flow vectors or the point correspondences are still the open problem for real objects on the rotating disc. For polyhedral objects, features computed by a corner response function were suggested for correspondence analysis [19]. For the practical evaluation of these reconstruction formulae, complex synthetic objects were considered. The experiment specifications were as follows:

Input images and ground truth: Synthetic objects (visualized by shaded surfaces) are assumed on a rotating disc in front of a camera. During rotation, several projections are computed, and exact correspondences are assumed. The visualized surface structure is used as qualitative ground truth.

Error measure: For interactive evaluation, the resultant surface is graphically represented (3-D triangulation and shaded surface).

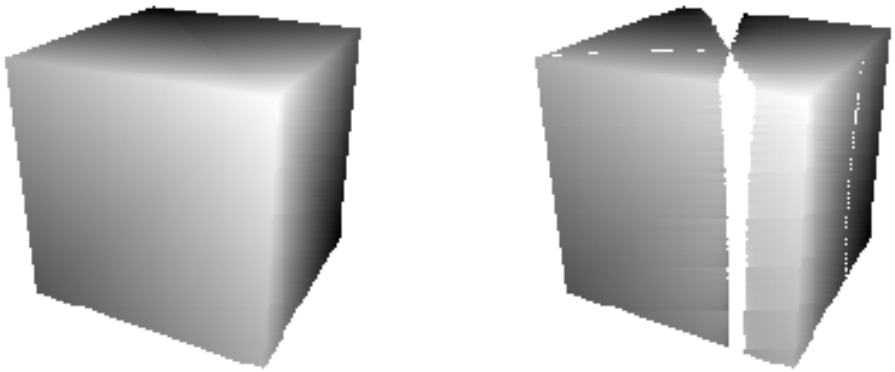


Figure 4.3: Reconstructed depth map of a cube if the rotation angle is known (left), and reconstructed depth map if this angle is unknown (right).

In Fig. 4.3 (left), depth reconstruction is illustrated if the rotation angle is known, or (right) if it is unknown. In the experiments, the algorithm with known rotation angle was robust for any 2-D motion of point $\mathbf{C}_1$ into $\mathbf{C}_2$ within the image plane. The algorithm with unknown rotation angle did not work if the direction of the motion vector (u,v) of point $\mathbf{C}_1$ into point $\mathbf{C}_2$ is "nearly parallel" to the image rows or to the image columns, i.e.

$$\frac{u}{v} \gg 1 \quad \text{or} \quad \frac{u}{v} \ll 1 \ .$$

So far, no mathematical explanation is available for this "bad behavior" in the case of applying the algorithm for unknown rotation angle.

As conclusion, a two-step procedure is proposed. At first the algorithm with unknown rotation angle is used for calculating the unique (!) rotation angle:

For some corresponding pairs $\mathbf{C}_1$ and $\mathbf{C}_2$, the rotation angle is calculated. Then, for the resulting angles a certain mean value is derived as unique rotation angle φ_Δ.

Then, the algorithm with unknown rotation angle is used to calculate depth values for all pairs of corresponding points $\mathbf{C}_1$ and $\mathbf{C}_2$.

In Fig. 4.4, the reconstruction results are illustrated of a more complex synthetic object than the cube of Fig. 4.3. Only two projected images were assumed. The correct motion field was available (motion vectors rounded to pixel positions !), the rotation angle was given (for correct flow vectors, the first approach is very robust for calculating this correct rotation angle), and the second approach was applied.

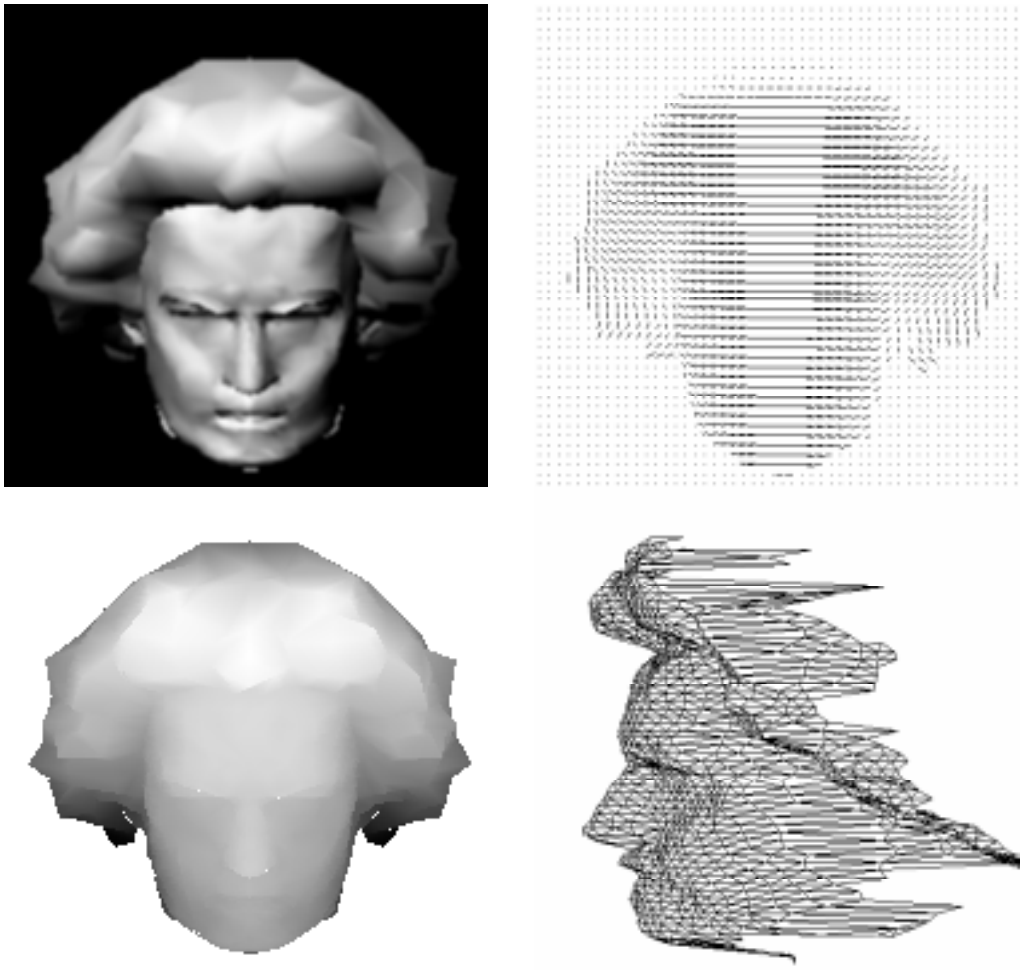


Figure 4.4: Synthetic 3-D object, its motion vector field simulating the rotating disc, the reconstructed depth map based on two (!) projections only, and a 3-D visualization of these depth values as used for interactive error analysis.

Because the sketched calibration method of Section 2.2 is very accurate, the same could be qualitatively evaluated for the defined surface reconstruction experiment.

Unfortunately, so far automatic dense flow vector field computation is not available at a quality level allowing similar reconstructions for real objects just by using one of these two approaches. By adding noise to ideal motion vector fields it became clear that relatively small distortions will have a great impact on the reconstructed surfaces.

The experiment specifications for the BEETHOVEN scene were as follows:

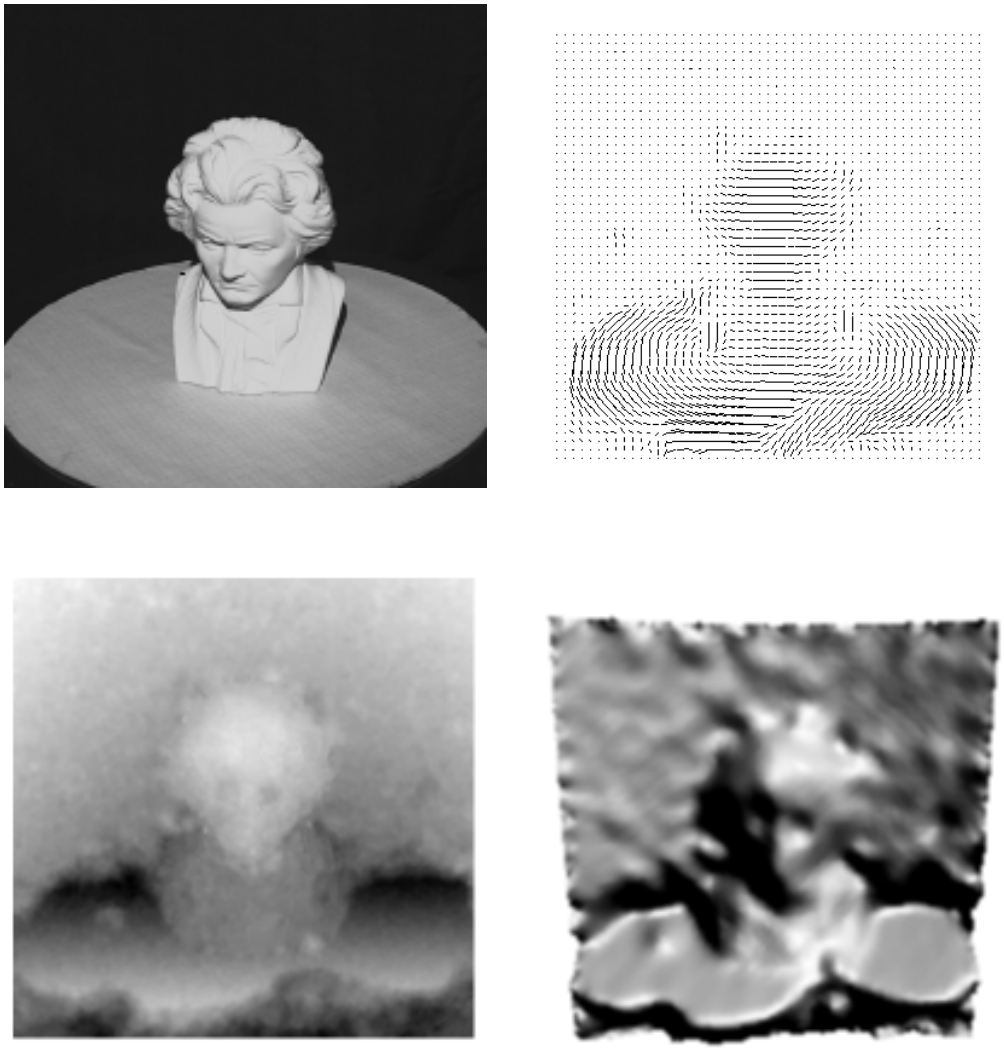


Figure 4.5: A plaster statue on the rotating disc, a needle representation of the optical flow field calculated by the Anandan method, cp. [1], and two representations of the reconstructed 3-D surface, namely a depth map (below left) and a shaded surface (below right). Cp. also Figure 6.1 for texture mapping.

Input images and ground truth: Objects as the plaster statue were placed on a rotating disc in front of a camera. During rotation, several pictures were taken. The visible surface structure was used as qualitative ground truth.

Error criterion: For interactive evaluation, the resulting surface was graphically represented (depth map, floating horizon, isolines, 3-D triangulation, shaded surface, texture mapping).

In Fig. 4.5 it is illustrated what quality of depth map representation or shaded surface representation was obtainable. Comparing many point-based differential methods for computing the optical flow, the Anandan-method was chosen to behave best for this experiment (by comparing with evaluations in [1], note that *dense* flow fields have to be used in this experiment). The depth map in Fig. 4.5 shows smoothed depth values.

The high error rate of such a motion based surface reconstruction technique is without doubt. The surface of these plaster statues is not covered with homogeneous surface textures as assumed in the experiments in Section 4.1. Even such textures did lead to quite erroneous motion fields. The reconstruction algorithms, either for known or for unknown rotation angle, produce very rough surface drafts.

However, these rough drafts can be of interest for the control of 3-D environments, i.e. it is not possible to recognize an object, or to reconstruct the surface in a precise way, but it can be calculated "that there is something" what a robot may view as an obstacle. Also, all the processes can be in parallel what makes on-line processing achievable.

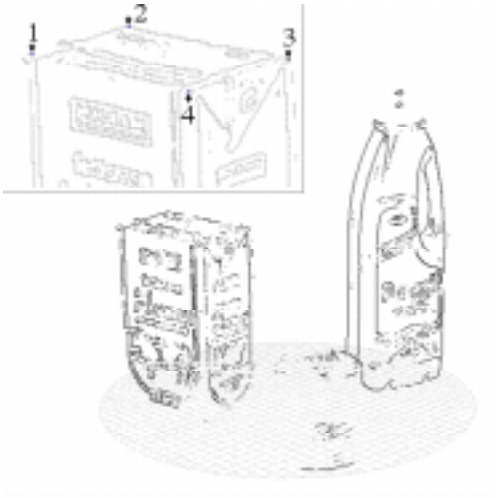


Figure 4.5: Localization of isolated points in 3-D space using motion analysis.

Dense optical flow field computation does not lead to complete surface reconstructions at acceptable quality. However, singular correspondences ("sparse flow fields") can be calculated for real objects with good accuracy, e.g. by interactively supported correspondence assignments, allowing computations of a few 3-D positions of surface points (these can be called *fixation points*). Such fixation points of a PACKAGE scene are illustrated in Fig. 4.6.

This figure illustrates a result of the following processing steps: A sequence of four input images was used. The rotation angle between two images was 5°. The camera was about 1.80 m away from these objects. The rotational disc is shown in the image just for illustration purposes. The features as shown in Fig. 4.6 are calculated with the Marr-Hildreth operator. These features were used for calculating corresponding points. The known rotation angle was used for reconstruction. In the enlarged subimage, four points are shown which were interactively selected. The reconstruction algorithm did calculate the 3-D world coordinates $\mathbf{P}_1$ = (-116.3, 87.7, 168.7), $\mathbf{P}_2$ = (-124.6, 26.3, 167.4), $\mathbf{P}_3$ = (-34.3, 13.2, 167.1), and $\mathbf{P}_4$ = (-29.1, 74.6, 166.0) for these four points, all values are scaled in mm. Further results are given in Section 6 in comparison with reconstruction results of the other methodologies.

5 SHADING BASED SHAPE RECOVERY

Shading based methods for surface reconstruction consider the imaging sensor as a photometric measuring device [7]. Assuming special reflection characteristics and known illumination configurations, the measured image irradiances (intensities) reduce possible surface orientations to a closed curve on the Gaussian sphere. For Lambertian reflection these curves are circles. Several approaches exists to achieve a unique surface in spite of this ambiguity in image irradiance. Most approaches require hard limitations to the object world. This implies that these approaches are theoretically very interesting but not useful in practice.

However shading based methods have the important advantage that three-dimensional features they calculate can be recovered in a dense manner. Therefore, a surface orientation or a relative depth value can be assigned to each visible surface point in the image, except for sharp discontinuities.

In our work we focus on methods which are independent of the actual surface albedo. The reasons are practical considerations. Albedo independence allows the surface to be of an arbitrarily colored texture. Hence, no reflection factor measurements are needed for the original unknown surface.

5.1 Photometric Stereo Analysis with Table Look-Up

Using at least three light sources surface orientations together with albedo values can be measured for each image point independently. This is known as the *photometric stereo method* [48]. Algorithms following this methodology calculate surface shape (orientations, gradients or normals) from the shading variations in three images, taken of the objects with light sources in different positions. Opposite to the binocular stereo techniques (Section 3), the visual sensor is fixed in the same position and orientation. Opposite to the dynamic stereo techniques (Section 4), the illumination changes during picture sequence acquisition. Therefore no matching or registration strategy is necessary. From the surface orientations the depth can be calculated using an integration technique. Comparisons with several approaches have shown that a FFT-based method [5] produces the best results.

Applying the photometric stereo method to a large number of different objects it has turned out that this shading based approach can serve as a good starting-point for high-quality surface reconstruction. The investigations have shown, that the photometric stereo method is open to improvement in several ways. The most important problems for applying this method are listed in Subsection 1.1, listed there as the last six problems.

All these problems concern the accuracy of the surface reconstruction. Some problems are furthermore related to practical considerations, i.e. what

techniques improve the applicability of the calibration and of the reconstruction processes.

The *inaccurate illumination parameter estimation* problem can be solved if the influence of these parameters on reconstruction can be reduced. Traditionally for the photometric stereo method [48], the accuracy of the orientation determination depends strongly on the estimation of the light source parameters, i.e. the light source strength and the illumination direction. On the other hand, table look-up techniques exist which need no knowledge of these parameters, since the table look-up is built with an calibration object [8]. Table look-up techniques in general have the disadvantage that only objects with surface materials can be analyzed which have exactly the same reflection characteristics as the calibration object. This implies many practical problems, since often it is difficult to generate a calibration object with a special surface material, and since the albedo of the analyzed object has to be constant. We developed a table look-up technique which is independent of the surface albedo. The light source parameters remain unknown. In comparison to other table look-up techniques, a further assumption is that the sensor has a linear characteristic.

Also some remarks to the *extension to non-static scenes* problem. Multi-irradiance shading based methods assume that the object does not move with respect to the sensor and the light source. It is assumed that the measured irradiances triplets come from the same scene point. Using a color sensor as imaging device it is possible to overcome this limitation [4, 38, 39], i.e. also non-static scenes can be analyzed for (partial) surface reconstructions during motion.

5.2 Reconstruction of polyhedral objects

This Subsection deals the *reconstruction of polyhedral objects* problem as defined in Subsection 1.1. For generating polyhedral reconstructions the first partial derivatives generated from the photometric stereo method have to be transformed to a depth map or to a geometrical $2\frac{1}{2}$-D model. We have developed a method [2] that applies dense gradient information as well as a line drawing of the object. The method consists of two steps. The visible surface of the projected object is segmented into planar and curved patches. Then these segments are fitted together to build up a $2\frac{1}{2}$D surface.

Region growing is used for segmenting the *gradient images*. The values of these symbolic images are the computed gradients. This segmentation technique allows to extract planar and curved patches. Curved patches can be approximated by planar patches, and these planar patches are attributed as belonging to a curved segment. This is necessary for the treatment of occlusions and the elimination of approximation edges in the recognition part. Since region boundaries in the interior of curved patches are determined by the growing process, these patches are post-processed with a balancing algorithm. Pixels on region boundaries are reclassified if the average orientation in an adjacent region

has a smaller angular deviation than the original region. The reclassification process is done iteratively until an equilibrium is reached. Subsequently to this process the boundaries are polygonally approximated. Consequently, for the next steps consistent region boundary information is available.

Now, from the region and the boundary data a winged edge model is generated. The model includes the vertices, edges, faces and the face orientations from the 2-D structure. This structure must be modified if concave objects with partially occluded boundaries occur. In this case, we have to assign more than one depth value to some vertices. To prepare the depth calculation, such occluding edges have to be detected. Occluding boundaries are detected by using the face orientations. Parallel projection is assumed. Therefore, the expected edge orientation depends on the adjacent face orientations $\mathbf{n}_1$ and $\mathbf{n}_2$ as follows,

$$(x,y)^T = (q_2 - q_1, p_1 - p_2)^T = proj_{XY}(\mathbf{n}_1 \times \mathbf{n}_2),$$

where $\mathbf{n}_1 = (p_1, q_1, -1)^T$, $\mathbf{n}_2 = (p_2, q_2, -1)^T$ and $proj_{XY}$ denotes the projection function on *XY*-coordinates (i.e. the parallel projection as introduced in Subsection 2.1).

If this orientation is inconsistent with the line drawing, the edge becomes an occluding edge. Vertex splitting is carried out if both adjacent edges are occluding edges, or if one edge is occluding and the other is a 3-D boundary edge.

The depth is locally calculated for each vertex. The depth value of a vertex constrains the depth values of all adjacent face vertices to some extend. Therefore the following five cases are distinguished:

(i) At no adjacent vertex any depth value is available: the current vertex is neither constrained with respect to its 2-D coordinates nor with respect to its depth value. The depth value can arbitrarily be chosen.

(ii) For exactly one of the adjacent faces a 3-D fixation is already available: the 2-D coordinates of the vertices can be substituted into the plane equation. This determines the depth values of these vertices.

(iii) For exactly two of the adjacent faces a 3-D fixation is already available: a 2-D inconsistency of vertices can arise. Therefore a parallel projection onto the line of intersection of the two planes is determined. Thereafter the point of intersection is substituted into the plane equation.

(iv) For exactly three of the adjacent faces a 3-D fixation is already available: consistency and depth can be attained simultaneously by calculating the intersections of the planes.

(v) More than three faces are fixed in 3-D space: if there is more than one point of intersection, than this inconsistency cannot be repaired. Therefore the depth calculation is scheduled in an order dependent on the number of adjacent faces. Such accidental events occur very rarely.

This procedure ensures that face orientations determined by the shape recovery method leave the assembling process unchanged. The edge structure obtained by such a process can vary. Calculated surface slopes at or close to edges are less reliable than in the interior of regions. For each face surface orientations are calculated within the whole segment, and all these orientations are used to determine a unique orientation of the face.

The above scheme (i) ... (v) is applied to each visible vertex. When the process is finished the 2-D winged edge model derived from the line drawing is transformed into a $2\frac{1}{2}$-D model. The following experiment is performed for images of the PACKAGE scene space.

Input data and ground truth: Three light sources were used, cp. Tab. 5.1. Images are taken if exactly one of the light sources is turned on (i.e. typical photometric stereo arrangement). The sizes of the pictured box are used as ground truth, see Fig. 1.2.

Error criterion: The deviations between reconstructed sizes and actual sizes are used as quantitative error criterion.

Algorithm: The table look-up technique (Subsection 5.1) is used for calculating surface gradients. Subsequently the polyhedral reconstruction scheme (i)...(v) is applied.

The parameters shown in Tab. 5.1 are given to describe the illumination configuration. However, they were not used within the surface orientation calculations.

light source	relative strength	estimated p_s	estimated q_s	estimated tilt	estimated slant
1	1.0	-0.311455	-0.231252	-143.41	21.20
2	1.02206	0.048908	0.303985	80.86	17.11
3	0.77149	0.410855	-0.235918	-29.86	25.35

Table 5.1: Light source parameters as used in the experiment with PACKAGE scenes.

In Fig. 5.1 one of the input images and a needle map representation of the reconstructed gradient image are shown. The cow pictured at the box has a very small albedo, i.e. close to zero. Therefore (among other regions) in the cow no surface orientations could be determined.

Fig. 5.2 shows perspective plots of the reconstructed box seen from two different view directions. Only an $2\frac{1}{2}$-D model can be recovered from a single sensor position, i.e. invisible parts of the milk can not be included in this model.

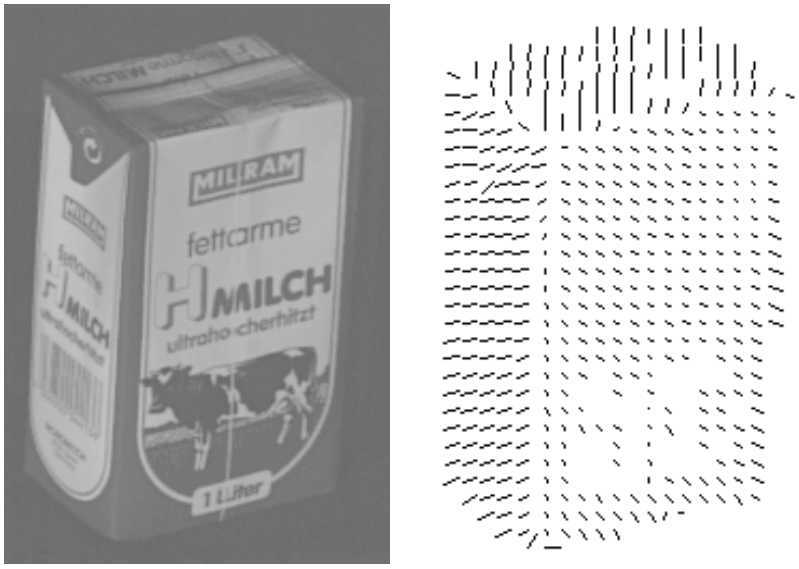


Figure 5.1: A photometric stereo input image of a milk package and a needle map representation of the reconstructed gradient image using the albedo independent photometric stereo technique, cp. Subsection 5.3.

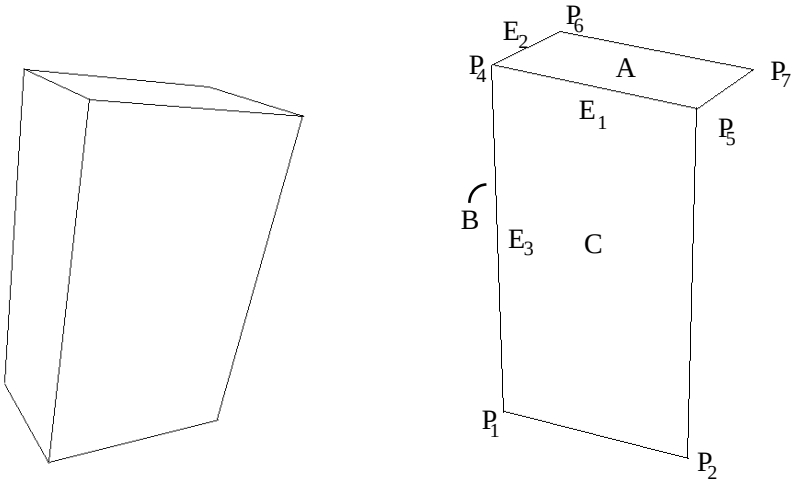


Figure 5.2: Perspective plots of the $2\frac{1}{2}$-D reconstruction.

vertex	X	Y	Z
$\mathbf{P}_1$	-0.186663	-0.763393	0.062641
$\mathbf{P}_4$	-0.133929	0.320592	-0.567323
$\mathbf{P}_5$	0.504743	0.461217	-0.204844
$\mathbf{P}_6$	-0.391741	0.508092	-0.220501

Table 5.2: Examples of 3-D coordinates: the four visible vertices are labeled as in Fig. 5.2. An object coordinate system is used.

face	X	Y	Z
A	0.054525	0.894764	-0.443198
B	-0.836804	-0.244101	-0.490076
C	0.513713	-0.449641	-0.730699

Table 5.3: Examples of unit normals of three box faces labeled as in Fig. 5.2.

Tab. 5.2 and Tab. 5.3 summarize some of the data of the generated $2\frac{1}{2}$-D model. The values in Tab. 5.2 are given in an object coordinate system. In Tab. 5.3 the three determined unit surface normals are listed for the three visible faces. The angles between these faces are as follows,

$$angle(A,B) = 92.7°,\ angle(A,C) = 92.9°,\ angle(B,C) = 87.8°.$$

From this data the ratios of the edge lengths can be calculated: $\mathbf{E}_3$:$\mathbf{E}_1$ = 1.6783, $\mathbf{E}_3$:$\mathbf{E}_2$ = 2.6638, $\mathbf{E}_1$:$\mathbf{E}_2$ = 1.5872. The real edge ratios are: $\mathbf{E}_3$:$\mathbf{E}_1$ = 1.7579, $\mathbf{E}_3$:$\mathbf{E}_2$ = 2.6508, $\mathbf{E}_1$:$\mathbf{E}_2$ = 1.5079.

In Fig. 5.3 at the left an iso-depth plot of a depth map determined from the data of the reconstructed $2\frac{1}{2}$-D model is shown. For comparison the gradient image has been transformed to a depth map using the FFT-method [5]. An iso-depth plot of this depth map is shown in Fig. 5.3 at the right. Using edge $\mathbf{E}_2$ as the reference edge the calculated lengths differ by 5 mm ($\mathbf{E}_1$) and 1 mm ($\mathbf{E}_3$).

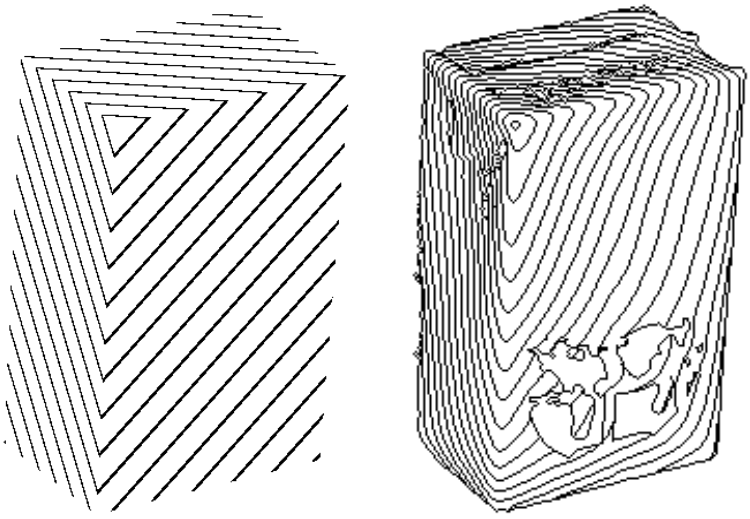


Figure 5.3: Iso-depth plots of the box based on the reconstruction results of the albedo independent photometric stereo technique, see Subsection 5.3. Left: calculated iso-depth map using the approach in [5]. Right: depth map generated from the reconstructed $2\frac{1}{2}$-D model using the polyhedron reconstruction method.

5.3 Using More General Reflection Models

Traditionally the photometric stereo method restricts the reflection characteristics of the analyzed surface to Lambertian reflection. This approximate assumption can be used for many materials. However, in general surface reflection is composed of a diffuse and of a specular component. This Subsection deals with this *using more general reflection models* problem.

With the specular reflection component more parameters must be known in advance to model this hybrid reflection. These are roughness values and weighting factors, for example. Using color is a good way to get more information about the scene without introducing more light sources and without making severe restrictions to the surface reflection properties. We apply the dichromatic reflection model. This model can be used to separate the two types of reflection components [20, 40, 41]. Moreover, roughness values and weighting factors can be extracted from the images. Since the photometric stereo method additionally recovers surface albedo values, illumination independent surface color descriptors can be obtained. These color descriptors allow that a set of CIE tristimulus values can be assigned to each surface element. The next experiment illustrates the influence of highlights on surface reconstruction.

Input images and ground truth: Scenes of a WATERING CAN arrangement are used as input data. In general no ground truth data are available for complex curved objects besides the visual appearance. If surface data are available as ground truth, the comparison of this data with the reconstruction data is difficult and instable (see the *3-D surface error measure* problem at the end of Subsection 1.2). Therefore it is not easy to find sufficient data to make comparisons. An object with tractable subsurfaces is chosen in this experiment. The watering can (without its spout) shown in Fig. 5.4 at the left has a body which can be locally approximated as being cylindrical. This means that most surface points have approximately a zero principal curvature. Furthermore the body has a symmetrical shape and the non-zero principal curvature is small at the symmetry axis. We can expect that the change in slant (i.e. angle between optical axis and surface normal) is small in a region nearby the symmetry axis.

Error criterion: The shape of the slant histogram in the region close to the symmetry axis is chosen as qualitative error criterion.

Fig. 5.4 at the left shows one of the three input images of the photometric stereo shape recovery approach. In Fig. 5.4 at the right a sketch of a horizontal cut of the can is shown. The position of the cut is drawn in Fig. 5.4 at the left as a horizontal line.

Tab. 5.4 specifies the light source parameters of this experiment. The light source strengths are given relatively to the first light source. The illumination directions are estimated with an inverse photometric stereo method [8] based on 150

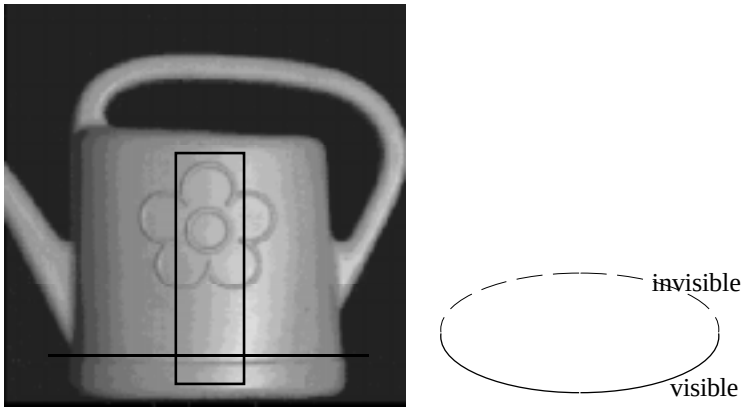


Figure 5.4: Left: a watering can illuminated with the third light source, see Tab. 5.4. Right: cut through the watering can body. The cut is indicated in the left part as a horizontal line.

light source	relative strength	estimated p_s	estimated q_s	estimated tilt	estimated slant
1	1.0	-0.323552	-0.157596	-154.03	19.79
2	0.79952	0.009482	0.572829	89.05	29.81
3	0.76223	0.450235	-0.012140	-1.54	24.25

Table 5.4: Light source parameters of the experiment with the watering can.

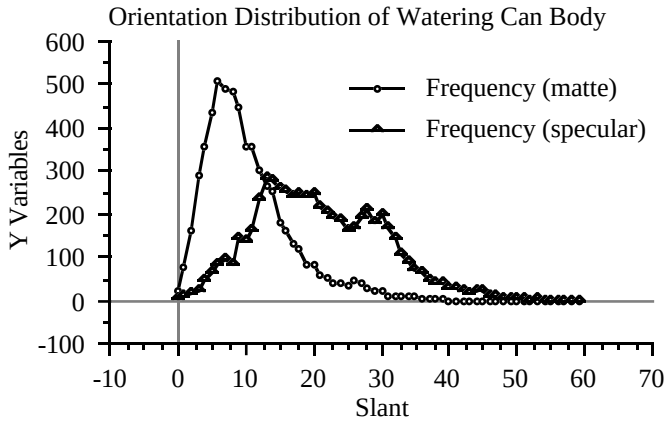


Figure 5.5: Slant histograms of the watering can body region indicated in Fig. 5.4.

normals. They are given as gradient space co-ordinates (p_s, q_s) and as spherical coordinates (tilt, slant).

The highlight area on the body of the watering can is quite large. The photometric stereo method was applied to the original specular input images and to the

matte input images after eliminating the specular component. Fig. 5.5 shows the slant histogram of the rectangular area drawn in Fig. 5.4.

It can be seen that the slant angles of the original input images are scattered. The slant angles obtained from the specular-free images generated by our method are much more concentrated and therefore more reliable. The maximum frequency is achieved at 6°, with a slant bin interval of one degree. This value is not zero since the symmetry plane of the whole watering can and of the projection (image) plane were not exactly coplanar during image acquisition.

Fig. 5.6 shows the needle map of the watering can using the matte images. It can be seen that the normals in the area with high specular component have small Y coordinate components. In the case of highlight influence they are tilted towards the respective light source.

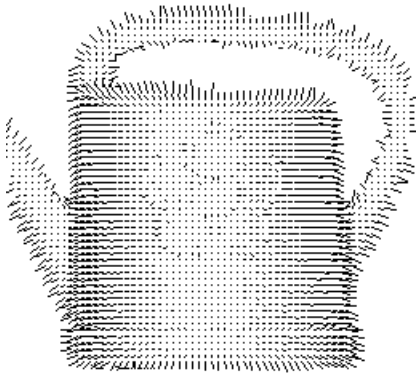


Figure 5.6: Needle map representation of the calculated gradient image where the reconstruction was based on the matte images of the watering can.

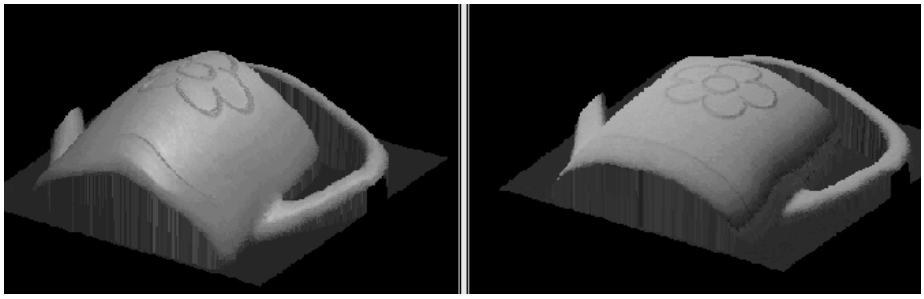


Figure 5.7: Reconstructed surfaces of the watering can without highlight elimination (left) and with highlight elimination (rig ht).

Fig. 5.7 at the left shows the rotated reconstructed surface. A texture mapping using the original image is applied to visualize the range data. The reconstructed can is strongly deformed. The specular reflection component locally causes a

dent. The result of surface reconstruction using three matte images is shown in Fig. 5.7 at the right. Here a calculated matte image is mapped onto the range data.

The next experiment aims at the three-dimensional analysis of BEETHOVEN scenes using the photometric stereo method. The FFT-based integration method [5] is used to calculate a relative depth map from the calculated surface orientations. The light source parameters used in this experiment are listed in Tab. 5.5.

light source	relative strength	estimated p_s	estimated q_s	estimated tilt	estimated slant
1	1.0	-0.387930	0.059682	171.25	21.43
2	0.56271	0.031802	0.556538	86.73	29.14
3	0.52581	0.389363	0.104828	15.07	21.96

Table 5.5: Light source parameters of the experiment for reconstructing the surface of the Beethoven plaster statue.

The albedo of this object is constant. Plaster has approximately a Lambertian reflection characteristic. Therefore no highlight elimination is necessary. The input image illuminated from the first light source is shown in Fig. 5.8 at the left. Fig. 5.8 at the right is a visualization using a mesh plot. The sampling rate of the mesh is three pixels. Therefore details in the depth map are lost.

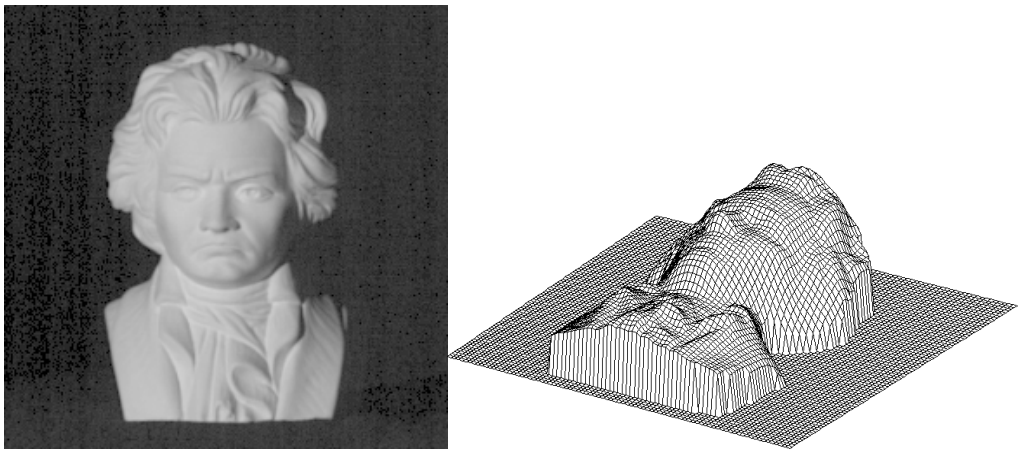


Figure 5.8: Left: input image of the Beethoven plaster statue illuminated with the first light source. Right: mesh plot of the resultant depth map.

The albedo independent table look-up method (Subsection 5.1) is used to reconstruct the surface orientations in areas where three irradiances are available. In areas illuminated from two light source the surface orientations are projected

onto the self-shadow circle of the third source on the Gaussian sphere. In areas illuminated from one source the orientations are projected onto the intersection of the two shadowing sources. In Fig. 5.9 a surface plot is shown using texture mapping. As texture image the scene illuminated with the second light source was used.

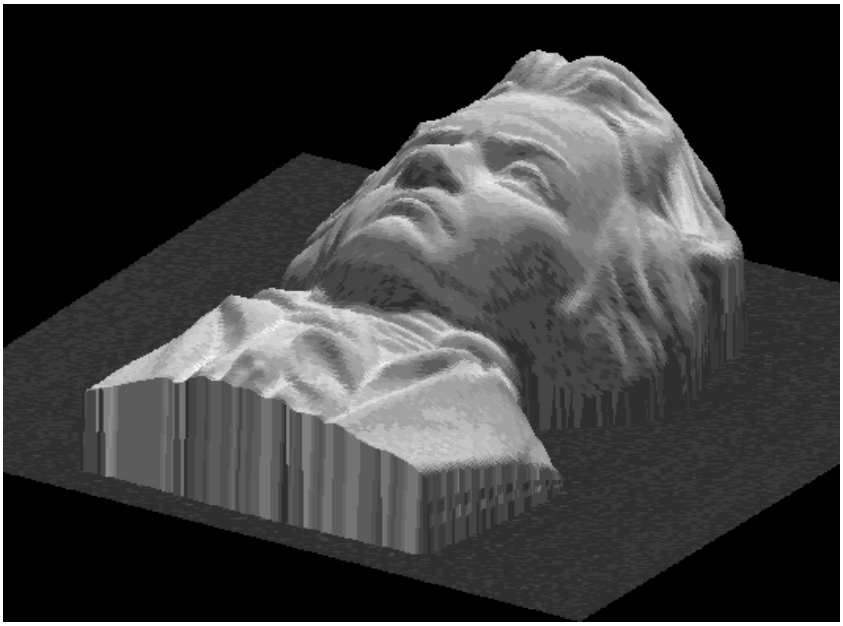


Figure 5.9: Visualization of the reconstructed surface of the Beethoven plaster statue using texture mapping.

Two reflection problems mentioned in Subsection 1.1 are not sketched in this Section. For dealing with the *treatment of shadow regions* problem, shadows have to be recognized. The photometric stereo method needs three non-zero irradiance measurements at each surface location. For convex surfaces the degree of shadowing depends on the illumination configuration and only self-shadowing occurs. Concave surfaces cause self-shadows and cast-shadows. For and contributions to the *consideration of interreflections* problem, see [27, 32, 37].

6 CONCLUSIONS

The reports discusses some solutions of the problems listed in Subsection 1.1 in different depth. The main aim was in illustrating the state of the art where a polyhedral object (box) and a curved object (plaster statue) was used for comparisons.

Static and dynamic stereo analysis allows to measure absolute depth values. For the box used in the experiments, such measurements are given in Tab. 6.1. Note that different world coordinate systems were used in the static and in the dynamic approach. From these measurements, the edge sizes of the box can be calcutated, see Tab. 6.2. These edge sizes allow a direct comparison, e.g. also with the ground truth (see also Fig. 1.2). Photometric stereo analysis offers gradient data, see Tab. 5.3, which can be used for computing a relative depth map. The computed angles were close to 90˚, i.e. nearly accurate.

	static stereo						**dynamic stereo**		
	intensity based			feature based					
	X	Y	Z	X	Y	Z	X	Y	Z
$\mathbf{P}_4$	-45.3	104.8	169.4	-41.7	110.1	171.6	-116.3	87.7	168.7
$\mathbf{P}_6$	-54.5	58.1	173.5	-59.8	56.0	171.5	-124.6	26.3	167.4
$\mathbf{P}_7$	3.9	17.1	167.1	3.6	9.8	165.2	-34.3	13.2	167.1
$\mathbf{P}_5$	23.5	67.8	166.4	20.3	64.3	165.5	-29.1	74.6	166.0

Table 6.1: Reconstructed 3-D world coordinates (in mm) of the box in the PACKAGE scenes.

distance between		edge length (ground truth)	**static stereo**		**dynamic stereo**
			intensity based	feature based	
$\mathbf{P}_1$	$\mathbf{P}_4$	167.0	169.4	171.6	168.7
$\mathbf{P}_2$	$\mathbf{P}_5$	167.0	166.4	165.5	166.0
$\mathbf{P}_3$	$\mathbf{P}_7$	167.0	167.1	165.2	167.1
$\mathbf{P}_4$	$\mathbf{P}_6$	63.0	56.5	57.1	62.0
$\mathbf{P}_5$	$\mathbf{P}_4$	82.0*	75.4	77.3	88.2
$\mathbf{P}_6$	$\mathbf{P}_7$	82.0*	71.7	78.7	91.3
$\mathbf{P}_7$	$\mathbf{P}_5$	63.0	54.4	57.0	61.6

*The edge detection algorithm did not find the real positions of the vertices $\mathbf{P}_4$-$\mathbf{P}_7$ because their contrast to the background is too low. In this case the edges of the white label were chosen.

Table 6.2: The comparison of the 3-D point differences (in mm) with the ideal size of the box in the PACKAGE scenes.

Altogether these experiments with the box show that approximate surface data can be obtained with these methodologies based on the used calibration. However, for static and dynamic stereo analysis the correspondence problem for these measurements was solved interactively by selecting a few corresponding point pairs. Also, a better quality of these reconstruction results seems to be impossible because of the camera-object geometry (size of discrete pixels, distance between objects and camera, size of objects etc.).

The reconstruction of surface patches was possible by photometric stero analysis. The Beethoven plaster statue could be recognized at reasonable quality, cp. Fig. 5.9. This reconstruction process is also very fast and in on-line only restricted by the time of using three different light source illuminations. The reconstruction of surface patches using static stereo analysis is possible at a very rough level, and dynamic stereo analysis absolutely fails. Dynamic stereo analysis can only be suggested to get some information that there exists something in 3-D space in a certain distance. The discrete measurements of 3-D points using static or dynamic stereo analysis could be used for scaling the results of photometric stereo analysis.

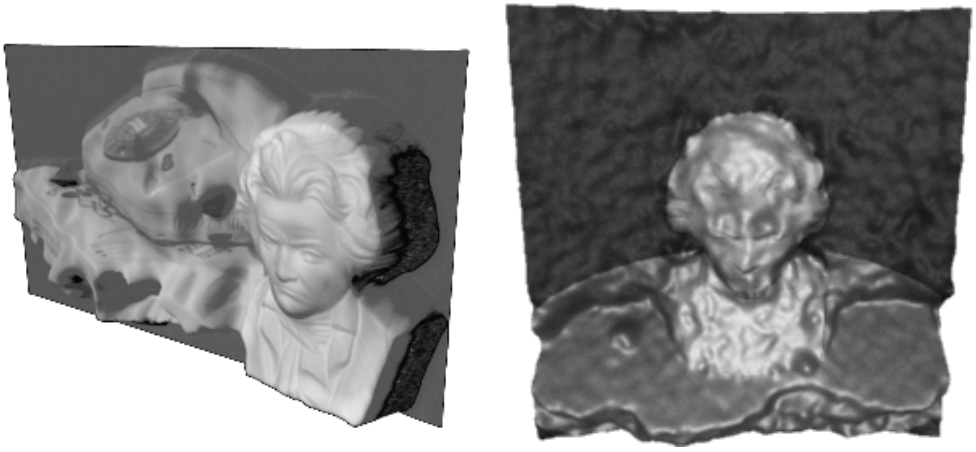


Figure 6.1: Visualization of the reconstructed surface of the Beethoven plaster statue using texture mapping: results of static stereo analysis (left) and of dynamic stereo analysis (right), cp. Fig. 5.9 for photometric stereo analysis.

The evaluation of surface patch reconstruction was performed based on the qualitative (visual) appearance of reconstructed surfaces. Of course this strongly depends upon the used method for surface representation. For example, in general texture mapping suggests better reconstruction results than a floating horizon representation. However, in the case of dynamic stereo also texture mapping did not help to suggest a reasonable reconstruction, see Fig. 6.1. For static stereo analysis some improvements seem to be possible by analysing corresponding

segments in the images [15] instead of corresponding points or line segments (edges). However, this could not yet verified in practical experiments and is a theoretical proposal so far.

For studying surface reconstruction based on structured lightening [9, 30, 44] a few experiments were already performed in our group. For example, a plaster statue was placed on a rotating disc hit by a light plane (generated with a point laser). During rotation of the object on the disc, a sequence of images is taken. Thresholding, thinning, curve approximation and triangulation allow the calculation of isolated points on the object surface. These points can be used for computing a certain surface model, e.g. by triangulation. This approach leads to very accurate surface data. However, e.g. not each object can be placed on a disc, or viewed under constant rotation.

All the studied methodologies have to be related to such applicational situations (what objects, what object motion, what illumination etc.) where they can be used with some benefit. Such proposals should be based on a very detailed performance analysis.

Acknowledgments

The first author thanks Chi-Ping Tsang and Louise Bolitho, University of Western Australia, for friendly support during his stay at Perth. Parts of this work (project PARVIS - parallel computation of stereo correspondence algorithms) were funded by the Deutsche Forschungsgemeinschaft (DFG).

REFERENCES

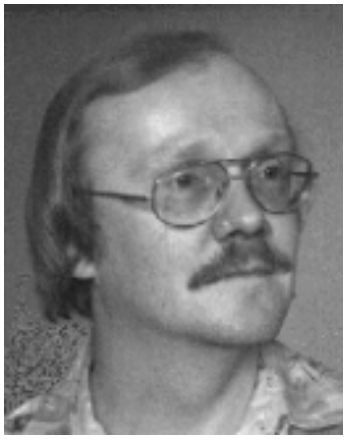
1. Barron, J.L., D.J. Fleet, S.S. Beauchemin: Performance of optical flow techniques. Int. J. Computer Vision **12** (1994), pp. 43 - 77.
2. Bellaire, G., K. Schlüns, A. Mitritz, K. Gwinner: Adaptive matching using object models generated from photometric stereo images. Proc. 8th Int. Conf. on Image Analysis and Processing, San Remo, Italy, Sept. 13 - 15, 1995, to appear.
3. Chen, J.-S., G. Medioni: Parallel multiscale stereo matching using adaptive smoothing. Proc. 1st ECCV, Antibes, France (1990), pp. 99 - 103.
4. Drew, M.S., L.L. Kontsevich: Closed-form attitude determination under spectrally varying illumination. Proc. CVPR 94, Seattle, Washington, USA, June 21 - 23, 1994, pp. 985 - 990.
5. Frankot, R.T., R. Chellappa: A method for enforcing integrability in shape from shading algorithms. IEEE Trans. on PAMI **10** (1988), pp. 439 - 451.

6. Jaisimha, M.Y., R.M. Haralick, D. Dori: Quantitative performance evaluation of thinning algorithms in the presence of noise. in: C. Arcelli, L.P. Cordella, G. Sanniti di Baja (eds.), *Aspects of Visual Form Processing*. World Scientific, Singapore 1994, pp. 261 - 286.
7. Horn, B.K.P.: Understanding image intensities. Artificial Intelligence **8** (1977), pp. 201 - 231.
8. Horn, B.K.P.: *Robot Vision*. McGraw-Hill, New York 1986.
9. Jarvis, R.: Range sensing for computer vision. in: A.K. Jain, P.J. Flynn (eds.), *Three-Dimensional Object Recognition Systems*. Elsevier, 1993, pp. 17 - 56.
10. Jordan, J.R., III, A.C. Bovik: Using chromatic information in dense stereo correspondence. Pattern Recognition **25** (1992), pp. 367 - 383.
11. Kanatani, K.: *Group-Theoretical Methods in Image Understanding*. Springer, Berlin 1990.
12. Klette, R.: A framework to computer vision research. Machine Graphics & Vision **1** (1992), pp. 331 - 341.
13. Klette, R.: Integrative approaches to "shape from motion". in: R. Klette, W. G. Kropatsch (eds.), *Theoretical Foundations of Computer Vision*. Akademie Verlag, Berlin 1992, pp. 203 - 214.
14. Klette, R., V. Rodehorst: Algorithms for shape from shading, lighting direction and motion. in: D. Chetverikov, W. G. Kropatsch (eds.), *Computer Analysis of Images and Patterns*. Proc. CAIP'93, Springer, Berlin 1993, pp. 420 - 427.
15. Klette, R.: Shape from area and centroid. Proc. Int. Conf. AIICSR´94, Smolenice, Slovakia, World Scientific, Singapore 1994, pp. 309 - 314.
16. Klette, R., P. Handschack: Quantitative comparisons of differential methods for measuring of image velocity. in: C. Arcelli, L.P. Cordella, G. Sanniti di Baja (eds.), *Aspects of Visual Form Processing*. World Scientific, Singapore 1994, pp. 241 - 250.
17. Klette, R., P. Handschack: Evaluation of differential methods for image velocity measurement. Computers & Artifical Intelligence (1995), to appear.
18. Klette, R.: Surface from motion: without and with calibration". Computing (1995), to appear.
19. Klette, R., D. Mehren, V. Rodehorst: An application of shape reconstruction from rotational motion. Real-Time Imaging (1995), to appear.
20. Klinker, G.J., S.A. Shafer, T. Kanade: A physical approach to color image understanding. IJCV **4** (1990), pp. 7 - 38.
21. Koschan, A.: Stereo matching using a new local disparity limit. in: R. Klette (ed.), *Computer Analysis of Images and Patterns*. Proc. IVth Int. Conf. CAIP'91, Dresden, September 17 - 19, 1991, pp. 48 - 53.
22. Koschan, A.: A framework for area-based and feature-based stereo vision. Machine Graphics & Vision **2** (1993), pp. 285 - 308.

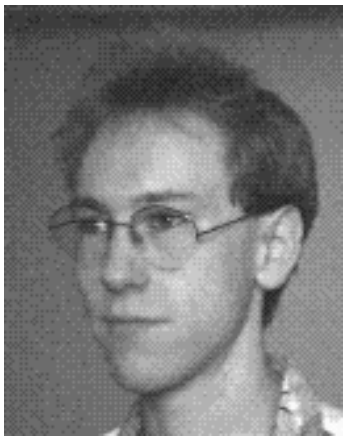
23. Koschan, A.: Chromatic block matching for dense stereo correspondence. in: S. Impedovo (ed.), *Progress in Image Analysis and Processing III*. World Scientific, Singapore 1993, pp. 641 - 648.
24. Koschan, A.: What is new in computational stereo since 1989: A survey on current stereo papers. Technical Report 93 - 22, Berlin Technical University, Computer Science Department, August 1993.
25. Koschan, A.: How to utilize color information in dense stereo matching and in edge-based stereo matching. in: *Automation, Robotics and Computer Vision*. Proc. 3rd Int. Conf. ICARCV´94, Singapore, November 8 - 11, 1994, Vol. 1, pp. 419 - 423.
26. Koschan, A., V. Rodehorst: Towards real-time stereo employing parallel algorithms for edge-based and dense stereo matching. in: *Computer Architectures for Machine Perception*. Proc. of the IEEE Workshop CAMP´95, Como, Italy, September 18 - 20, 1995, to appear.
27. Langer, M.S., S.W. Zucker: Diffuse shading, visibility fields, and the geometry of ambient light. Proc. ICCV 93, Berlin, Germany, May 11 - 14, 1993, pp. 138 - 147.
28. Maybank, S.: *Theory of Reconstruction from Image Motion*. Springer, Berlin 1993.
29. Muller, J.P., G.P. Otto, K.W. Chau, K.A. Collins, N.M. Dalton, T. Day, I.J. Dowman, D. Gugan, D. Morris, M.A. O´Neill, J.G.B. Robert, A. Stevens, M. Upton: Real-time stereo matching using transputer arrays. Proc. IGARSS 88, Edinburgh, GB, 1988, pp. 1185 - 1186.
30. McIvor, A.M.: Three-dimensional vision applied to natural surfaces. Industrial Research Limited Report 325, Auckland, October 31, 1994.
31. Nasrabadi, N.M., C.Y. Choo: Hopfield network for stereo vision correspondence. IEEE Trans. on Neural Networks **3** (1992), pp. 5 - 13.
32. Nayar, S.K., K. Ikeuchi, T. Kanade: Shape from interreflections. IJCV **6** (1991), pp. 173 - 195.
33. Nerendra, P.M.: A separable median filter for image noise smoothing. IEEE Trans. on PAMI **3** (1981), pp. 20 - 29.
34. Ohta, Y.-I., T. Kanade, T. Sakai: Color information for region segmentation. CGIP **13** (1980), pp. 222 - 241.
35. Okutomi, M., O. Yoshizaki, G. Tomita: Color stereo matching and its application to 3-D measurement of optic nerve head. Proc. 11th IAPR Int. Conf. on Pattern Recognition, The Hague, The Netherlands, vol. I, 1992, pp. 509 - 513.
36. Onn, R., A. Bruckstein: Integrability Disambiguates Surface Recovery in Two-Image Photometric Stereo. IJCV **5** (1990), pp. 105 - 113.
37. Rumpel, D., K. Schlüns: Szenenanalyse unter Berücksichtigung von Interreflexionen und Schatten. Proc. 17. DAGM-Symposium Mustererkennung 1995, Bielefeld, Germany, Sept. 13 - 15, 1995, to appear.

38. Schlüns, K.: Colourimetric stereo. in: R. Klette, W. G. Kropatsch (eds.), *Theoretical Foundations of Computer Vision*. Akademie Verlag, Berlin 1992, pp. 181 - 190.
39. Schlüns, K.: Eine Erweiterung des Photometrischen Stereo zur Analyse nicht-statischer Szenen. Proc. 14. DAGM-Symposium Mustererkennung, Dresden, Germany, Sept. 14 - 16, 1992, pp. 405 - 410.
40. Schlüns, K., O. Wittig: Photometric stereo for non-Lambertian surfaces using color information. Proc. 7th Int. Conference on Image Analysis and Processing, Monopoli, Italy, Sept. 20 - 22, 1993, pp. 505 - 512.
41. Schlüns, K.: Photometric stereo for non-Lambertian surfaces using color information. in: D. Chetverikov, W. G. Kropatsch (eds.), *Computer Analysis of Images and Patterns*. Proc. CAIP'93, Springer, Berlin 1993, pp. 444 - 451.
42. Shafer, S.A., T. Kanade, K. Ikeuchi: Image understanding at CMU. Proc. Image Understanding Workshop, Monterey, Ca., Nov. 13 - 16, 1994, Vol. I, www-online.
43. Shirai, Y., Y. Nishimoto: A stereo method using disparity histograms of multi-resolution channels. Proc. 3rd Int. Symp. on Robotics Research, Gouvieux, France, 1985, pp. 27 - 32.
44. Stahs, T., F. Wahl: Fast and versatile range data acquisition in a robot work cell. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, 1992, pp. 1169 - 1174.
45. Tremblay, M., M. Savard, D. Poussart: Medium level scene representation using a VLSI smart hexagonal sensor with multiresolution edge extraction capability and scale space integration processing. Proc. IEEE Conf. Computer Vision and Pattern Rec., Seattle, Washington, USA, 1994, pp. 632 - 637.
46. Tsai, R.Y.: An efficient and accurate camera calibration technique for 3D machine vision. Proc. IEEE Conf. Computer Vision and Pattern Rec. 1986, pp. 364 - 374.
47. Voss, K., R. Neubauer, M. Schubert: Monokulare Rekonstruktion für Robotvision. Verlag Shaker, Aachen 1995.
48. Woodham, R.J.: Photometric method for determining surface orientations from multiple images. Optical Engineering **19** (1980), pp. 139 - 144.

Reinhard Klette is professor in computer science at the Berlin Technical University, Germany. He is a co-author of the books „*Fast Algorithms and their Implementation on Specialized Parallel Computers*“ (North Holland, 1989) and „*Handbuch der Operatoren für die Bildbearbeitung*“ (Vieweg, 1992, 2nd extended ed., Vieweg, 1995, English ed., Wiley-Interscience, 1995). As chairman of symposia and conferences, he was the editor or co-editor of several proceedings. His research interests are in fundamentals and applications of 3-D computer vision.



Andreas Koschan received his Diploma and his Doctor of Engineering in Computer Science from the Technical University Berlin, Germany, in 1985 and 1991 respectively. He is currently assistant professor in the computer vision group at the Technical University Berlin. His research interests include computational color vision, 2D and 3D computer vision.



Karsten Schlüns received the M.S. degree in computer science from the Technical University Berlin, Germany, in 1991. He is a Research Assistant at the Technical Computer Science Institute, Technical University Berlin, where he is teaching and working towards the Ph.D. degree in Computer Science.

He has written several papers on the combination of color and three-dimensional shape recovery. His research interests include computer vision, color vision, robotics, and computer graphics.

Volker Rodehorst obtained the degree M. Sc. of computer science at the Technical University of Berlin, Germany, in 1994. He is currently research assistent for the project PARVIS in the computer vision group at the Technical University of Berlin, where he is developing highly parallel algorithms for stereo matching. His research interests are in three-dimensional computer vision, shape reconstruction, parallel algorithms, visualization and computer graphics.