8 REFERENCES

[1] J. Rasure and M. Young. Open environment for image processing and software development. In R.B. Arps and W.K. Pratt, editors, *Image Processing and Interchange: Implementation and Systems*, volume 1659 of *SPIE–Proceedings Series*, pages 300–310, San Jose, CA, February 1992. SPIE.

[2] D.H. Ballard, C.M. Brown. *Computer Vision.* Prentice-Hall, Englewood Cliffs N.J., 1982.

[3] O. D. Faugeras and M. Hebert. The Representation, Recognition, and Locating of 3-D Objects. *The International Journal of Robotics Research*, 5(3):27–53, 1986.

[4] O. D. Faugeras, F. Lustman, and G. Toscani. Motion and Structure from Motion from Point and Line Matching. *IEEE International Conference on Computer Vision*, pages 25–34, 1987.

[5] J. Gettys and R. Scheifler. The X–Window System. *ACM TOG*, 5(2):79–109, April 1986.

[6] C.G. Harris, M.J. Stephens. A combined corner and edge detector. In *Proc. of the 4th Alvey Vision Conference*, 1988.

[7] A. Hildebrand. SMART: Die Umsetzung bestehender Objekte in abstrakte 3D-Beschreibungen. *TOP-Zeitung*, 1993.

[8] A. Hildebrand and U. Köthe. SMART: System zur Segmentierung, Matching und 3D-Rekonstruktion. *erscheint in: Elektronik Journal, Zeitschrift für industrielle Elektronik*, 28(2 or 3), 1993.

[9] A. Hildebrand, R. Hofmann. Studie über Systeme für die Bauaufnahme. Projektabschlußbericht FAGD–90i014, Fraunhofer–Arbeitsgruppe für graphische Datenverarbeitung, Darmstadt, 1990.

[10] A. Hildebrand, U. Köthe, W. Luth, U. Mönninghoff. SMART: ein Segmentierungs-, Matching, 3D-Rekonstruktionssystem. *Computer Graphik topics*, 3(1), 1992.

[11] B. K. P. Horn. *Robot Vision.* MIT Press, 1986.

[12] K. Kanatani. *Group Theoretical Methods in Image Understanding.* Springer, Berlin, 1990.

[13] C.A. Kohl, A.R. Hanson, E.M. Riseman. Goal-directed control of low-level processes for image interpretation. In *Proc. of the IUW*, 1987.

[14] K. Kraus. *Photogrammetrie, Theorie und Praxis der Auswertesysteme.* Dümmler, Bonn, 1984.

[15] H.C. Longuet-Higgins. A Computer Algorithm for Reconstructing a Scene from Two Projection. *Nature*, pages 133–135, 1981.

[16] D. Marr. *Vision.* Freeman, San Francisco, 1982.

[17] P.J. Mercurio. Khoros. *Pixel*, pages 28–33, March/April 1992.

[18] R. Mohan, R. Nevatia. Perceptual Organization for Szene Segmentation and Description. *IEEE-PAMI*, 14(6), 1992.

[19] H. H. Nagel, editor. *Image Sequences - Ten (octal) Years From Phenomenology towards a Theorectical Foundation*, Paris, october 1986. ICPR.

[20] J. Pauli, B. Radig, A. Blömer C.-E. Liedke. Integrierte, adaptive Bildanalyse. Technical Report TUM–I9204, Tech. Univ. München, Institut für Informatik, 1992.

[21] E. Saund. Putting knowledge into a visual shape representation. *Artif. Intell.*, 54, 1992.

[22] O. D. Faugeras G. Toscani. The Calibration Problem for Stereo. *IEEE International Conference on Computer Vision and Pattern Recognition CVPR*, pages 15–20, 1986.

[23] R.Y. Tsai. An Efficient and Accurate Camera Calibration Technique for 3D Machine Vision. *IEEE International Conference on Computer Vision*, pages 364–374, 1986.

[24] W. Wester-Ebbinghausen. *Einzelstandpunkt-Selbstkalibrierung ein Beitrag zur Feldkalibrierung von Aufnahmekammern.* Verlag der Bayerischen Akademie der Wissenschaften, M"unchen, 1983.

[25] B. Wrobel. *Photogrammetrie III, Skriptum zur Vorlesungsreihe.* Darmstadt, Institut für Photogrammetrie und Kartographie, 1987.

All this data can be obtained by the previous calculations. The algorithm performs a gradual improvement of the coordinates relative to the basis system and the relative camera orientation. In the course of this the cones of rays are translated and rotated to link the rays of corresponding points. If there are some control points (these are points were the coordinates in the world coordinate system are known) an additional improvement of the results is possible. The compensating calculation is based on a NEWTON-GAUSS iteration method. The algorithm supplies gradually, after each iteration improved 3D coordinates of the object.

6 REALIZATION

A main goal during the development of the SMART system was the usage of a general-purpose equipment. As a result a standard photographic camera or CCD camera and a general-purpose-workstation are used. The user-interface is developed employing X-Windows and the athena widget set [5]. The different modules were integrated in the KHOROS [1] [17] system, which is an open system for information processing and visualization. As a result adding new modules to the system can be easily done. The current stage of development of the SMART-system consists of four components:

- An image browser: the browser facilitates the choice of an image sequence which should be used to reconstruct an object. For the selection of an image sequence the underlying device is traversed. After the identification of the image sequence different views of the object can be chosen. This can be done by selecting an icon which represents a specific view of the object.
- A module to perform the correspondence of significant homologic points in a pair of images. The correspondence problem is solved by graphical–interactive means. Two methods can be used: The user can select corresponding points by marking the related pixels or confirming the points which are suggested by the system (the development of a user interface to support the second case is currently in work). In the second case the system performs the segmentation and matching algorithms which are described in the sections (3) and (4).
- The algorithm for reconstructing the 3D data. The method is a combination of a stereo reconstruction followed by a module which connects the parts of the reconstructed data, and an additional module to improve the quality of the results.
- A module to visualize the reconstructed object.

The internal data structure of the SMART system is application independent, thus a connection to an existing modelling or CAD system can be easily performed. Due to the modular structure of the system the extension and modification of the functions are very simple. The goal of the current work is the gradual automation of the system. Another objective is the improvement of the accuracy of the reconstruction algorithm using real data. We made some tests to measure the accuracy with both synthetic and real data. For the tests with synthetic data, we rounded the measured points to receive a precision of three digits which corresponds to the number of digits in real applications (by using CCD-cameras). We could find out a correlation between the number of points and the quality of the received results. In addition all the results could be improved by the bundle block adjustment. Comparing the results received by synthetic data with the results coming from real data we established a worsening up to a factor of six. We put this factor down to the cause of the inaccurate determination of the intrinsic coordinates. To improve these results there are two things we want to do. First we want to replace the algorithm we described in 2.1 by the techniques described in [23] or [24]. With these methods it is possible to determine the intrinsic parameter without knowing the 3D coordinates of any point of the object. Second we want to include subpixel coordinates to increase the number of digits.

7 ACKNOWLEDGEMENTS

The authors would like to thank U. Mönninghoff for their valuable contributions to the presented work and Dr. W. Luth and P. J. Neugebauer for fruitful discussions and suggestions.

the solution can be found by using quaternions [3]. After the determination of R and t the coordinates of the 3D points can be expressed relative to the camera coordinate system K_1 :

$$Z_{k1} = \frac{({R_1}^T - x_{b2}{R_3}^T)t}{({R_1}^T - x_{b2}{R_3}^T)T_1)} \tag{27}$$

$$X_{k1} = x_{b1}Z_{k1} \tag{28}$$

$$Y_{k1} = y_{b1}Z_{k1} \tag{29}$$

R_i and T_i are equal to the i.th row vector of the matrixes.

5.2 Combining the reconstructed parts of the object

The technique described in the previous section supplies only a part of the whole object. This part is set by the common points of the two images which are used for the reconstruction. To obtain the full geometrical description of the object the different parts of the stereo reconstruction must be joined. (see figure (8)). All the pairs of images which have common points were used to reconstruct a part of the object. During

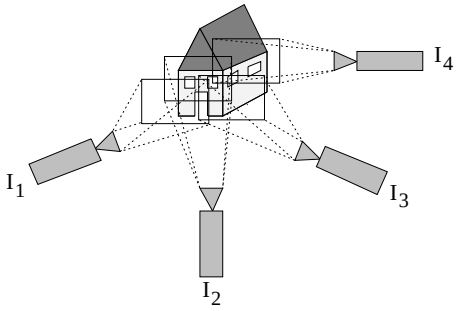

Fig. 8: Combining the reconstructed parts of the object

the stereo reconstruction which is described in the previous section, the relative orientation between every pair of images is obtained. Afterwards the 3D coordinates are determined relative to one of the two camera coordinate systems. If there is a direct or an indirect connection between all the camera coordinate systems the 3D data can be represented in a common coordinate system. In the algorithm, described above the translation vector $\vec{t}$ was normalized. Corresponding to the real distance between the camera coordinate systems the reconstructed 3D data is received in different scales. If there are common points in two stereo pairs, the distance between these points can be used to relate the scale of the first stereo pair to the scale of the second stereo pair. Doing this gradually with all the stereo pairs of the image sequence the reconstructed 3D data can be obtained in one common coordinate system, due to the evaluation of the extrinsic parameters (see section 2.1) the coordinates can be related to the world coordinate system.

5.3 Bundle block adjustment

Due to the limited resolution of the image and the transformations into a common coordinate system, the object coordinates which are obtained by the methods described in the previous sections are inaccurate. To increase the quality of the results we use the so called *bundle block adjustment* [25] [14]. Within this method a direct connection between the object coordinates and the camera coordinates is established. The connection of the points of an image plane and the corresponding 3D points in the camera coordinate system define a cone of rays. The extrinsic orientation of these cones of rays, i.e. the relative orientation to a basis coordinate system, is determined simultaneous for all the images of the image sequence. Apart from the coordinates of the points concerning the different camera coordinate systems, a rough estimation of the coordinates relative to the basis coordinate system and the relative camera orientation are necessary.

$$\vec{w_1}\,(\vec{t}\times R\vec{w_2}) = 0 \tag{16}$$

Introducing the antisymmetric matrix T:

$$T = \begin{pmatrix} 0 & -t_3 & t_2 \\ t_3 & 0 & -t_1 \\ -t_2 & t_1 & 0 \end{pmatrix} \tag{17}$$

we can describe the transformation of a point P, whose coordinates are given in the coordinate system K_1 by an expression relative to the coordinate system K_2. The transformation is given by the matrix E:

$$E = T\,R \tag{18}$$

Due to the coplanarity of $\vec{w_1}$, $\vec{w_2}$ and $\vec{t}$ the following equation holds for the vectors ${\vec{w_1}}^T = (x_{b1}, y_{b1}, 1)$ and ${\vec{w_2}}^T = (x_{b2}, y_{b2}, 1)$:

$${\vec{w_1}}^T E\vec{w_2} = 0 \tag{19}$$

Expanding the equation (19) we receive the equation (20) for each pair of points (x_{b1}, y_{b1}) and (x_{b2}, y_{b2}):

$$\begin{aligned} x_{b1}x_{b2}e_{11} + x_{b1}y_{b2}e_{12} + x_{b1}e_{13} + \\ y_{b1}x_{b2}e_{21} + y_{b1}y_{b2}e_{22} + y_{b1}e_{23} + \\ x_{b2}e_{31} + y_{b2}e_{32} + e_{33} = 0 \end{aligned} \tag{20}$$

By separating the points in the image planes:

$$P_i = (x_{b1i}x_{b2i}, x_{b1i}y_{b2i}, x_{b1i}, y_{b1i}x_{b2i}, y_{b1i}y_{b2i}, y_{b1i}, x_{b2i}, y_{b2i}, 1) \tag{21}$$

from the parameter of the matrix E:

$$X = (e_{11}, e_{12}, e_{13}, e_{21}, e_{22}, e_{23}, e_{31}, e_{32}, e_{33}) \tag{22}$$

and if we use $||\vec{t}|| = 1$ respectively $||T^2|| = 2$ or $||E^2|| = 2$, as an additional constraint we can determine the unknown of the matrix E by using eight equations:

$$P_n X = 0 \tag{23}$$

If the rank of the matrix $P_n = 8$ the equation system can be solved directly. On the other hand the determination of the pixel coordinates in digital images causes some error, thus the usage of a least square method in combination with a large number of points will yield better results. As a consequence the following minimization problem can be defined:

$$\min_X ||P_n X|| \tag{24}$$

the solution is the eigenvector of matrix ${P_n}^T P_n$ of norm $\sqrt{2}$ corresponding to the smallest eigenvalue. For the determination of the matrix R and T matrix E has to be reduced [19]. Using the constraints $t^t E = t^t TR = 0$ and for unambiguous reasons $||t||^2 = 1$ the translation vector $\vec{t}$ is given by the solution of the following minimization problem, which can be solved by using least square techniques:

$$\min_t ||E^t t||^2 \tag{25}$$

the solution is the eigenvector of matrix EE^t corresponding to the smallest eigenvalue. Furthermore we have to estimate the rotation matrix R. Again the solution can be obtained by solving a minimization problem:

$$\min_R ||E - TR|| \tag{26}$$

Beside the transformation parameters we receive the 3D camera coordinates as additional unknowns Z_{k1} and Z_{k2}. Each pair of points provides three additional equations and two unknowns. In other words each pair of points provides only one constraint. There are two characteristics which make the equations harder to solve. First the equations are nonlinear and second the relative orientation is only defined up to a scale factor. In practical terms the second characteristics causes: scaling the triangle K_1K_2P consisting of the vertices of the two camera positions and an object point, does not influence the position of this point in the image planes (see left side of figure (7)). To receive an unambiguous solution the length of the translation vector $\vec{t}$ should be fixed (i.e. $\vec{t} \cdot \vec{t}^T = 1$). Enforcing the orthonormality of the rotation matrix

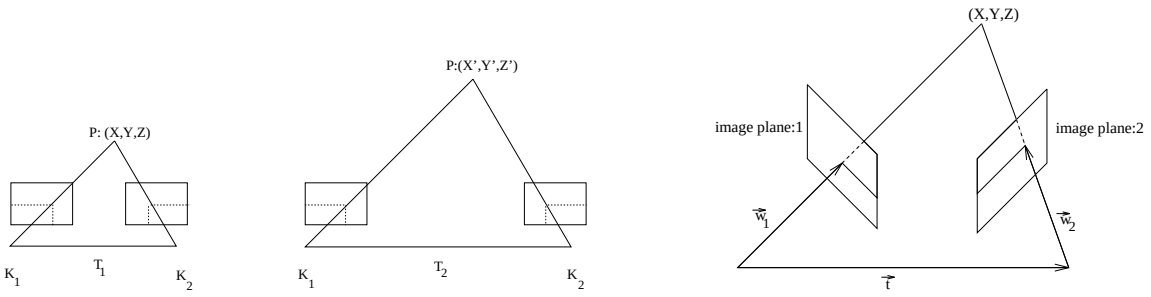

Fig. 7: Left: Scaling the triangle K_1K_2P, Right: Coplanarity of the vector $\vec{w_1}, \vec{w_2}$ and $\vec{t}$

($RR^{-1} = E$) there are another six constraints which can be incorporated:

$$\begin{aligned} r_{11}^2 + r_{12}^2 + r_{13}^2 &= 1 \\ r_{21}^2 + r_{22}^2 + r_{23}^2 &= 1 \\ r_{31}^2 + r_{32}^2 + r_{33}^2 &= 1 \\ r_{11}r_{21} + r_{12}r_{22} + r_{13}r_{23} &= 0 \\ r_{21}r_{31} + r_{22}r_{32} + r_{23}r_{33} &= 0 \\ r_{31}r_{11} + r_{32}r_{12} + r_{33}r_{13} &= 0 \end{aligned} \tag{14}$$

Given n pairs of points and introducing the constraints (14) we have $12 + 2n$ unknowns and $7 + 3n$ constraints. Thus the relative orientation can be calculated, if we have five point pairs in the image planes B_1 and B_2. After the unknowns Z_{k1} and Z_{k2} have been determined, we can calculate the coordinates X and Y in relation to both camera coordinate systems:

$$\begin{aligned} X_{k1} &= (x_{b1}Z_{k1})/f \\ Y_{k1} &= (y_{b1}Z_{k1})/f \\ X_{k2} &= (x_{b2}Z_{k2})/f \\ Y_{k2} &= (y_{b2}Z_{k2})/f \end{aligned} \tag{15}$$

In real applications the results can be improved by using more than five points to determine the relative orientation. A solution of the nonlinear equation system can only be found by iterative methods. Longuet-Higgins [15] describes a method to compute the relative orientation of the two projections by solving linear equations using a set of eight point pairs in the image plane. In digital images there is a limitation of the accuracy which can be obtained determining the coordinates of these points. This restriction is caused by the spatial resolution of the image and noise which results from the acquisition process. Faugeras, Lustman and Toscani [4] generalized the ideas of Longuet-Higgins to improve the robustness of the technique.

They substituted the direct solution of the matrix which represents the relative orientation by a least-square technique. We decided to use this method, because of the reduced sensitivity to errors. The basis of this method is the geometrical relation between the vectors $\vec{w_1}$ $\vec{w_2}$ describing corresponding points in the two image planes and the translation vector $\vec{t}$, which links the two camera positions K_1 and K_2. These vectors are coplanar (see right side of figure (7)) and therefore their determinant is zero:

2. the position of the point relative to the neighboring regions, and
3. the position of the point relative to (resp. on) the epipoles of the characteristic points of a corresponding region in another image.

The already known region correspondence allows the definition of a "reference image" for each region where the contour can be measured best due to good contrast and more or less perpendicular viewing. Starting with this reference image we can improve the precision of position measurement in the less favorable views.

The result of all these efforts is a list of corresponding points which serves as input for the exact 3D reconstruction, and a data-structure where the connections of the points with lines to polygons and the grouping of the polygons to planes and objects are stored.

5 RECONSTRUCTION OF THE 3D GEOMETRY

5.1 Relative camera orientation

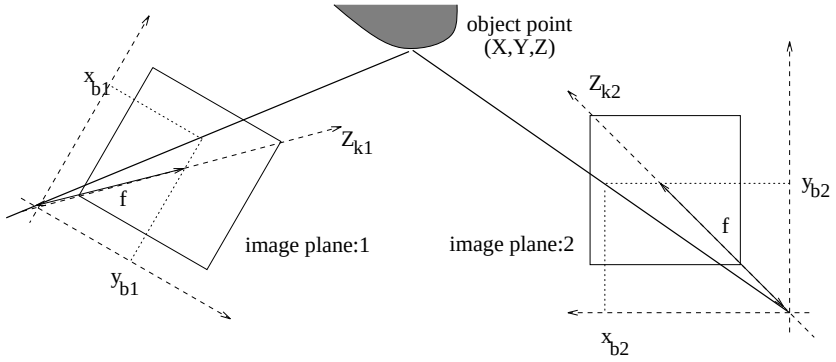

Fig. 6: Arbitrary arrangement of two camera systems

The foundation of the SMART-system is a sequence of single shots of the object. Before the 3D coordinates can be determined the relative orientation between the camera coordinate systems must be evaluated.

The basis of the procedure are two views of a motionless scene (see figure (6)). The transformation of one coordinate system to the other, can be represented by a rotation followed by a translation. Consequently there are twelve unknown parameters which should be determined (9 concerning the rotation $r_{11}, r_{12}, .., r_{33}$ and 3 for the translation t_1, t_2, t_3).

Let $P_1 = (X_{k1}, Y_{k1}, Z_{k1})$ be the coordinates of a point relative to the coordinate system K_1 and let $P_2 = (X_{k2}, Y_{k2}, Z_{k2})$ be a point in the coordinate system K_2. The related points in the image plane, after performing the perspective projection, are the points (x_{b1}, y_{b1}) and (x_{b2}, y_{b2}) in the image planes B_1 and B_2 (The coordinates of (x_{b1}, y_{b1}) and (x_{b2}, y_{b2}) characterize the calibrated image coordinates derived from the methods which are described in section 2.1). Considering the intrinsic orientation of the camera and assuming a focal length f we have:

$$\frac{x_{b1}}{f} = \frac{X_{k1}}{Z_{k1}} \qquad \frac{y_{b1}}{f} = \frac{Y_{k1}}{Z_{k1}}$$

the pair (x_{b2}, y_{b2}) is determined analogous. Due to the relative orientation of the two camera coordinate systems we have three equations for every pair of corresponding points:

$$\begin{aligned} r_{11}x_{b1} + r_{12}y_{b1} + r_{13}f + t_1\frac{f}{Z_{k1}} &= x_{b2}\frac{Z_{k2}}{Z_{k1}} \\ r_{21}x_{b1} + r_{22}y_{b1} + r_{23}f + t_2\frac{f}{Z_{k1}} &= y_{b2}\frac{Z_{k2}}{Z_{k1}} \\ r_{31}x_{b1} + r_{32}y_{b1} + r_{33}f + t_3\frac{f}{Z_{k1}} &= f\frac{Z_{k2}}{Z_{k1}} \end{aligned} \tag{13}$$

$$y_s = \frac{1}{F}\sum_g \frac{y}{(x^2+y^2+f^2)^2}, \tag{9}$$

where f is the focus length of the camera and g the region under consideration.

Finally we have to find the correspondence between the images of a series. We prefer region matching to contour matching because (1) regions have more attributes that help to reduce the search complexity of the matching, and (2) many region attributes are integral features over the whole area of the region and thus less sensitive to noise and perspective distortion. The following criteria are applied to find the corresponding regions:

1. The optical flow of rigid objects is a smooth function.
2. The neighborhood of the regions has to be preserved.
3. The attributes of corresponding regions must be similar.

The similarity of the regions is currently measured by the following cost-function:

$$\text{cost}(i \equiv j) = \left(\frac{\text{Area}(\text{XOR}(\text{Trans}_j(i), j))}{\text{Area}(i)+\text{Area}(j)}\right)^2 + \left(\frac{I_i - I_j}{I_{max} - I_{min}}\right)^2, \tag{10}$$

where i and j are candidate regions in two different images, $\text{XOR}(i,j)$ extracts the non-overlapping parts of those regions, $\text{Area}(k)$ returns the area of a region k, and $\text{Trans}_j(i)$ translates the region i so that its centroid coincides with that of region j. I_i and I_j are the mean grey levels of regions i and j, I_{max} and I_{min} are respectively the maximum and minimum grey levels.

The neighborhood between regions is represented in a neighborhood matrix N. If for example (i_1, j_1) is the matching with global lowest cost, we try to match the neighbors of region i_1 in the one image onto the neighbors of region j_1 in the other, in such a way that their mutual neighborhood is also preserved and the costs of the matching are minimized. If this is impossible, the matching (i_1, j_1) was incorrect and we try another possible relationship. Otherwise we continue with the correspondence that has the next lowest costs. In that manner the list of corresponding regions grows from already matched regions to the not matched ones.

4 EXACT SEGMENTATION AND POINT MATCHING

In this step we return to the original images and segment them again using all information obtained until now. As mentioned already this allows us to modify the strategy of the segmentation. Instead of applying operators that find regions or edges without any background information we can now search for the exact borders of the regions whose positions are approximately known. That the borders must lie in the *contour region* which enclosed the *non-contour regions* under consideration follows by definition. Consequently, we have a clearly defined region of interest (ROI) where the image can be optimized locally for further processing and where the parameters of image processing operators can be chosen appropriately.

After optimizing the image properties within each region of interest we detect corners and crossing points by means of the *corner response function* (Harris and Stephens [6]). These significant points are now connected by minimizing the following cost function (Ballard and Brown [2]):

$$\text{cost} = \text{length(path(k,l))} - \sum_{\text{path(k,l)}} \text{grad}, \tag{11}$$

where

$$\text{grad} = \frac{1}{\text{grad}_{max}}\sqrt{\left(\frac{\partial I}{\partial x}\right)^2 + \left(\frac{\partial I}{\partial y}\right)^2} \tag{12}$$

is the normalized absolute gradient of the grey levels I in the ROI, $\text{path}(k,l)$ the path between the points k and l and $\text{length}(\text{path}(k,l))$ the length of that path. grad_{max} is the maximum gradient in the image. Fig. (5) right shows the exact position of the contour in a detail of Fig. (2).

Afterwards the detected points are matched using three criteria:

1. the position of the point relative to the principle axes of the region it belongs to,

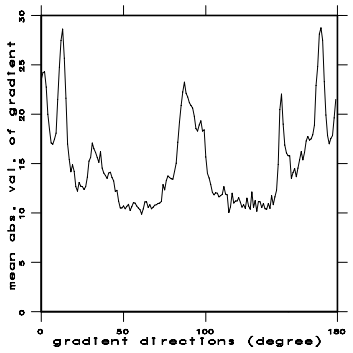

Fig. 4: Mean absolute value of the gradient vs. gradient direction for fig. 2

defined as bands of about 10 pixel width rather than as one pixel wide lines. This has some advantages too:

- Contour gaps are relatively unlikely to occur.
- The image is clearly classified into contour regions and non-contour regions.
- Both kinds of regions can be used as a mask for one of the following processing steps.

Fig. (5) left shows the result of the described process. Note that the topologic relations between the different regions are reflected very well.

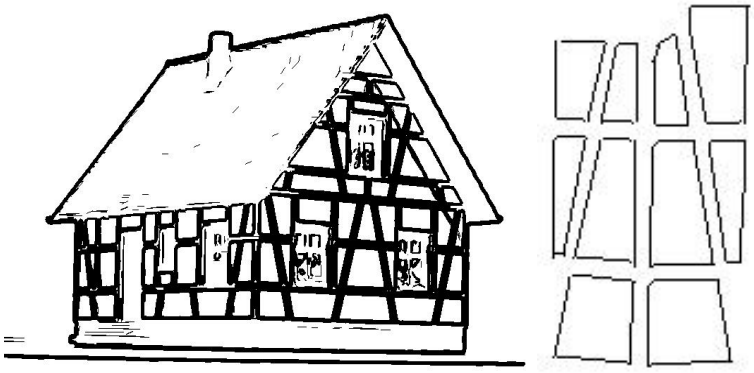

Fig. 5: Left: Rough segmentation. Right: Exact segmented detail (front edges of the house)

Of course the *main direction criterion* is appropriate only for a limited range of applications. But because of the modular design of the SMART system this heuristic may easily be replaced or extended by other heuristics.

A first labelling is now obtained by attaching an unambiguous number to each connected region. Additionally each region is characterized by some attributes like the mean grey level within the region, the standard deviation of the grey level distribution, the centroid, and the principal axes. We prefer the application of perspective invariant attributes as stated for example by Kanatani [12]. The invariant centroid (x_s, y_s) is defined as:

$$F = \sum_g \frac{1}{(x^2+y^2+f^2)^2}, \tag{7}$$

$$x_s = \frac{1}{F}\sum_g \frac{x}{(x^2+y^2+f^2)^2}, \tag{8}$$

Using the results of equation (6) in equation (5) we get the transformation matrix which transforms a point of the world coordinate system to a point of the image plane.

On the basis of a series of 3D points intrinsic to the scene and the corresponding 2D points of the image plane the 12 unknown coefficients of the transformation matrix can be determined. Each pair of points provides two equations, thus six non coplanar 3D points and the related 2D points of the image plane are necessary to solve the equation system. To increase the robustness of the algorithm the usage of more than six points in combination with a least square method is to be favored. Once the coefficients of the matrix have been determined, they can be decomposed into the parameters of the intrinsic and the extrinsic orientation [22].

3 ROUGH SEGMENTATION AND REGION MATCHING

This first step of the vision pipeline is designed to segment all images of an image series into labelled regions in such a way that the projections of each object region have the same label in all images of the series. The main emphasis in this step is on the correct distinction between characteristic and non-characteristic image primitives and on the correct reflection of the topological relations between those primitives. The exact positions of the primitives are less important at the moment.

We start the segmentation process with common techniques for the creation of a *primal sketch* like edge filtering, corner detection, and texture analysis. All of these techniques have a lot of parameters that influence the quality of the obtained primal sketch. The adaptive control of those parameters is an essential requirement for the automatic image segmentation. Often expert systems are used for parameter control, as described e.g. by Pauli et al. [20] or Kohl et al. [13]. But despite of a global optimal parameter setting the results are usually not satisfactory over the entire image, because the characteristics of the image normally differ to much.

So we could try to adapt the parameters locally. To do that one has to choose an appropriate region for the calculation of local image characteristics. This requires basically that the segmentation has already been done. Additionally correlated noise may cause pseudo edges or regions. Hence some characteristic features are harder to spot than disturbances. In such cases the segmentation will remain unsatisfying despite an optimal local adaptation of the parameters.

Recent publications (e.g. Mohan and Nevatia [18], Saund [21]) show that these difficulties can be overcome by means of *perceptual organization* of the data. In this method the image primitives obtained by standard procedures are grouped into a hierarchy of primitives with increasing semantic level on the basis of an appropriate heuristic (Marr [16]). So it is possible to correct or remove defective primitives and to add missing ones. The crucial question is to find the appropriate heuristic. Mohan and Nevatia [18] suggest the following criteria:

- Primitives that are close together tend to be grouped together.
- Elements that lie along a common line or a smooth curve are grouped together.
- Symmetric curves are grouped together.
- Curves are connected to enclose regions.

On the basis of these criteria they obtain very good segmentation results, but with quite high computational costs.

Because in the current phase our system is mainly designed to deal with man-made environments we can use another characteristic property of artificial objects: They contain a lot of parallel straight lines that tend to have the same direction after perspective projection (namely the direction to the vanishing points) if the perspective distortion is not too big. Figure (4) shows characteristic maxima of the mean absolute value of the gradient versus the gradient directions. Thus we can replace the above mentioned complex *symmetry criterion* with this *main direction criterion.* Now we are able to evaluate our heuristic very efficiently by means of a combination of a gradient method and morphologic operations. As structuring elements we apply lines in those main directions thus erasing undesirable line segments by means of morphologic erosion and connecting appropriate segments via morphologic dilation. However, the information about the correct position of the edges is lost during this grouping process. Therefore the so obtained *contour-regions* are

- shearing, due to unequal scaling in the image axes

assuming the orthogonality of the sensor the composition of the matrix coefficients can be simplified as it is described below. The camera coordinate system is specified by the parameter O_k, X_k, Y_k, Z_k where O_k is the optic center of the camera and X_k, Y_k, Z_k are the vectors which define the orientation of the coordinate system. The image plane is described by the parameters o_b, x_b, y_b where o_b is the origin and x_b, y_b are the vectors which are fixing the orientation of the image plane. In addition to the focal length f, the perspective transformation can be described with the following equation (1):

$$\frac{f}{Z_k} = \frac{x_b}{X_k} = \frac{y_b}{Y_k} \tag{1}$$

Furthermore, the unit of the x- and the y-axes of the image plane is changed by scanning and sampling:

$$x_b = \frac{u}{s_u} \qquad\qquad y_b = \frac{v}{s_v} \tag{2}$$

Moreover, the image coordinate system is translated during the computer acquisition:

$$u = u - u_0 \tag{3}$$

$$v = v - v_0 \tag{4}$$

Consequently, the acquisition transformation can now be described (up to the scale factor s, which depends on the Z coordinate of the actual point) with the following matrix ($a_u = s_u f$ and $a_v = s_v f$):

$$\begin{pmatrix} s\,u \\ s\,v \\ s \end{pmatrix} = \begin{pmatrix} a_u & 0 & u_0 & 0 \\ 0 & a_v & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} X_k \\ Y_k \\ Z_k \\ 1 \end{pmatrix} \tag{5}$$

The parameters to be calibrated are the optic center coordinates u_0, v_0 (where the optical axes pierces the image plane), the scale factors of the image plane s_x , s_y and the focal length f. The extrinsic parameters

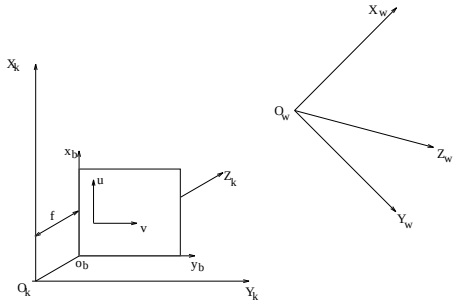

Fig. 3: Relation between image-, camera- and world coordinate system

are representing the relative orientation of the camera coordinate system and the world coordinate system. This relationship can be expressed with two affine transformations. At first we rotate the world coordinate system X_w, Y_w, Z_w to receive the orientation of the camera coordinate system.

Then the origin of the world coordinate system is translated to the origin of the camera coordinate system:

$$\begin{pmatrix} X_k \\ Y_k \\ Z_k \end{pmatrix} = \begin{pmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{pmatrix} \begin{pmatrix} X_w \\ Y_w \\ Z_w \end{pmatrix} + \begin{pmatrix} t_1 \\ t_2 \\ t_3 \end{pmatrix} \tag{6}$$

If the coefficients of the rotation matrix and the translation vector are known, the extrinsic parameters are determined.

camera positions.

In the last step *Exact 3D Reconstruction* we apply a photogrammetric reconstruction algorithm to the corresponding characteristic points. It is designed as a self-calibrating process, thus requiring no additional information about the points than just their 2D positions and their correspondence (with the exception of the unknown scale factor and the present determination of the intrinsic parameters, see section 5.1 and 2.1). On the basis of the exact 3D data of the points one may correct and extend the semantic grouping of primitives and may improve the precision of the measurement of the 2D positions. If necessary the last two procedures can be iterated until the achieved precision meets some user defined level.

The result of the reconstruction process, i.e. the primitives on different semantic levels and their 3D positions, are stored in an application independent data structure and can be used for the data exchange with for instance CAD and rendering-programs.

2 ACQUISITION OF THE IMAGE DATA

Fig. 2: Example of an image from one data set we analyzed

The image data is received from an off the shelf camera or CCD-camera. The shot positions are chosen in order to record the whole object and furthermore every point of the object is noted in at least two images. Figure (2) shows an example of an image sequence we analyzed with the SMART system.

2.1 Intrinsic and extrinsic orientation of the camera

Most of the algorithms for reconstructing the 3D geometry on the basis of 2D images presuppose idealized conditions concerning the perspective projection. These preconditions can not be achieved by common off the shelf cameras and lenses. Therefore, at first the camera specific parameters (intrinsic parameters) which affect the projection must be determined. Furthermore, there is no relationship between the camera coordinate system, where the reconstructed 3D data is described, and the world coordinate system (extrinsic orientation). Next the description of a method is following, which makes it possible to determine the intrinsic and the extrinsic parameters using six points where the 3D coordinates as well as the related image coordinates are known [22].

Ignoring the nonlinear lens distortion, the projection of the 3D points of the world coordinate system to the image plane can be represented by a linear transformation matrix. The coefficients of this matrix are set by the following influences [11]:

- scaling, due to inaccurate knowledge of the focal length f
- translation, due to movement of the origin
- rotation, due to image sensor rotation
- skewing, due to departure from orthogonality in the sensor

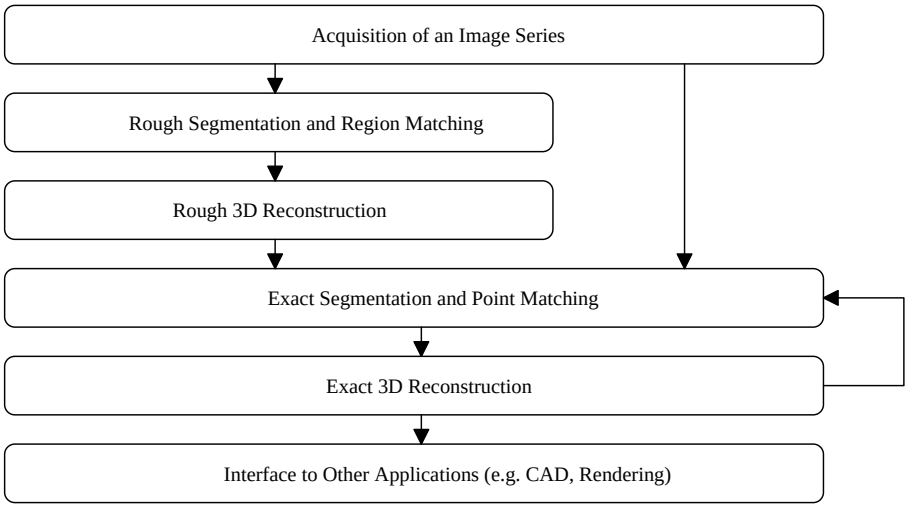

Fig. 1: The vision pipeline

non characteristic primitives and noise artefacts. This requires some heuristic that tells which primitives are the characteristic ones and how to group simple primitives (where the decision is impossible) to primitives of a higher semantic level. As the output of this step we defined an image that is segmented into *contour regions* which contain all characteristic edges and *non-contour regions* which contain no edges or only non characteristic ones. A *non-contour region* has to be surrounded completely by a *contour region*, and the topological relations of the *non-contour regions*, especially neighborhood, must be reflected correctly in the segmented image.

Consequently we are able to use the *non-contour regions* for a region based matching between the images of a series. The region based approach to the matching problem is, in general, preferable to the edge based approach, since regions have far more attributes that allow the correct identification of corresponding primitives than edges. In the next step *Rough 3D Reconstruction* we (1) determine roughly the 3D coordinates of the regions and (2) approximate the camera coordinates for each image. One can pursue two different strategies:

- For each region that is visible in at least two images we can calculate their 3D coordinates by means of the *shape from motion* method (e.g. Longuet-Higgins [15], Faugeras et al. [4]), i.e. essentially with the same algorithm that is used for the exact reconstruction.
- Under some conditions the methods of *shape from shading* and *shape from texture* allow the determination of the 3D orientation of each region and thus the calculation of a $2\frac{1}{2}$d representation of the objects in the images (e.g. Kanatani [12], Horn [11]).

Presently we apply a simplified version of the reconstruction algorithm that is applied to the exact reconstruction (see section 5.1 for more details), using the region centroids as corresponding points. Based on their 3D coordinates we can further group primitives to higher semantic levels, e.g. 3D planes and objects, which reduces again the ambiguity in the images. The camera coordinates allow the construction of epipoles which offers great help in the following step. However, the inner three dimensional shape of the regions is still unknown. Therefore the integration of *shape from shading* and *shape from texture* techniques into the SMART system is one of our next tasks.

With the information obtained, we return to the original images and segment them exactly (step: *Exact Segmentation and Matching*). We are now able to modify the segmentation problem: Instead of looking for edges (or regions) in the image, without any previous assumptions, we can search for the correct borders of each particular *non-contour region* whose positions are already known approximately. So we can locally optimize the precision of the image processing operators without losing the already achieved clarity of segmentation. In the resulting exactly segmented images we determine characteristic points and match them by means of the known matching between regions and the epipoles that may be calculated from the

SMART: System for Segmentation, Matching and Reconstruction

Axel Hildebrand[1] *and Ullrich Köthe*[2]
Fraunhofer Institute for Computer Graphics (Fraunhofer–IGD)
[1]*Department: Multimedia Systems and Image Processing, D–6100 Darmstadt*
email: hildebr@igd.fhg.de
[2]*Department: Imaging, O–2500 Rostock*
email: koethe@egd.igd.fhg.de

Abstract

In many areas of application, such as medicine, robot technology, photogrammetry [9] etc., the acquisition of an abstract description of three dimensional objects is an important task. A common approach to this problem are the photogrammetric methods. Although the basic algorithms within this field are well known, many questions are still open. Among other problems these methods require an exact determination of the camera positions before the photos can be taken. Additionally the measurement of points in the images and the combination of data from different views often has to be done by hand. Therefore a lot of skilled work and specialized equipment is necessary during both the acquisition and the evaluation of an image series.

Our approach is directed to an integrated system called SMART (**S**egmentation **M**atching **A**nd **R**econs**T**ruction) [7] [8] [10] that is based on general purpose equipment (general purpose workstation, photographic camera or CCD camera). It is designed as a self-calibrating system, i.e. the camera positions, as well as their relative orientations, are derived automatically during the evaluation of the image series. Hence the photos need not to be taken by a specially trained person.

The whole procedure within the SMART system can be reflected in a *vision pipeline* (see section). After the image acquisition we perform a rough segmentation of the images (1) to find characteristic geometric primitives of the objects and to reject non characteristic ones those occurrence is unavoidable during the acquisition process and (2) to calculate the camera positions approximately. Based on this information we measure the exact positions of the characteristic details and their correspondence. This data allows the usage of an self-calibrating reconstruction algorithm. Now, the obtained partial reconstructions are connected to one complete reconstruction. At the same time the precision of the 3D coordinates is improved by means of a bundle block adjustment [25] [14]. Since the obtained data structure is independent of a specific application area, it can easily exported to, e.g. rendering or CAD applications.

1 THE VISION PIPELINE

The basis of our system is the following pipeline we call it the *vision pipeline* to suggest an analogy to the rendering-pipeline in computer graphics (Fig. 1): In contrast to other 3D reconstruction systems we divided the pipeline into two main parts: (1) the rough treatment of the images, and (2) the exact reconstruction. We introduced a separate rough treatment step to deal effectively with noise and ambiguity in the given image series. Noise effects and ambiguity of image primitives are, in general, unavoidable during the image acquisition process and due to the projection of a 3D scene onto a 2D image. In the *Rough Segmentation* step we therefore try to find characteristic primitives in the image and we attempt to distinguish them from