Model–Based 3–D Scene Analysis from Stereoscopic Image Sequences

Reinhard Koch

Institut für Theoretische Nachrichtentechnik und Informationsverarbeitung

Universität Hannover, Appelstrasse 9A, 30167 Hannover, Germany

email: koch@tnt.uni–hannover.de

A vision–based 3–D scene analysis system is described that is capable to model complex real–world scenes like buildings automatically from stereoscopic image pairs. Input to the system is a sequence of stereoscopic images taken with two standard CCD Cameras and TV lenses. The relative orientation of both cameras to each other is known by calibration. The camera pair is then moved throughout the scene and a long sequence of closely spaced views is recorded. Each of the stereoscopic image pairs is rectified and a dense map of 3–D surface points is obtained by area correlation, object segmentation, interpolation, and triangulation. 3–D camera motion relative to the scene coordinate system is tracked directly from the image sequence which allows to fuse 3–D surface measurements from different viewpoints into a consistent 3–D model scene. The surface geometry of each scene object is approximated by a triangular surface mesh which stores the surface texture in a texture map. From the textured 3–D models, realistic looking image sequences from arbitrary view points can be synthesized using computer graphics.

Key Words: Image Processing, Virtual Reality, 3–D Scene Analysis, Stereoscopic Image Sequence Analysis, Robot Vision, Scene Reconstruction, Close Range Photogrammetry.

1 Introduction

Modeling of 3–D scenes from 2D image sequences has been a research topic for a long time as Aggarwal and Nandhakumar showed in their overview of this field (Aggarwal and Nandhakumar, 1988). The goal of such modeling is to extract a compact description of the scene for purposes of reconstruction (Blake and Zissermann, 1987), recognition (Marr and Nishihara, 1978), or data compression (Harashima and Kishino, 1991). When analyzing complex scenes with multiple moving flexible objects a complete description of all properties of the scene is necessary. In previous works the different properties 3–D object shape, 3–D object motion, and object surface texture were treated separately. Great effort went into developing algorithms that estimate 3–D object shape from various sources, termed shape from motion, stereo, texture, and others (Jarvis, 1983). On the other hand research was conducted to find solutions to the problem of rigid object motion (Netravali and Salz, 1985). Only recently the problem of nonrigid bodies and nonrigid motion was addressed (Pentland and Horowitz, 1991).

An important scene property needed for visualization is the photometric surface description. People in the field of image communication, multi media, flight and driving simulation, and virtual reality demand the construction of complete realistic environments. Sometimes it is even more important to have a good surface texture description than to obtain a refined 3–D geometry. Texture maps that store real views of the object appearance can be used for that purpose (Koch, 1990).

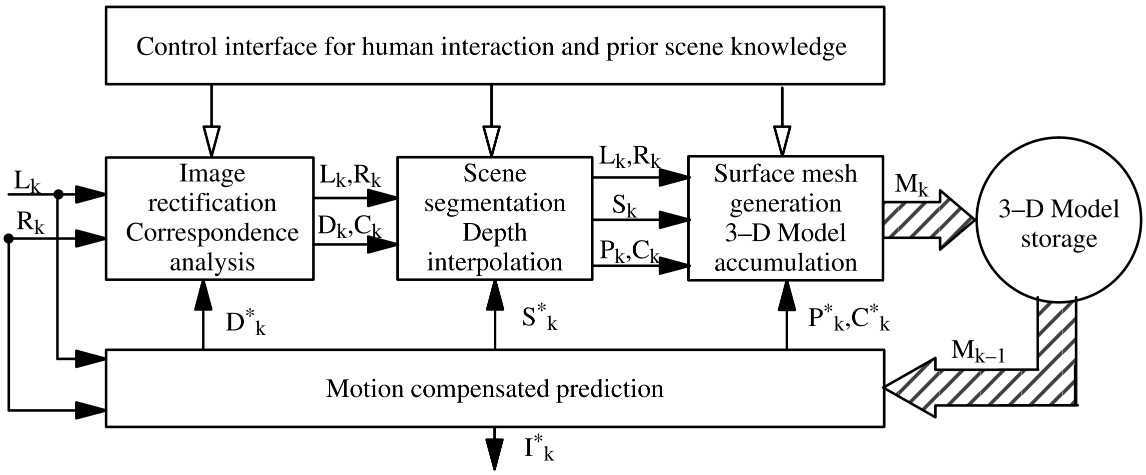
Automatic evaluation of all scene properties, camera position and 3–D object geometry as well as photometric surface mapping, for the purpose to reconstruct 3–D scene models for visualization, are discussed in this contribution. To overcome the problem of simultaneous estimation of object geometry and camera position, a calibrated stereoscopic image sequence is recorded. From each image pair the geometry is measured and from the sequence information relative camera motion can be extracted. All measurements obtained from the image sequence need then to be integrated into a consistent 3–D scene model that contains not only the scene geometry but also texture maps of the object surface. Visual simulations of the scene from this complete scene model can be performed with computer graphics.

2 Concept of 3–D Scene Analysis System

The structure of the scene analysis process is shown in Fig. 1. Four main modules (image analysis pipeline, control interface, motion compensated prediction, and 3–D model storage) can be identified. Central to the system is the image analysis pipeline that computes a model scene M_k from a stereoscopic image pair L_k, R_k at time instant k and from the accumulated sequence information contained in the model storage M_{k-1}. Sequence information is included into the analysis pipeline by motion compensated prediction at all stages. The scene model M_{k-1} is transformed from frame k–1 into the current camera position at frame k by compensation of the camera motion. From the transformed model the predictions of disparity, segmentation, and object geometry are computed and merged with the new measurements to yield a depth map of the new scene model M_k.

In order to obtain an efficient 3–D surface description and to treat hidden surfaces properly, the depth map is converted into a triangular surface mesh. In addition, the surface texture for each triangular surface patch, which represents the photometric information, is stored in M_k. From the geometric and photometric information realistic looking image sequences I^*_k can be synthesized.

The analysis pipeline is controlled by a user interface, which takes commands from the operator and supplies the analysis procedures with the proper parameters. This interface allows to insert prior scene knowledge into the analysis process. It is planned that this control interface will be replaced by a knowledge based system that automatically adapts the analysis parameters based on high level scene knowledge.



L_k: Left image
R_k: Right image
D_k: Disparity map
C_k: Confidence map
S_k: Segmentation map
P_k: Depth map
M_k: 3–D Scene model
k: Index indicating the present time instant
I^*_k: Virtual camera image from model
$^*_k$: = data predicted from model M_{k-1} using motion compensation

Fig.1: Structure of 3–D scene analysis from stereoscopic sequences.

In the following sections the procedures for the image analysis pipeline and the 3–D motion compensated prediction are explained in more detail.

3 The Image Analysis Pipeline

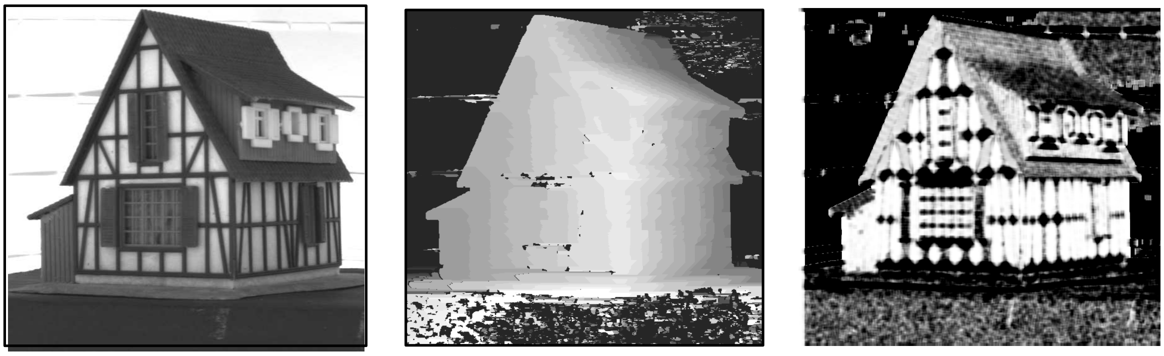
The analysis of a stereoscopic image pair is split into correspondence analysis and object generation. Correspondence analysis tries to locally estimate image plane correspondences while during object generation areas in the image that belong to physically connected regions are merged by similarity measures. Each region is interpolated to yield a dense depth map and the measurements are triangulated and transformed into object space.

3.1 Correspondence analysis

The input to the system at time instant k is a stereo pair L_k, R_k. In a preprocessing step the stereoscopic camera is calibrated and each image pair is rectified to obtain an image pair where the camera axes are parallel and both cameras are displaced along horizontal image plane coordinates only. The calibration estimates radial lens distortion and the external orientation parameters of both cameras from a calibration pattern using a bundle block adjustment (Jacobson, 1992). A projective transformation can be computed from the calibration parameters that warps the images to standard geometry. This image rectification greatly simplifies correspondence analysis and the search space is reduced to parallel horizontal epipolar lines **E**.

From the rectified images a disparity map D_k is obtained by correlation matching techniques. The quality of the match and therefore the quality of each displacement value is recorded in a confidence map C_k.The correspondence analysis is split into three parts. First a candidate for a corresponding point is identified in one image, then the corresponding candidate in the other image is searched for along the epipolar lines **E** and third the most probable candidate match between both images is selected based on a quality criteria. This search is repeated for each candidate, that is for each pixel. To select candidates the image grey level gradient **g** is evaluated. The image gradient is a vector field pointing into the direction of changing image texture like grey level edges. Only areas exceeding a minimum image gradient value $|\mathbf{g}| > g_{min}$ can be candidates for correspondence. The quality of the candidate can be estimated when comparing the gradient direction with the search direction. Edges perpendicular to the search direction can be located best while edges parallel to the search direction cannot be located at all. This quality measure C_1 can be calculated in Eq. (1). Candidates with $C_1 = 0$ can not be estimated while candidates with $C_1 = 1$ have highest confidence in estimation.

The estimation of C_1 is carried out for each image pixel. Each pixel with a gradient quality measure of $C_1 > 0$ will be selected as candidate. For each candidate a small measurement window (typically 7*7 pixel) around the candidate position in one grey level image is chosen and the corresponding grey level distribution is searched for in the other image. The search space is reduced to a one–dimensional search along the epipolar line between minimum and maximum disparity values derived from the known minimum and maximum scene distance. The search space may be extended to +/– 1 horizontal lines to account for calibration inaccuracies. The normalized cross correlation (NCC) is calculated between the candidates to select the



a) left original image

b) disparity map
(dark = far from camera,
light = near to camera,
black = undefinded regions)

d) confidence map
of disparity measurement
(dark = low confidence,
light = high confidence)

Fig. 2: Stereoscopic disparity analysis of image pair "house".

most probable corresponding candidate along the search line. The most probable candidate pair is the pair with maximum cross correlation.

In complex scenes there may be multiple maxima or false maxima in the search space due to occlusions, repeated structures or image noise. This ambiguity can be reduced when uniqueness and ordering constraints are exploited. These constraints are based on the fact that there can be no more than one match between left and right image points and that matches are in order for physical surfaces (Marr, 1982). These constraints are employed in an optimum search procedure using dynamic programming that matches all correspondences between left and right image that lie on the same epipolar line. The dynamic programming algorithm was adapted from the work of (Cox et al., 1992). The disparity value obtained for each candidate is recorded in a disparity map.

The NCC is additionally used to define the correspondence quality. Selected corresponding pairs with low NCC are corresponding points with low confidence. Therefore a second quality measure C_2 in Eq. (1) can be defined that reflects the correspondence measurement confidence. Experiments have shown that candidates below a minimum threshold NCC_{min} (NCC_{min} being approximately 0.5) are most often false matches that should be discarded. The confidence quality is therefore defined to be zero below NCC_{min} and NCC elsewhere.

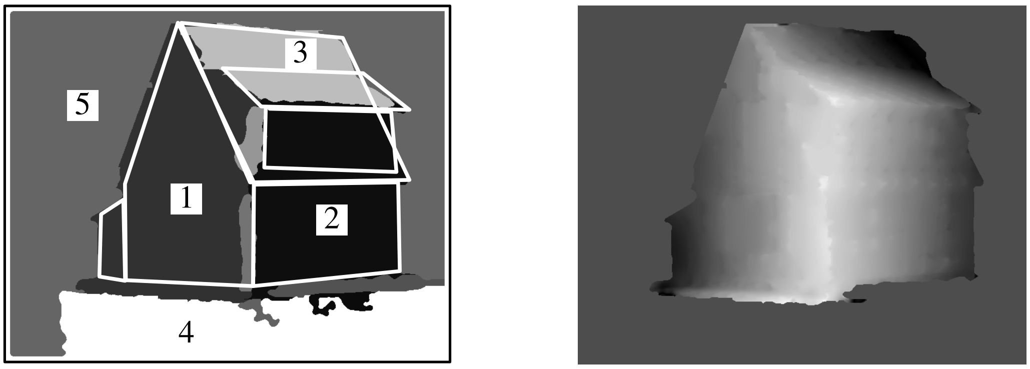
$$C_1 = \begin{Bmatrix} 0 & \text{for } |\mathbf{g}| < g_{min} \\ \dfrac{\mathbf{g} \cdot \mathbf{E}}{|\mathbf{g}|} & \text{else} \end{Bmatrix}, \quad C_2 = \begin{Bmatrix} 0 & \text{for NCC} < NCC_{min} \\ NCC & \text{else} \end{Bmatrix} \qquad (1)$$

Both quality measures can be merged to one measure $C_c = C_1 \cdot C_2$ with $\{0 \leq C_c < 1\}$ that contains the combined quality measure for each candidate. The confidence measure C_c is recorded for each candidate pixel in a confidence map. Fig. 2 demonstrates the correspondence analysis for the image pair ”house”. Disparity values between 50 and 90 pixel were measured. Fig. 2a shows the left of both input images, Fig. 2b the measured disparity map and Fig. 2d the corresponding confidence map. Light grey levels in Fig. 2b show large disparities (foreground) and dark grey levels indicate small disparities (background). Light values in the confidence image indicate high, dark values low measurement confidence. Black regions are regions where the confidence measure is zero and where no measurement was possible.

3.2 Scene segmentation, Interpolation and Triangulation

The correspondence analysis yields a disparity map based on local depth measurement only. These measurements are corrupted by noise and must be merged to regions that describe physical object surfaces. Based on similarity measures the segmentation divides the viewed scene into object surfaces. As similarity measure the local surface orientation which is computed from the estimated disparities is used to group pixels into regions that belong to the same surface. The region boundaries are then corrected from the grey level image with a contour approximation by assuming that physical object boundaries most often create grey level edges in the image. The object segmentation for the image pair ”house” is shown in Fig. 3a with each surface having a distinct label marked as grey level in the map. The segmentation areas correspond to the front (1) and side wall (2), the roof (3), and foreground (4) and background (5) areas. To show the fit of the segmentation, the true edges of the house were superimposed as line drawing.

Disparity Interpolation



a) object segmentation map, each grey level labels one object surface

b) Thin plate interpolation of disparity map (dark = far from camera, light = near to camera)

Fig. 3: Segmentation and interpolation of image pair "house".

The disparity measurements are noisy and there exist gaps in the surface that need to be filled. Once the disparity map is segmented into object regions all measurements of one region are interpolated by a thin plate surface model that calculates the best quadratic surface approximation of the disparity map based on the uncertain depth measures. Each disparity measurement has an uncertainty attributed to it which serves as a weight of the measurement. A multi grid surface reconstruction algorithm (Terzopoulos, 1988) was chosen to calculate the interpolation with a finite element approximation. It is assumed that each segmented area contains a smooth coherent surface that can be modeled as a thin plate with a certain stiffness and that inside such a region the disparity measurements are corrupted by noise. The physical model of a thin plate can be formulated as a variational functional of the Euler–Lagrange equation $\Delta^2 d_{(x,y)} = 0$ with additional constraints at the boundaries. The interpolation solves the problem of minimizing the potential energy function of the thin plate that is deformed by the disparity measurements. The result of the disparity interpolation is shown in Fig. 3b for the scene "house". From the discrete and noisy disparity measurements in Fig. 2c together with the associated confidence values in Fig. 2d and the segmentation mask from Fig. 3a, a continuous and dense disparity interpolation for each segmented region was performed that filled the gaps and smoothed the disparity estimates. Disparity discontinuities are preserved at the segmentation boundaries.

Triangulation

The interpolated depth map contains the visible scene geometry measured from a single camera view point. Whenever the scene contains occluded surfaces then the camera must be moved around the objects and the measurements from multiple view points must be included. For that purpose the 2D depth map is first converted into a 3–D surface description that can be modified to include hidden surfaces. The transformation is very simple because the images are rectified and relative 3–D–coordinates are obtained relative to the left camera center. The camera centers are displaced by the basis b in x–direction and both cameras have the same focal length f. In this case the relative object coordinate $\mathbf{P}_{(x,y)}$ for each pixel (x, y) in the left image with corresponding disparity value d(x, y) is recorded in a depth map $\mathbf{P}_k$.

$$\mathbf{P}_{(x,y)} = (P_x, P_y, P_z)^T = \frac{b}{d_{(x,y)}} \cdot (x, y, f)^T \tag{2}$$

The depth map can be converted into a piecewise continuous, parametric 3–D surface description by spanning a wireframe in space for each segmented object surface. For each object region the depth map is approximated by triangular, planar surface patches. The triangular mesh was chosen because it is capable to approximate arbitrary surface geometries without singularities. On the surface of each triangular patch the object surface texture is stored in a texture map from which a naturally looking view of the original objects can be synthesized with texture mapping. In Fig. 4a the generation of the wireframe for the house is shown. For each triangular patch the corresponding image texture is stored and used to synthesize computer generated views which is shown in Fig. 4b and c. The surface geometry was computed from the interpolated disparity map while the surface texture was taken from the left original image.

4 3–D motion estimation using analysis by synthesis

In this section an algorithm to directly estimate 3–D scene motion from a monocular or stereoscopic image sequence is described shortly. A complete discussion of the algorithm can be found in (Koch, 1993).

a) Triangular surface mesh of walls and roof

b) texture mapping of scene texture onto the triangular surface model

c) synthesized image from another view point

Fig. 4: Triangulation, texture mapping and image synthesis.

An object is defined as a rigid 3–D–surface in space that is spanned by a set of N control points. A set of six motion parameters is associated with each object. Object motion is defined as rotation of the object control points around the object center followed by a translation of the object center, measured between two successive image frames k–1 and k. The object center **G** is the mean position vector of all N object control points. Each object control point $\mathbf{P}_{\mathbf{i}(k-1)}$ at frame k–1 is transformed to its new position $\mathbf{P}_{\mathbf{i}(k)}$ in frame k according to the general motion Eq. (3) between frame k–1 and k.

$$\mathbf{P}_{\mathbf{i}(k)} = [\mathbf{R}_G] \cdot (\mathbf{P}_{\mathbf{i}(k-1)} - \mathbf{G}) + \mathbf{G} + \mathbf{T} \tag{3}$$

with $\mathbf{T} = (T_x, T_y, T_z)^T =$ translation vector, $\mathbf{G} = (G_x, G_y, G_z)^T = \sum_{i=1}^{N} \frac{\mathbf{P}_i}{N} =$ component center, and

$[\mathbf{R}_G] =$ matrix of rotation vector $\mathbf{R} = (R_x, R_y, R_z)^T$

Object rotation can be expressed by a rotation vector $\mathbf{R} = (R_x, R_y, R_z)^T$ that describes the successive rotation of the object around the three axes $(x, y, z)^T$ parallel to the scene coordinate system centered at **G**. From this vector the rotation matrix $[\mathbf{R}_G]$ is derived when the identical matrix $[\mathbf{I}]$ is rotated around the coordinate axes with R_x first, R_y second and R_z last. Because $[\mathbf{R}_G]$ is derived from the rotation vector **R**, the six parameters of **T** and **R** suffice to describe the 3–D object motion.

The only information available to the analysis system is the surface texture projected onto the camera target throughout the image sequence. From this sequence the motion parameters have to be derived. Assume a scene with an arbitrarily shaped, moving textured object observed by a camera during frames k–1 and k. The object moves between frame k–1 and k according to the general motion Eq. (3) with motion parameters **R** and **T**. A point on the object surface, called observation point $\mathbf{P}_{(k-1)}$, holds the surface intensity I_1, which is projected onto $\mathbf{p}_1$ in the image plane at frame k–1. At frame k $\mathbf{P}_{(k-1)}$ is moved to $\mathbf{P}_{(k)}$, still holding I_1 that is now projected onto $\mathbf{p}_2$. In image frame k the surface intensity I_1 will now be projected at image position $\mathbf{p}_2$, whereas the image intensity at point $\mathbf{p}_1$ has changed to I_2.

The image displacement vector $\mathbf{d} = \mathbf{p}_2 - \mathbf{p}_1$ is called optical flow vector and describes the projection of the observation point displacement $\mathbf{P}_{(k)} - \mathbf{P}_{(k-1)}$ onto the image plane. When assuming a linear dependency of the surface texture between I_1 and I_2 and a brightness constancy constraint between frame k–1 and k it is possible to predict I_2 from I_1 and its corresponding image intensity gradients and hence to estimate **d** from the measurable difference $I_2 - I_1$. I_2 is measured at position of $\mathbf{p}_1$ at frame k, whereas I_1 is taken from image position $\mathbf{p}_1$ at frame k–1. When approximating the spatial derivatives as finite differences the optical flow vector $\mathbf{d} = (d_x, d_y)^T$ can be predicted from the image gradients $\mathbf{g} = (g_x, g_y)^T$ and the temporal image intensity difference $\Delta I_{\mathbf{p1}} = I_2 - I_1$ between frame k and k–1 at $\mathbf{p}_1$ in Eq. (4):

$$\Delta I_{\mathbf{p1}} = \mathbf{g} \cdot \mathbf{d} = g_x \cdot d_x + g_y \cdot d_y = g_x \cdot (p_{2x} - p_{1x}) + g_y \cdot (p_{2y} - p_{1y}) \tag{4}$$

In Eq. (4) **d** is related to intensity differences. Substituting the perspective projection of $\mathbf{P}_{(k-1)}$ and $\mathbf{P}_{(k)}$ for $\mathbf{p}_1$ and $\mathbf{p}_2$ in Eq. (4) yields a direct geometric to photometric transform that relates the spatial movement of **P** between frame k–1 and k to temporal intensity changes in the image sequence at $\mathbf{p}_1$.

$$\Delta I_{\mathbf{p1}} = f \cdot g_x \cdot \left(\frac{P_{(k)x}}{P_{(k)z}} - \frac{P_{(k-1)x}}{P_{(k-1)z}} \right) + f \cdot g_y \cdot \left(\frac{P_{(k)y}}{P_{(k)z}} - \frac{P_{(k-1)y}}{P_{(k-1)z}} \right) \tag{5}$$

With this approach, rigid 3–D object motion can be estimated directly from the image sequence when the object shape $\mathbf{P}_{(k-1)}$ is known. Assuming that rotation between successive images is small, $[\mathbf{R_G}]$ can be linearized and $\mathbf{P}_{(k)}$ is substituted in Eq. (5) as a function of the unknown parameter **R** and **T** as derived in Eq. (3) :

$$
\begin{aligned}
\Delta I_{p1} \cdot P_z^2 \quad = \; & f \cdot g_x \cdot P_z \cdot \mathbf{T_x} + f \cdot g_y \cdot P_z \cdot \mathbf{T_y} - [\, \Delta I_{p1} \cdot P_z + f \cdot P_x g_x + f \cdot P_y g_y \,] \cdot \mathbf{T_z} \\
& - [\, \Delta I_{p1} \cdot P_z \cdot (P_y - G_y) + f \cdot P_x \cdot g_x \cdot (P_y - G_y) + f \cdot P_y \cdot g_y \cdot (P_y - G_y) + f \cdot P_z \cdot g_y \cdot (P_z - G_z) \,] \cdot \mathbf{R_x} \\
& + [\, \Delta I_{p1} \cdot P_z \cdot (P_x - G_x) + f \cdot P_x \cdot g_x \cdot (P_x - G_x) + f \cdot P_y \cdot g_y \cdot (P_x - G_x) + f \cdot P_z \cdot g_x \cdot (P_z - G_z) \,] \cdot \mathbf{R_y} \\
& + [\, f \cdot P_x \cdot g_y \cdot (P_x - G_x) - f \cdot P_z \cdot g_x \cdot (P_y - G_y)] \cdot \mathbf{R_z}
\end{aligned}
$$

with $(P_x, P_y, P_z)^T = \mathbf{P}_{(k-1)}$. (6)

For 3–D motion estimation the object shape is assumed to be known. An initial estimate of the scene shape was generated from stereoscopic image analysis. When the initial estimate fails this dependency may affect the analysis and will sometimes lead to estimation errors. As long as the initial shape approximation is reliable, however, this dependency can be neglected. When a stereoscopic image sequence is available, then both images of the pair can be used to further improve the motion estimation. The left image coordinate system is used as reference system and measurements are taken from the left camera as before in Eq. (6). Measurements taken from the right camera will be transformed according to Eq. (7), where an observation point $\mathbf{P}_{R(k)}$ is expressed relative to the left camera coordinate system.

$$
\begin{aligned}
\mathbf{P}_{R(k)} \quad & = [\mathbf{R_{LR}}] \cdot \mathbf{P}_{L(k)} + \mathbf{T}_{LR} \\
& = [\mathbf{R_{LR}}] \cdot \left([\mathbf{R_G}] \cdot (\mathbf{P}_{L(k-1)} - \mathbf{G}_{L(k-1)}) + \mathbf{T} + \mathbf{G}_{L(k-1)}\right) + \mathbf{T}_{LR} \\
& = [\mathbf{R_{LR}}] \cdot [\mathbf{R_G}] \cdot \left(\mathbf{P}_{L(k-1)} - \mathbf{G}_{L(k-1)}\right) + [\mathbf{R_{LR}}] \cdot \left(\mathbf{T} + \mathbf{G}_{L(k-1)}\right) + \mathbf{T}_{LR} \qquad (7)
\end{aligned}
$$

with: $[\mathbf{R_{LR}}], \mathbf{T}_{LR}$ = Transformation from left to right camera coordinate system

The Transformation ($[\mathbf{R}_{LR}]$, $\mathbf{T}_{LR}$) is known from calibration and is particularly easy for rectified images. The motion Eq. (7) for the right image can be inserted in Eq. (5) as before and the measurement equation for the right image is derived which doubles the number of independent measurements for motion estimation.

At least six distinctive observation points that lead to six linear independent equations are needed to solve for the six motion parameters **R** and **T**. In real imaging situations the measurements of the spatial and temporal derivatives are noisy and some of the observation points selected may be linear dependent of each other. To cope with those conditions more than six observations are evaluated and a linear regression is carried out using least squares fit.

4.1 Accumulation of multiple depth maps into a common 3–D scene model

For each image pair of the sequence a depth map D_k was calculated by stereoscopic analysis together with its associated confidence map C_k. 3–D camera motion between successive frames was estimated which allows to register the image pairs relative to another. The goal of sequence accumulation is to fuse the depth measurements from the image sequence into a consistent 3–D scene model to improve estimation quality. Consistency is achieved by compensating the camera motion from frame k–1 to k. The scene model is transformed into frame k with the estimated motion parameters. From the model geometry in this position a prediction of the disparity map d^*_k can be computed and compared with the measured disparity map d_k to detect geometric errors.

Depth measurements are improved by weighted depth accumulation from the motion compensated sequence of depth maps. For each observation point **P** of the surface there exist a confidence value C_c from Eq. (1) that expresses the measurement accuracy. The confidence value C_c is converted into the weight S according to Eq. (8) that can easily be accumulated throughout the sequence. Each observation point holds not only its position $\mathbf{P}_{k-1}$ in space but also its corresponding confidence weight $S_{k-1} = S^*_k$. $\mathbf{P}_{k-1}$ is transformed to $\mathbf{P}^*_k$ according to Eq. (3) and its projection (x,y) in the image is computed. The disparity d_k, measured in frame k at image position (x,y) with corresponding weight S_k, is converted to a depth measurement $\mathbf{P}_k$ and fused with $\mathbf{P}^*_k$ in a weighted accumulation to compute the improved depth estimate $\mathbf{P}_k$new and weight $S_k$new:

$$\mathbf{P}_k\text{new} = \frac{\mathbf{P}^*_k \cdot S^*_k + \mathbf{P}_k \cdot S_k}{S^*_k + S_k} \quad \text{and} \quad \begin{array}{l} S_k\text{new} = S^*_k + S_k \\ \text{with } S = \dfrac{C_c}{1 - C_c} \end{array} \tag{8}$$

5 Conclusion

A system for automatic 3–D scene analysis was discussed. The system is capable to analyze scenes from an arbitrarily moving stereoscopic video camera system. It segments the scene into smooth surfaces and stores the true 3–D geometry of the scene in a 3–D scene model, including surface texture. Camera motion is tracked throughout the sequence and measurements from different view points are integrated into the model data base.

The system implementation is not jet finished. With the current implementation, we are not able to add new scene contents (e.g. from moving around a corner of a house) automatically into the model to include truly occluded surfaces. We are further investigating the impact of erroneous model shape on the camera tracking algorithm and we are working to improve shape accumulation further through Kalman filtering. Some of the analysis parameters for disparity estimation, image segmentation, and surface mesh generation were chosen prior to the analysis process. An important additional step towards fully automated scene analysis will be the extension of the control interface with knowledge based scene interpretation, a project we are currently investigating.

Acknowledgement

This work has been supported by a grant of the German postal service TELEKOM.

References

Aggarwal, J.K., Nandhakumar, N. 1988. "On the Computation of Motion from Sequences of Images –A Review," *Proc. of the IEEE*, Vol. 76 (8), pp. 917–935.

Blake, A., Zissermann, A., 1988. "Visual Reconstruction," *MIT–Press*, Cambridge.

Cox, I., Hingorani, S., Maggs, B., Rao, S., 1992. " Stereo without Regularisation", *British Machine Vision Conference,* Leeds, UK, pp. 337–346, David Hogg & Roger Boyle (ed.), Springer Verlag.

Harashima, H., Kishino, F., 1991. ”Intelligent Image Coding and Communications with Realistic Sensations – Recent Trends,”, *IEICE Transactions*, Vol. E 74 (6), pp. 1582–1592.

Jacobson, K., 1992. ”BUNOR – Stereoscopic bundle block adjustment program”, Institut für Photogrammetrie und Ingenieurvermessung, Universität Hannover, internal communication.

Jarvis, R.A., 1983. ”A Perspective on Range Finding Techniques for Computer Vision,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 5 (2), pp. 122–139.

Koch, R., 1990. ”Automatic Modelling of Natural Scenes for Generating Synthetic Movies”, *Eurographics ’90*, Montreux, Switzerland.

Koch, R., 1993. ”Dynamic 3D Scene Analysis through Synthesis Feedback Control”, *IEEE Trans. Patt. Anal. Mach. Intell., Special issue on analysis and synthesis,* Vol. 15:6.

Marr, D., Nishihara, K.H., 1978. ”Representation and recognition of the spatial organization of threedimensional shapes,” *Proc. Royal. Soc. Lond. B*, vol. 200, pp. 269 –294.

Marr, D., 1982. ”Vision – A Computational Investigation into the Human Representation and Processing of Visual Information”, *W.H. Freeman & Co.*, New York, USA.

Netravali, A.N., Salz, J., 1985. ”Algorithms for Estimation of Three–Dimensional Motion,” *AT&T Technical Journal*, Vol. 64 (2).

Pentland, A., Horowitz, B., 1991. ”Recovery of Nonrigid Motion and Structure,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 13 (7), pp. 730–742.

Terzopoulos, D., 1988. ”The computation of visible–surface representations”, *IEEE Trans. Patt. Anal. Mach. Intell.*, Vol 10, pp.417–438.