Published in the Proceedings of CGI’98,
22 June 1998 in Hannover, Germany.

3–D Modelling of Buildings using High–Level Knowledge

Oliver Grau
University of Hannover, Institut fuer Theoretische Nachrichtentechnik
und Informationsverarbeitung. D–30167 Hannover (Germany), Appelstr. 9A,
E–Mail:grau@tnt.uni–hannover.de

Abstract

A scene analysis system for automated 3–D modeling of buildings is presented. It combines surface reconstruction techniques with object recognition to generate 3–D models for computer graphics applications. The system permits the insertion of high level constraints, like a specific angle between two house walls, in an explicit knowledge base as a semantic net. The applicability of those constraints is proved by asserting and testing hypotheses in an interpretation phase. In the case of rejection, a more general constraint or model is selected. The capabilities of the system were shown for the modeling of buildings using depth from stereo and contour information. The system reconstructs the surface of scene objects using constraints selected in the prior interpretation.

1 Introduction

The presented system is designed for the automatical reconstruction of object shapes from digital images. After restoring the 3–D geometry, texture and color of the objects are taken from the original input images and will be stored in a texture map. With this approach, photorealistic models can be obtained. Manual construction of those models with CAD systems is expensive and often fails to reach photo realism.

Recent progress in graphics hardware development opens new application fields for computer generated 3–D models of buildings, e.g. for flight and driving simulators, architecture and landscape planning and internet applications (VRML). In each case, the resulting models should be represented efficiently, i.e. with the smallest possible number of graphical primitives. General approaches like [2][3] mostly use triangular meshes. Special components, like walls or roofs, are represented by several triangles in these approaches.

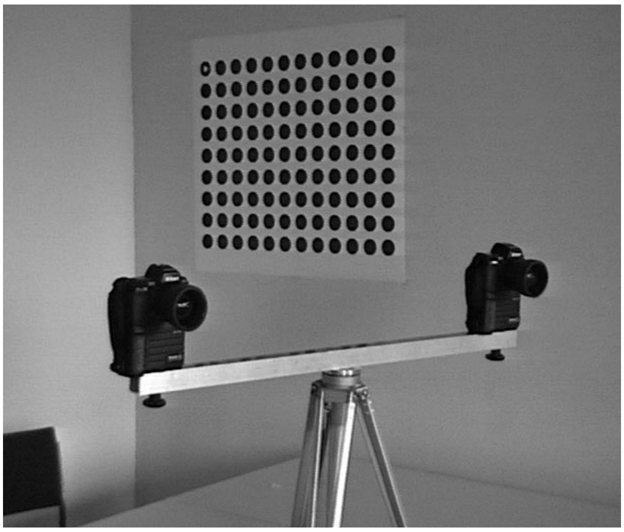


Figure 1: Stereo sensor using two digital cameras with calibration pattern in the background

The approach described in this paper uses a recognitive process to assign high–level knowledge to the components found in the image. The final 3–D model is created using this specific information and results in very efficient polygonal surface representations. For example, components like roofs and walls are reconstructed with only one polygon instead of a more or less high number of triangles.

The input to reconstruct the 3–D object surface are stereo image pairs taken by a two camera set–up as shown in figure 1. In a first step, the scene depth is estimated pointwise for these image pairs. In a second step, the depth values are integrated into a 3–D surface description. Both processes need additional constraints to be well conditioned in the mathematical sense.

The pointwise computation of depth employs a block-matching–based depth estimator under the assumption of piecewise continuous surfaces [1]. The resulting depth maps are usable but noisy for the considered outdoor scenes (see

Published in the Proceedings of CGI'98,
22 June 1998 in Hannover, Germany.

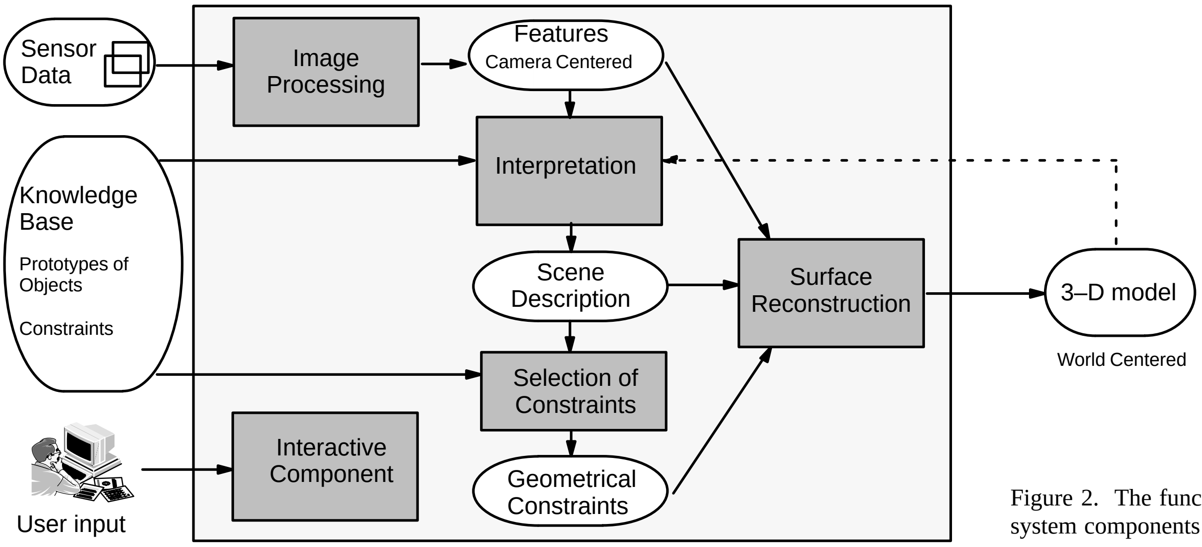


Figure 2. The functional system components

fig. 4 a). Each depth value represents a 3–D point relative to the camera coordinate system.

To integrate the 3–D points into a consistent surface description a suitable surface model must be selected. A commonly used method is to interpolate the 3–D points with a spline function. This general surface model is sufficient for natural objects like persons [2]. For objects with a more special geometry, like buildings, this surface model is not sufficient, because the spline surface smoothes edges and the noise in the depth maps produces a rough surface even in object parts that are planar. For those objects it is necessary to select a better matching surface model, like a planar polygon. In addition relations between object parts, like symmetries or known intersection angles are very useful constraints for the reconstruction of the surface.

The contribution presented here makes use of the scene understanding system AIDA [5][6][7] as a framework for the knowledge representation and the basic control structure of the interpretation. AIDA is also applied to the modeling of landscapes from arial images [7].

The approach discussed here adaptively selects suitable constraints during an interpretational phase. The constraints are represented in a generic, exchangeable knowledge base. The system can be easily adapted to new object classes. After a successful symbolic interpretation of the scene content the derived constraints and the data (depth values, contours) are used for the reconstruction of the 3–D surface.

The following section gives a system overview. Section 3 gives a short description of the image processing methods. Section 4 gives an overview over the interpretation and the used knowledge representation scheme. Section 5 describes the integration of data in the surface reconstruction. The paper closes with a presentation of the results and conclusions.

2 System Overview

Figure 2 shows the main functional components of the presented system. Input data are sequences of stereo image pairs. The data is processed sequentially, i.e. at each cycle one stereo pair is presented at the input. The images pass through several image processing modules where various features are extracted (contours, regions, depth from stereo) and grouped to primitives (2–D edges and 2–D polygons). The features are camera centered at this stage.

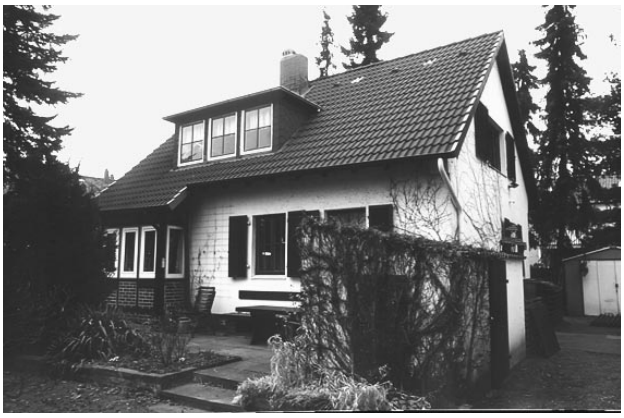


Figure 3: a) left input image of stereo input image pair

The interpretation module groups and labels the extracted primitives and creates a semantic description of the scene in terms of the generic description found in the knowledge base. Section 4 gives an overview over this component and the knowledge representation.

The *selection of constraints* module selects additional geometrical constraints from the knowledge base that are suitable for recovering the object surface.

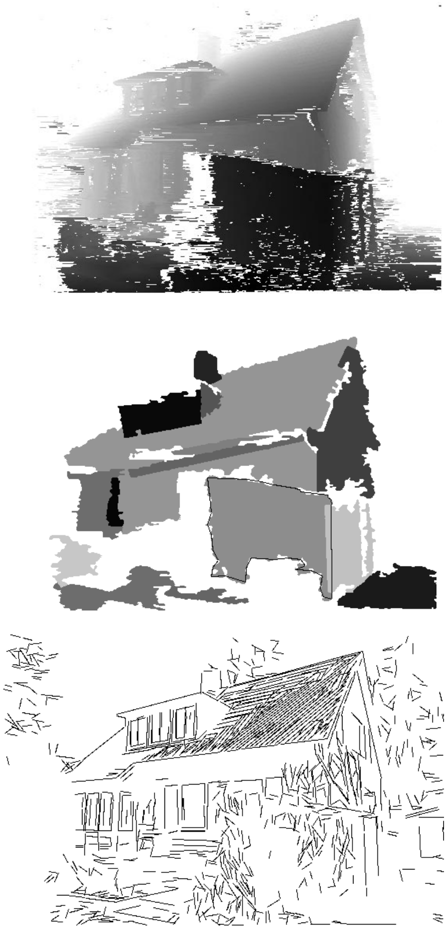
The surface reconstruction described in section 5 performs several tasks: first it transforms new camera oriented primitives extracted from the image processing pipeline into world coordinates and inserts them into the 3–D model data base. This transformation requires the knowledge of the coordinates of the input stereo camera system in a common world coordinate system, which are determined through an estimation of the view point. The surface reconstruction employs depth information from new camera views and the geometrical constraints selected from the knowledge base to further improve already present surface patches created from the prior stereo image pairs.

With the interactive component the user can control the system. The most important role of the interactive component is the development of the generic model description in the knowledge base. The quality of this description determines the performance of the system's recognition facilities and can be improved by an interactive analysis of intermediate results. The interactive component visualizes the processing steps and explains the results of the recognition process.

3 Image Processing

The image processing modules extract various features from the input images, including depth maps. The most important visual cue for the depth estimation is depth from binocular stereo. For the considered application this method turned out to be a reliable and robust technique to recover the depth information from camera images [1][2]. The images were taken with a pair of photographic cameras of the same type mounted on a bar 1 m apart as depicted in figure 1 (example in figure 3).

In a first pre–processing step the stereoscopic camera system is calibrated using a regular pattern of control points. The calibration estimates the intrinsic and relative extrinsic camera parameters. The intrinsic parameters contain the radial lens distortion and the focal length. The relative extrinsic parameters cover the orientation of both cameras relative to each other. With this information the image pair is rectified to achieve epipolar geometry. In the next step a disparity map is calculated using stereoscopic correspondence analysis. The camera parameters are used to calculate depth values from the disparity estimation (depicted in figure 4a).



a) color coded depth map

b) result of the segmentation

c) scene conturs

Figure 4: Image processing results

These depth maps are then used as input for the segmentation of the scene (figure 4b). The segmentation is based on a region growing algorithm and groups 3–D points computed from the depth values that have a small distance to a plane in 3–D. The plane parameters are computed iteratively from all points in the already found region. After the update the region growing algorithm is called again until terminated by a stop condition. The basic algorithm is adapted from [4] and considers the problems of the noisy and discrete depth maps computed with the approach.

In addition to the estimation of depth maps and regions, contours are extracted from the luminance images using an edge detector and a line segment approximator (figure 4c). All features or primitives are camera centered at this processing stage.

4 Scene Interpretation

The goal of scene interpretation is to assign a meaning found in a knowledge base to each primitive found in the input images. The image primitives are the scene segments and conturs as described in the previous section.

A lot of approaches for scene interpretation can be found in literature. While nearly all of them try to cope with the unreliable results of low level vision modules, most can not properly handle 3–D information or do not provide the flexibility to cope with new visual cues. The main problem with 3–D data is to handle occlusions in different camera views. To meet these requirements the interpretation system AIDA was developed.

4.1 The Knowledge Representation

In AIDA knowledge about the scene to be modeled is represented explicitly as a semantic net. To describe objects and their relations a problem independent net language was defined which mainly resembles the net syntax of ERNEST [8].

Each node of the generic prototype net represents an object and is called concept. Concepts are depicted as boxes in figure 5. The nodes in the scene description are called instances and represent real objects. They are depicted as ellipses in the following figures. The interpretation starts with a copy of the prototype net and creates and inserts those instance nodes into the net that match the signal.

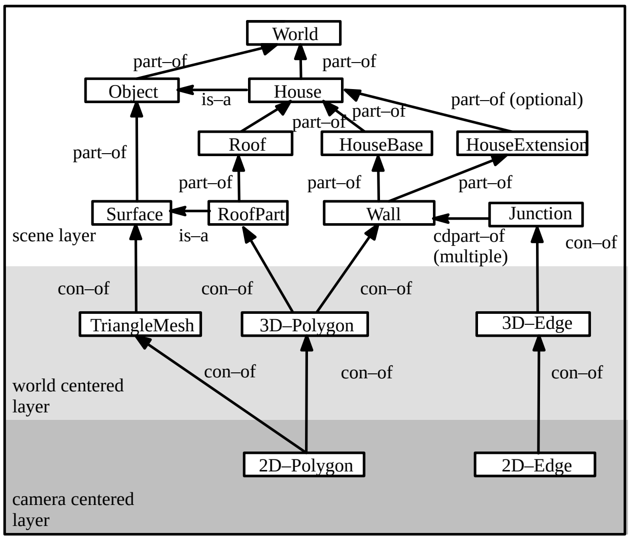


Figure 5: Simplified part of the knowledge base.

Each node can contain several attributes like photometric or geometric features. These attributes are employed to test the compatibility of a concept with an instance.

The nodes of the net are connected via links. The most important links are: *instance–of, part–of, cdpart–of, is–a* and *concrete–of*. The *instance–of* link connects instances with their prototype concept. The *part–of* and *cdpart–of* links allow a decomposition of complex structures. The *cdpart–of* (context dependant part of) joins concepts that can not exist on their own, like the concept *Junction* that is only defined in the context of a wall.

The link *concrete–of* has an important role. It links two levels of different abstractions. The knowledge base is structured into three layers: The scene layer, the world centered layer and the camera centered layer. For example the *concrete–of* link assigns, as depicted in figure 5, the concept *Wall* from the most abstract scene layer to its concretization *3D–Polygon* in the world centered layer and to *2D–Polygon* in the camera centered layer. The nodes in the camera centered layer are related to the signal and contain attributes that are directly measured in the image or image primitives.

The *is–a* link implements inheritance and provides the specialization of object descriptions. In figure 5 the node *World* contains the parts *Object* and the specialization *House*. The interpretation module will consider both nodes as competing hypotheses. In the case an object can be detected as a house more special knowledge can be applied in the latter surface recognition.

Nodes in the prototype net can be connected to more than one child node, like different parts. Also some nodes in the prototype net have more than one parent node, for example the node *Wall* which has the nodes *HouseBase* and *HouseExtension* as parents. The quantity of obligatory and optional connections (links) is a parameter of the corresponding link in the prototype network. In figure 5 the *HouseExtension* is connected as an optional part.

In the scene description net the number of connections of a node to its parent is usually exactly one. An instance of the concept *Wall* for example has exactly one connection to the appropriate node *House*. For the concept *Junction* in figure 5 a multiple binding is enabled in the *cdpart–of* link from *Junction* to the concept *Wall*. This setting causes the interpretation to search for two parents of an instance of *Junction* during the interpretation and joins these (see figure 9).

4.2 The Interpretation Module

During interpretation semantics described in a knowledge base are assgined to each primitive found in the input image. To describe the scene, the interpretation process successively builds up a network of instance nodes. Figure 6 shows results of two simple example scenes.

Starting with a *World* instance the interpretation tries to build up a scene description. Hypotheses are generated for each part of the concept *World* like the occurrence of a house and unspecific objects (of type *Object* in figure 5) in the scene. Then the mandatory parts of this node are searched for, and in the case of a match of the parts (e.g. walls, roof etc.) in the knowledge base the instance *House* can be established.

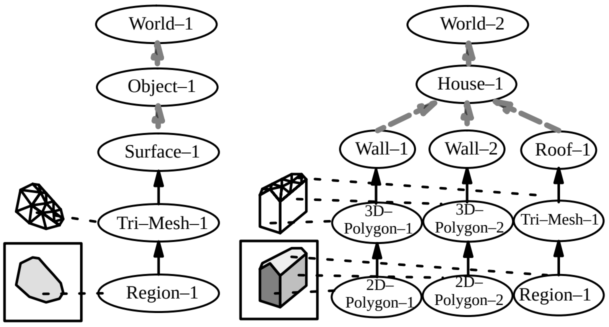


Figure 6: Scene descriptions with a semantic net
a) unspecific object (left), b) house (right)
Link types: ⇧ part–of, ▲ concrete–of

To assess the match of an instance with the generic description a composite numerical value ranging from 0 (no match) to 1 (optimal match) is generated according to geometrical and photometrical features. Structural relations are included in the assessment calculation too, and are represented as links between components of the object. The problem is that they can be missing, e.g. because the neighboring object components have not yet been established in the current state of interpretation. The geometric and photometric features are (mostly numerical) attributes of the concepts. A match function between two attributes usually evaluates the difference between the instance attribute and the expectation represented by the concept. The function that evaluates the compatibility of an attribute can be defined for each concept class separately. The judgement function of the concept finally sums up the available components, calculates a judgement value for the match and a factor for the certainty (or probability) of the judgement.

If the hypothesis of the occurrence of a house is rejected a new one will be generated and tested. If more than one hypothesis exists for one object the most probable will be selected. The selection is made by an A* algorithm that considers both; certainties and cost of an interpretation.

The most unspecific interpretation of a scene part is *Object* which is depicted in figure 6a. In this case no additional constraints can be selected for the surface reconstruction and the geometry will only be computed from the (noisy) depth maps approximated by a triangular mesh (node *Tri–Mesh* in figure 5 and 6).

During the interpretation 3–D information and knowledge about the objects are used to restrict the search space. Assuming one wall of a house already has been found in the previous interpretation stage there are four hypotheses of a possible neighboring wall. Three of them can be rejected because their surface normal is not directed towards the current camera position. By using knowledge about the expected size of a wall in 3–D space and the estimated orientation; the range to search of the walls projection in the camera image (2D–Polygon) can be reduced.

The interpretation ends if there are no further hypothesis instances that could be verified or if a user specified concept was instantiated. The second condition is true if all obligatory parts of the concept are found. A more complete description of the interpretation principles can be found in [6].

5 Surface Reconstruction

The surface reconstruction integrates the depth measurements into a consistent 3–D model employing the constraints generated during the interpretation. The module restores the shape of the 3–D surfaces found in the scene description by using all assigned data that is useful for the reconstruction. In addition to the data found in the depth map and the contours the model selection (3–D polygon or triangular mesh) is directly found in the scene description. The knowledge that a wall is a polygon is in fact a strong and important constraint for surface reconstruction. For each point in a segmented region the 3–D coordinates are computed from the depth map (figure 4a). All points within the region contribute to an overdetermined equation system. It is solved by a least square minimization to find the 3–D plane which belongs to the region. The boundaries of the polygons are determined by intersection of the found plane with neighboring polygon planes, as described in section 5.2.

After successful reconstruction of the object geometry the surface color is stored in a texture map. The texture map is generated from the original images as described in section 5.4.

5.1 Used Constraints and Data

The input data for the surface reconstruction is: (i) the regions of planar polygons in space found in the segmentation (section 3) and (ii) their bounding edges found as grey level contours in the input images.

All points within a labeled region of (i) are reprojected into 3–D space using the depth value and are integrated to a single plane measurement E_m. The plane is determined by a weighted regression with optional certainty factors of the depth estimation serving as the weights.

Both the surface reconstruction and the view point estimation (section 5.3) use a uniform representation of the input data. The measured data from the image processing is handled like the additional constraints derived during the scene interpretation. Therefore the contours and depth values are compared to the model.

The measured plane E_m and the bounding edges in the input image refer to the same polygon in 3–D space and can be formulated as cost functions c_{ji} that measure the fit to the model:

c_{1i} : the difference between pose and orientation of the polygonal description from the segmentation and the depth values to the polygonal model,

c_{2i} : the distance between model and corresponding contour edges.

These cost functions that correspond to data outside the model are called external constraint in the following. On the other hand internal constraints are not directly related to the image data. The following are used in this approach:

c_{3i} : The angle between two polygons,

c_{4i} : parallel polygon edges,

c_{5i} : equal length of edges and

c_{6i} : equal angles (between 3 or more polygons).

The use of internal constraints is optional but very helpful in cases of uncertain or noisy depth measurements.

5.2 Integration into a Consistent 3–D Model

The surface reconstruction integrates the plane and edge measurements together with additional internal constraints into a consistent polygonal 3–D model. The boundaries of the polygons are obtained by intersecting neighboring 3–D planes:

$$L_k = \text{Intersect}\ (E_i, E_{ik})$$

The different measurements (plane, edge and constraints) produce a conflicting description due to noise in the image and the depth map. This inconsistency is solved by a numerical optimization that minimizes the objects overall cost function f_{glob}:

$$f_{glob} = \Sigma\Sigma\ w_{ji} \cdot c_{ji}(\mathbf{p}) \ \text{–> Min.} \qquad (1)$$

Each constraint is represented by it's cost function $c_{ji}(\mathbf{p})$ in this equation. The parameter vector **p** is modified during the numerical minimization and contains the current object geometry, i.e. all polygons that describe the object surface. The factor w_i weights the influence of each constraint. It strongly depends on the application.

The minimization of the global cost function leads to a surface description which best meets both the measured information and the constraints derived from the knowledge base. The surface reconstruction progresses incrementally. Hence it is possible to include new data like depth values from new camera views and new surface patches from prior invisible object parts into the 3–D model.

5.3 View Point Estimation

For the integration of new camera views the camera position and orientation must be known. For practical tasks like the modeling of outdoor scenes the parameters are usually unknown and must be estimated.

The view point estimation takes into consideration the extracted features of a new camera image and tries to find a set of camera parameters (location and orientation) that optimally fits the already generated 3–D model. In case of planar polygons found in the new camera view the correspondence to the model is assessed by measuring the difference between normal vectors. Additionally the match between the model's projected 3–D edges and the edges found in the image is measured. The view point estimation uses the same minimization of the global cost function f_{glob} as described for the surface reconstruction in equation (1). Here the parameter vector **p** contains the camera position and orientation rather than the model geometry which is fixed. Precondition for the view point estimation is that a major part of the already modeled object is visible in the new camera view.

The correspondence of object parts in the image and the already modeled 3–D object shall be established by the interpretation. At the current state of the system implementation this step is mainly guided manually.

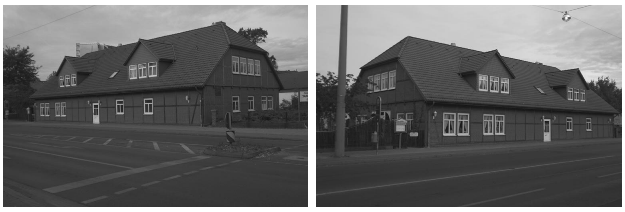


Figure 7: Input images of scene ”Restaurant”

5.4 Generation of Texture Maps

The last step of the surface reconstruction is the storage of the surface color in a texture map. The texture maps are created using an inverse perspective mapping [9]. In figure 8 an example is shown for the roof of the house depicted in figure 7. The top most picture shows the texture map generated from the right picture of figure 7. Due to occlusions of some parts of the roof the map is not complete.

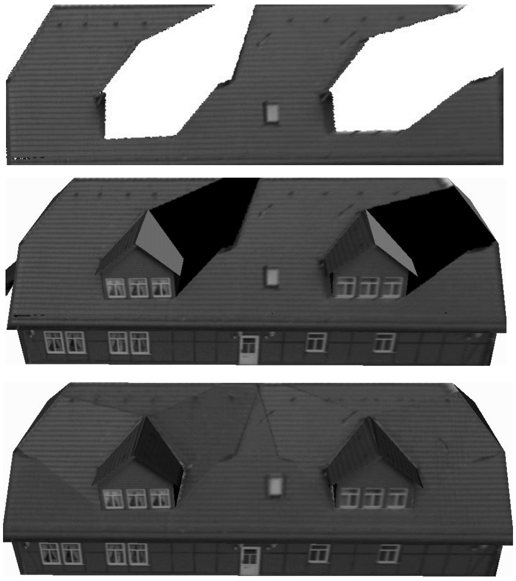


Figure 8: a) Texture map from right view of fig.7 (top)
b) textured model
c) model with combined texture map from two views (bottom)

The solution proposed here is the combination of several views. For that purpose the maps must be remapped to an equal (norm) size. The resulting map is then computed by adding and normalizing the single maps. The bottom picture of figure 8 shows a model using the resulting combined map.

6 Results

Figure 9 shows the 3D model of the house depicted in figure 3. The mantatory parts of the house (roof and walls) were reliably found by the interpretation. Further the house extensions, the dormer and chimney were found using geometrical and topological features.

Figure 10 shows the result of modeling another outdoor scene. It uses the data extracted in the image processing and additional constraints found in the interpretation. The model is created using two different camera views.

For the integration of the presented 3–D models, the view point estimation as described in section 5.3 was used. The neccessary correspondencies between the views were given manually.

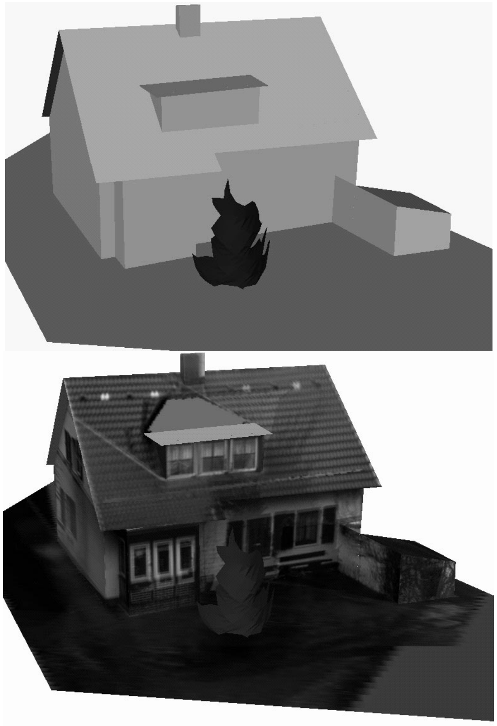


Figure 9: 3–D Modell ”House” generated from 3 views

7 Conclusions

An automated system for the modelling of 3–D models of buildings was presented, which combines recognition and reconstruction facilities. With the proposed knowledge representation it is possible to explicitly formulate properties of the objects to be analyzed. The interpretation module plays a central role. It selects the applicable models and constraints useful for the surface reconstruction. The most important advantage of the approach is the explicit knowledge which describes the objects to be separately modeled from the system implementation. Thus only the description of the objects has to be changed to apply the system to a new object class. The performance of the system has been demonstrated for the modelling of buildings. The adaptation to other objects like furniture should be easy.

The system allows the introduction of specialized high level constraints, like a specific angle between two walls. The applicability of those constraints is proved by asserting and testing hypotheses. In the case of rejection a more general constraint or model is selected.

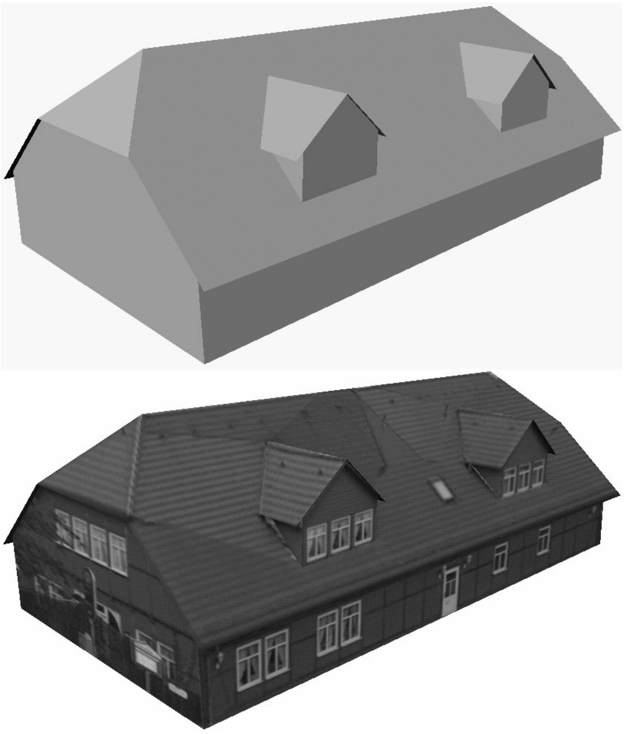


Figure 10: Result of the scene ”Restaurant” from two camera viewpoints (fig.7). In shaded (top) and textured (bottom) view

The surface reconstruction module integrates the different data types (depth maps and conturs) and the additional constraints using a numerical optimization. The optimization is robust and able to cope with a portion of badly assigned data or constraints.

For the considered application of modeling buildings the approach leads to accurate models as depicted in figures 9 and 10. Due to the use of polygons the surfaces are planar and edges are straight. Quality of photo realism is reached that satisfies most requirements of computer graphic applications. The critical structures for the modelling are considered in the knowledge base. This is the main difference compared to data driven approaches, where the surface approximation is made by a triangular mesh that doesn’t consider the semantics.

8 References

[1] L. Falkenhagen, ”Depth Estimation from Stereoscopic Image Pairs Assuming Piecewise Continuos Surfaces”, Workshop on Combined Real and Synthetic Image Processing for Broadcast and Video Production. Nov. 1994 Hamburg. ISBN 3–540–19947–0, Springer.

[2] Koch, R., ”3–D Surface Reconstruction from Stereoscopic Image Sequences”, ICCV’95, Boston, June 1995.

[3] Gool, Luc van, Zisserman, A., ”Automatic 3D Model Building from Video Sequences”, ETT Vol.8, No.4, Jul–Aug. 1997.

[4] Leonardis, A., Gupta, A., Bajcsy, R., ”Segmentation of Range Images as the Search for Geometric Parametric Models”, Int. Jour. of Computer Vision, Vol. 14, No.3, April 1995.

[5] C.–E. Liedtke, O. Grau, S. Growe, ”Use of Explicit Knowledge for the Reconstruction of 3–D Object Geometry”, 6th International Conference CAIP’95 Computer Analysis of Images and Patterns. Sep. 6–8, 1995, Prague, Czech Republic

[6] O. Grau, ”A Scene Analysis System for the Generation of 3–D Models”, Proc. of Int. Conf. on Recent Advances in 3–D Digital Imaging and Modeling. May 12–15, 1997, Ottawa, Ontario Canada. IEEE Comp. Soc. Press. ISBN 0–8186–7943–3

[7] C.–E.Liedtke, J. Bückner, O. Grau, S. Growe, R. Tönjes, ”AIDA: A System for the Knowledge Based Interpretation of Remote Sensing Data”, 3rd International Airborne Remote Sensing Conference, July 7–10, 1997, Copenhagen, Denmark.

[8] Niemann, H., Sagerer, G., Schröder, S., Kummert, F., ”ERNEST: A Semantic Network System for Pattern Understanding”, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 12, No. 9, pp. 883–905, Sept. 1990.

[9] Heckbert, P.S., ”Fundamentals of Texture Mapping and Image Warping”, Master’s thesis,UCB/CSD 89/516, CS Division, U.C. Berkeley, June 1989.