

ON THE USE OF GEOMETRIC AND SEMANTIC MODELS FOR COMPONENT-BASED BUILDING RECONSTRUCTION

André Fischer and Thomas H. Kolbe and Felicitas Lang

Institute for Computer Science III, Institute for Cartography & Geoinformation, Institute for Photogrammetry
University of Bonn

e-mail: andre@cs.uni-bonn.de, kolbe@ikt.uni-bonn.de, fl@ipb.uni-bonn.de

Semantische Modellierung; *SMATI 99*

KEY WORDS: 3D building reconstruction, hierarchical 2D/3D modeling, component-based reconstruction, inexact relational matching, minimum description length principle, multi-image correspondence analysis

ABSTRACT

3D building data is needed in many application areas. Besides the geometric description an increasing number of applications also demand thematic information about acquired buildings. We present a concept for the automatic extraction of buildings from aerial images. In contrast to other approaches generic building structures both are geometrically reconstructed and semantically classified. A component-based, parameterized building model is employed to control the reconstruction of buildings.

This paper describes how geometric and semantic knowledge of buildings is propagated through the different aggregation levels of the building model. Furthermore, it is shown how rules and constraints are derived from the model and exploited at each stage of the reconstruction process.

1 INTRODUCTION

3D building data play an important role for digital city models and geoinformation systems, which are used in numerous application areas: architecture, urban planning, telecommunication and environmental investigations show an increasing need for up-to-date 3D building models. To reduce the cost of time-consuming manual data acquisition the automation of 3D building reconstruction became an active research area in recent years.

The quality requirements like accuracy, degree of abstraction, and actuality of the 3D building data to be acquired in general depend on the application area. Especially in GIS applications like urban planning not only the geometric description of the buildings is necessary, but also their semantic classification to facilitate thematic inquiries. Therefore, it is essential that automatic building reconstruction procedures extract both the geometric and semantic aspects of buildings.

Several research groups presently are working on 3D building reconstruction and have demonstrated promising results. The work on automatic building reconstruction reveals essentially two different approaches. One employs a database of predefined buildings, which are matched with the images to extract the corresponding building instances. The other utilizes a general or prismatic polyhedral model, where buildings are reconstructed by geometric grouping of image features. Whereas the former approaches yield

both geometrical and semantical 3D building information, they are limited by their fixed number of predefined buildings. In contrast, the generic model of the latter approaches allow the representation of arbitrarily shaped buildings, but provide no building specific interpretation of the reconstructed polyhedrons.

Up to now only semiautomatic systems are able to perform both geometric and semantic reconstruction using generic scene descriptions (cf. [LLS95], [EG96], [TDM96]).

In earlier articles (see [BKL⁺95, FKL97, FKL⁺98]) we have presented a concept for automatic building reconstruction which enables both the reconstruction *and* the interpretation of generic building structures. The employed building model is based on building specific components that are combined in a flexible way. Rules and constraints represent the building specific knowledge. They are derived from the building model and guide respectively restrict the reconstruction. In this paper we present these rules and constraints in detail, and show how they are used and exploited in the different stages of the reconstruction process to ensure the reliable extraction and classification of buildings.

2 DOMAIN SPECIFIC MODELING

2.1 Related Work

In the past ten years the development of concepts for automatic building extraction has been a topic of ac-

tive research. For an up-to-date overview of the different approaches see the Ascona workshop proceedings 1995 and 1997 [GKA95, GBH97], the SMATI97 workshop proceedings [FP97], and the special issue of the CVIU journal on building extraction [FKL⁺98].

Although the proposed concepts differ wrt. to the type of input data (e.g. digital elevation models and raster images), their reconstruction strategy, and implementation, they can be compared by their employed building model. Following the line of Steinhage in [Ste98], we basically classify the employed modeling schemes into three categories:

Polyhedral models provide the most flexible approximation for the representation of buildings. They pose no constraints on the form of the objects and thus allow the representation of arbitrarily shaped buildings with plane faces.

The use of general polyhedral descriptions is problematic, because of the missing semantic classification of the objects and the object parts. To avoid the reconstruction of impossible objects, building knowledge has to be incorporated into the reconstruction procedure, for instance, in form of heuristics and implicit assumptions. This can be observed in the approaches of Bignone et. al. [BHFS96] and Frère et. al. [HVF⁺97]. Both concepts are based on a polyhedral model. A bottom-up strategy is used, starting with an initial image feature extraction, where features are successively grouped to 3D roof patches using photometric and chromatic attributes of image regions with spatial edge geometry. Roof patches then are grouped by an overall optimization according to the simplicity, compactness, and completeness of the resulting roof shape. To complete the building shape, vertical walls are assumed. Henricsson et. al. [HB97] present impressive results of their approach on some test data, but show no explicit modeling of building types or building specific parameterizations. Due to the lack of further building specific modeling it cannot be decided whether a reconstructed object is a building, a lorry or a doghouse.

Another drawback of a general polyhedral model is, that due to the absence of a more specific building modeling no predictions about occluded parts can be made. Buildings can only be extracted completely, if every part is visible and is detectable by the feature extraction process. Thus, this approach will only give good results, when there are several images from different viewpoints available and the quality of the images is high.

Prismatic models can be seen as a special case of polyhedral models. They allow the representation of buildings with arbitrary ground plans, but are restricted to vertical walls and flat roofs. They are based on the assumption that the ground plan of a flat roof house is congruent with the roof's outline.

Prismatic models are used in a number of different approaches: Nevatia et. al. [NP82, MN88, LHN95, NLH97], Fua and Hanson [FH87], and Weidner et. al. [WF95, Wei97]. Whereas the possible building shape is restricted explicitly in these concepts, buildings are not further classified and have no specific parameterization. In all cases the reconstruction strategy is bottom-up, following mostly the principles of perceptive grouping (cf. [Moh89]) to successively construct geometrically — but not necessarily semantically — more meaningful objects. Thus, prismatic models suffer from the same lack of building specific knowledge as general polyhedral models.

Parameterized volumetric primitives are building models with fixed topology and variable size. Modeling buildings by volumetric primitives has several particular advantages: because every building type is explicitly modeled, their different forms of appearance can be derived a priori. Second, even partially occluded buildings can be fully reconstructed, and third, the identification of an instance of a volumetric primitive in the image implies its classification.

Parameterized volumetric primitives are employed by, among others, McGlone and Shufelt [MS94], Jaynes et. al. [JHR97], and Schutte et. al. [Sch96, SSH97]. All approaches use a hypothesize-and-verify strategy, where building primitive instances are hypothesized in a data-driven way, and then are verified top-down according to the model. The major drawback of volumetric primitives is the lack of flexibility wrt. to different building shapes, because only buildings can be detected, that are explicitly stored in the model database.

Discussion: Whereas (prismatic) polyhedral models allow the representation of buildings in a generic way, they are too unspecific and do not take explicitly into account constraints wrt. to the domain 'building'. Parameterized volumetric models instead represent a semantically more meaningful modeling scheme, because building type specific parameters like width, height, and length can be used to further restrict models to reasonable sizes. Unfortunately, they are restricted to the detection of a small number of predefined building types. To bridge the gap between both approaches, we have developed a component-based building model, which is described in the next section.

2.2 Building Model

To overcome the limitations of parameterized volumetric primitives with their fixed topology, we employ a component-based modeling. Buildings with complex topology can be constructed by connecting parameterized building part primitives in a generic way.

Each building part primitive is parameterized by its own set of form parameters like roof height, building width etc. A set of constraints is defined on these pa-

rameters, introducing further specific building knowledge into our model. Simple constraints define upper and lower bounds for valid parameter intervals. More complex constraints relate parameters to others and enforce geometric properties (i.e. roof slopes). Parameters keep their meaning during the aggregation of building part primitives. Thus, a meaningful parameterization is maintained during the aggregation process.

In detail, the proposed building model consists of a four level part-of hierarchy (see fig. 1). It reflects different levels of semantic abstraction. The primitives of each aggregation level are specialized by an is-a hierarchy into subclasses. Building specific knowledge and restrictions are propagated top-down from the higher levels to the lower ones. To allow tight coupling of 2D and 3D reconstruction processes, the primitives on each level are coherently modeled in 2D and 3D.

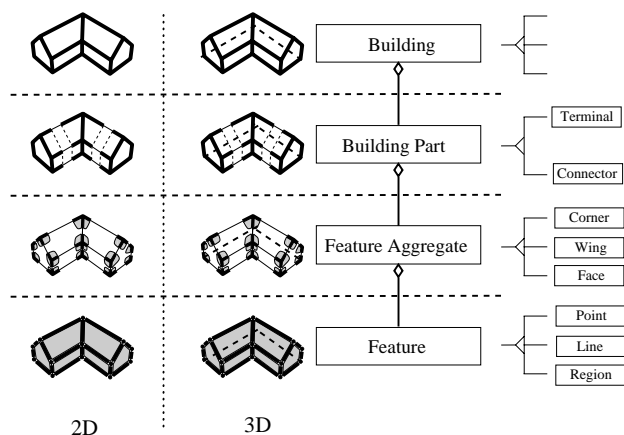


Figure 1: Building model: Different semantic levels of the part-of hierarchy are shown in vertical direction, different levels of the is-a hierarchy in horizontal direction (here shown only for the 3D side). The 2D image model describes the expected appearance of the building in the images.

The lowest level contains attributed *features*, namely points, lines and regions. They establish the link to a symbolic image description, that is derived from the original raster images by a feature extraction process. Attributes for lines and regions, for instance, are the orientation classifications *horizontal*, *vertical*, and *oblique*. Regions have an additional attribute describing their role: valid values among others are *wall* and *roof*.

The next level consists of feature aggregates, which are induced by points, lines and regions and contain all their topological neighbours. They are subdivided into corners, wings and faces. A corner, for example, contains one point and all its adjacent lines and regions.

The two top levels contain the already mentioned building parts and buildings. Building parts are composed

from feature aggregates. Actually, they are represented by corner graphs.

3 STRATEGY

In a data-driven preprocessing step the aerial images are segmented and features are extracted from them. Regions of interest which are likely to contain buildings are determined and all following steps focus on these regions and the features contained in them.

The further process is strictly model-driven and follows the paradigm of hypothesize and verify. It can be divided into three parts which use the building model in different ways.

The transition from 2D image data to the 3D object space is realized by the reconstruction of 3D vertices from 2D image features. Classification transforms the vertices into 3D corners.

The corners are used to index into the database of parameterized building part primitives, which are instantiated and aggregated into building hypotheses.

In order to verify the building hypotheses, they are projected back into the images. The obtained 2D views are matched with the image features using constraint solving techniques. Of all best matches the most probable is determined and selected. A final parameter estimation determines form and pose parameters.

A detailed descriptions of these three parts follows in the next three sections.

4 3D CORNER RECONSTRUCTION

One crucial part within the complete reconstruction process is to derive 3D object parts which serve as an appropriate basis for the subsequent aggregation process. For the transition into object space we use the level *feature aggregates*, especially the building components of type *corner*. The use of corners is motivated by the following reasons:

Observability: The projections of corners into the images are image structures which show a high stability against occlusions.

Structure: The topologic and structural properties of corners are helpful for steering the search procedure during the correspondence analysis and have the advantage of using aggregates in contrast to single features.

Interpretation: The richness of the observable geometry and topology of the corners, especially in 3D object space, is suitable for interpreting the observed 3D descriptions. The interpretation then can be used for stabilization and verification.

Aggregation: The corner geometry as well as the corner semantics give powerful information for the subsequent 3D aggregation process (cf. section 5.4) and thus enables to use a generic model for 3D aggregation.

The corner reconstruction process follows the hypothesize and verify paradigm. Hypotheses are build up mainly data-driven inferring the class membership of the observed data to a corner class from analyzing the observable geometric description. They represent model instances with fixed geometry, topology and structure. Having built up the hypotheses, the verification can be performed model-driven and exploits the class-specific constraints for evaluation and for increasing the accuracy of the reconstruction.

The corner reconstruction is performed as a multi-image procedure using all available image data simultaneously. The interior and exterior image orientation is presumed to be known and is used for geometrically restricting the search during the correspondence analysis.

4.1 3D and 2D Corner Model

The level *feature aggregates* of our building hierarchy distinguishes three different object types, namely corners C , wings W and faces S (cf. [FKL⁺98]), each of them given by its geometric and topologic form description and its semantic description. They represent components of a general polyhedron which additionally are specified by building-specific geometrical, topological and structural properties. In this context, we restrict ourselves to describing corners because wings and faces so far are of subsidiary importance for our building reconstruction procedure.

3D Corner Model: Each corner $C = (v, \psi_c)$ is given by its form description being a vertex v composed from elements on the level *features* and its semantic description by a class label ψ_c . The distinction of corners from wings and faces is due to the structural composition of the underlying feature aggregates. Each corner C of order n is described by the corner point P and each n lines L and regions F that are preliminarily open-ended in their spatial extension (cf. fig. 2 a.) and thus represent plug elements for grouping and aggregating corners. The graph representation (cf. fig. 2 b.) expresses the topologic and structural description of a corner where the graph nodes represent features and the graph arcs the adjacency relations between them.

The building-specific part of the corner model, which makes the difference of a building corner from a general polyhedral corner depends on the corner specialization hierarchy, which divides corners into subclasses. The specialization depends as well on the

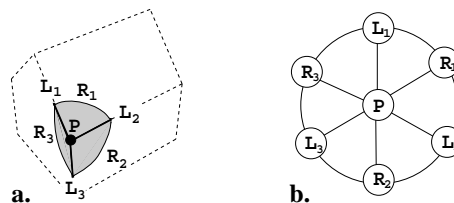


Figure 2: a. shows a corner represented by its components points, lines and regions. b. shows the graph representation of a corners.

corner topology as on its geometry. We use a *two-level specialization hierarchy*. Each subclass implies class dependent constraints Θ_{ψ_c} onto the form description of the vertex v . On the first level of the specialization hierarchy we use unary constraints which refer to single components of a corner. They especially restrict the corner components of type line to building-specific qualitative attributes like being horizontal, vertical or sloped. On the second level of the specialization hierarchy we use binary constraints which refer to the geometric relationship of pairs of corner components and likewise restrict the corner components of type region to building-specific attributes like being horizontal or vertical.

If no constraints are attached to the corner we call it the unconstrained corner with class label ψ_0 , which is identical to a polyhedral corner, that is a vertex v .

2D Corner Model The image model aims at describing object components coherent to the 3D modeling to provide direct access from image observations to 3D objects and vice versa. The image representation coherently uses components on the hierarchy level *features*. Therefore we use an appropriate feature extraction to derive a polymorphic symbolic image description consisting of points, lines and regions and their mutual neighborhood relations (cf. [Fuc98]). The result is stored in a feature adjacency graph analogous to the graph representation in object space (cf. fig. 2 b.).

At the beginning of the corner reconstruction, the geometric corner instances are unknown. As the geometric variability of the appearance of the different corner types is too large for an efficient representation in the image model we renounce integrating the corner specialization into the image model. So the image model is a vertex model, especially modeling the topological and structural properties of the appearance of a corner depending on its underlying vertex.

Furthermore, due to the image characteristics and the characteristics of the feature extraction procedure (cf. [FLF94]), the observed symbolic image description in general differs from the ideal projection of the corresponding 3D corner. Fig. 3 shows some typical deviations of the observed vertices from their ideal

projection for the example of a corner of order $n = 3$. To take into account the uncertainties of the feature aggregates we therefore statistically formulate the image model by a vertex classification model (cf. [FKL⁺98]) which classifies feature aggregates being vertices or non-vertices taking for reference the ideal projection of a corner of order 3 (cf. fig 3a.).

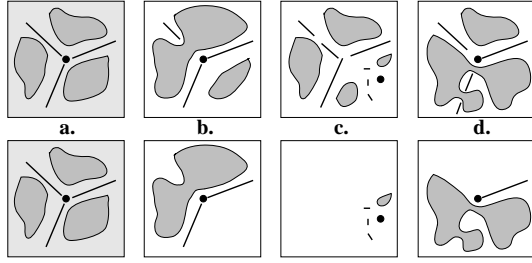


Figure 3: shows the extracted features (first row) and features with a neighborhood relation to the corner point (second row). a. shows the ideal projection of a corner of order 3, b.-d. show possible appearances of extracted image corners differing from the ideal projection.

4.2 Generation of 3D Corner Hypotheses

The corner reconstruction starts with the generation of corner hypotheses and comprises the reconstruction of 3D vertices and their interpretation as 3D corners.

Initial form reconstruction of 3D vertices: By multi-image correspondence analysis we build an initial form reconstruction of 3D vertices V^{3D} , which possess the topologic and structural properties of corners as defined by the image model.

- *Feature extraction and vertex selection:* The first step in reconstruction is the extraction of a symbolic image description. Based on the extracted points, lines and regions and their mutual neighborhood relations we build vertices V^{2D} by point-induced 2D aggregation analyzing the feature adjacency graph. The selected vertices serve as a basis for the correspondence analysis.
- *Correspondence analysis of vertices:* The correspondence analysis starts with the selected vertices and is formulated as search procedure which extends over three layers. In the first layer a most promising vertex v_i^{2D} in image i is selected for reference using a priority list of the 2D vertices by evaluating their suitability for the correspondence analysis and reconstruction. The evaluation is done by vertex classification using the image model and considering the stability, uniqueness and structural richness of a vertex. The second layer establishes a stereo correspondence of vertices $[v_i^{2D}, v_j^{2D}]$ in the images i and j . In addition to the evaluation score of single

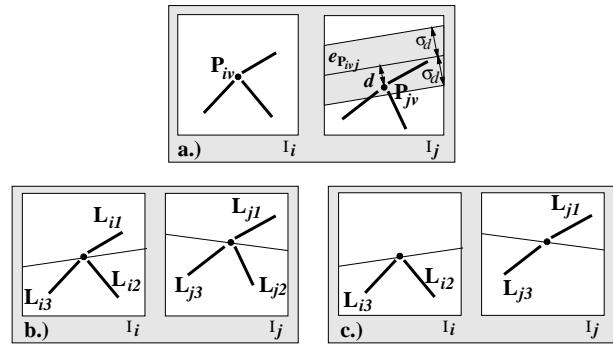


Figure 4: a.) shows the epipolar constraint for the corner point of a vertex pair. The corner point in image I_j is restricted to lie on the epipolar band around the epipolar line defined by the point in image I_i . b.) and c.) show two examples of corresponding vertices. The epipolar geometry restricts the matching of the corner lines. Example b.) is unambiguous. Thus three corner lines can be reconstructed. In example c.) only one corner line can be reconstructed using the line L_{j3} in image I_j . In image I_i the matching is ambiguous and both lines L_{i2} or L_{i3} can be taken for correspondence leading to two different vertex alternatives.

vertices the structural similarity of matching candidates which fulfill the epipolar constraints is incorporated. The third layer generates a multi-image correspondence tuple $[v_i^{2D}]$. Epipolar geometry one again gives restrictions of the search space.

- *Transition to 3D vertices:* Based on the correspondence tuple $[v_i^{2D}]$, the transition to object space is performed by a joint forward intersection of the corresponding image feature components points $[p_i^{2D}]$ and lines $[L_i^{2D}]$ of the vertex correspondence tuple $[v_i^{2D}]$ using all images $i \in I$ simultaneously. Epipolar geometry once again defines geometric restrictions, which facilitate the matching of the features.

Interpretation: For the 3D vertices we perform the interpretation by class assignment of each vertex to a corner classes given by the object model, which results in corner hypotheses. The interpretation is performed in object space as the 3D information defines stronger restrictions during the interpretation than in 2D. By geometric analysis each 3D vertex is assigned to one or several alternative corner classes defined by the 3D corner specialization hierarchy. The 3D vertex classification is performed hierarchically by checking possible class-specific constraints and inferring the corresponding corner classes:

- *Checking for unary constraints:* We start checking unary constraints on the first level of the corner specialization hierarchy. We use qualitative line attributes depending on the slope orientation with respect to the corner point, given by the qualitative geometric labels horizontal (h), vertical+ (v+),

vertical- (v-), oblique+ (o+) and oblique- (o-) [Gül92].

Actually we specialize corners C of node degree 3. We exclude meaningless corners like e.g. corners with line attributes $\{(h), (h), (h)\}$, which make no sense in the context of buildings and their functionality. Thus the first specialization level compounds 21 possible classes.

- *Checking for binary constraints:* Depending on unary constraints that are identified as being satisfied binary constraints are checked. Examples of binary constraints are *symmetry* of two lines with respect to the vertical or *orthogonality* of two lines. The explicit definition of subclasses on this level is not sensible as we cannot be capable to predefine it without restricting the variability of buildings.

The set of identified constraints of each vertex are related to a class label ψ_C resulting in a corner hypothesis $C^{3D} = (V^{3D}, \psi_C, or)$. If no constraints are attached to the corner we call it the unconstrained corner with the class label ψ_0 .

- *Model based corner prediction:* In general the vertex observations may be incomplete. Therefore we use the corner model on the one hand for prediction of unobserved corner components due to missing or fragmented features, or missing feature adjacency relations or on the other hand for prediction of constraints. The prediction only can be performed if the predicted instances are geometrically defined by the model. E. g. if we observe a reconstructed vertex of node degree 2 with line labels (h) and (o-) with the two lines further fulfilling the constraint *orthogonal*(h, o-), we perform a model based prediction of a corner of order 3 of class (h, o-, o-) where the line pair (o-, o-) fulfills the constraint *vertical symmetry*(o-, o-) as it occurs for the gable corner of a gabled roof building with the lines (o-, o-) spanning a vertical plane.

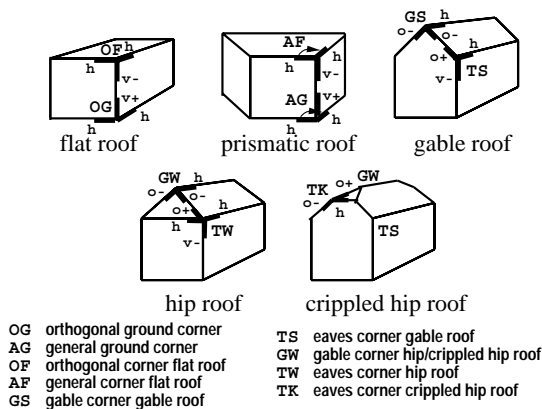


Figure 5: Different corner types that are defined by the building specific corner specialization hierarchy.

4.3 Verification of 3D Corners

In contrast to the mainly data-driven generation of corner hypotheses the verification is performed model-driven and profits from a strong model. On the one hand the geometric model instances give approximate values for further steps of the analysis. On the other hand the class membership defines class-specific constraints that can be used for geometric stabilization and for checking the conformity of data and model.

As each 3D vertex can be assigned to several corresponding corner classes and thus leads to alternative 3D corners the verification has to resolve this ambiguity. Further on the verification is necessary to sustain predicted corners possibly by additional observations to avoid blind prediction. Otherwise the predicted corners have to be rejected. In both cases the knowledge of the model instances defines stronger constraints than during the generation of corner hypotheses.

The verification of the corner hypotheses is performed by statistical analysis and is formulated as an optimization problem for finding the best interpretation \hat{C} of the data $[V_i^{2D}]_v$ from all possible corner hypotheses. Using bayes theorem, the optimization of the conditional probabilities $P(C_c | [V_i^{2D}]_v)$ can be broken down to

$$\hat{C} = \arg \max_{C_c} P(C_c | [V_i^{2D}]_v) \\ \propto \arg \max_{C_c} P([V_i^{2D}]_v | C_c) P(C_c) \quad (1)$$

neglecting the denominator by normalization. Equation 1 clearly shows the influence of the model C : The conditional probability $P([V_i^{2D}]_v | C_c)$ thereby evaluates how good the corner instance fits the observed image features of the vertex correspondence tuple $[V_i^{2D}]_v$ and can be derived using the classical modeling techniques of observation errors. The a priori probability $P(C_c)$ gives information on the probability distribution of the different corner classes.

For each corner hypotheses we perform a maximum likelihood parameter estimation using all supporting image features simultaneously.

The verification process is subdivided into the following steps: a. the model-based selection of observations, b. the model-based form reconstruction of each corner hypotheses and c. the optimization of the interpretation.

Model-based selection of observations: For each corner hypothesis the model instance defines approximate values for the model-based reconstruction. Thus the selection of matching features is done by back-projecting the instantiated corner model into the images and analyzing the deviations between the projected model and the symbolic image description.

That way we possibly get access to features which were originally not contained in the selected vertices and be able to partially bridge the incompleteness, fragmentation and missing neighborhood relations of the extracted symbolic image description.

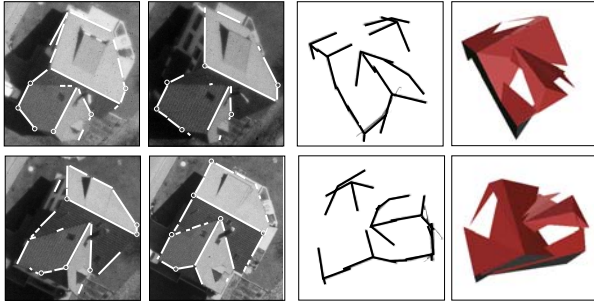


Figure 6: shows image features of type line and point (left) which are used for the corner reconstruction (right) during the verification steps. They are selected model-based by back-projection of the corner hypotheses into the images. Please note that the matching features contain fragmented lines and bridge missing feature adjacency relations to the vertex points.

Model-based form reconstruction: For each hypothesis, we estimate the geometric parameters by a maximum likelihood parameter estimation using all supporting image features simultaneously. The result of the parameter estimation gives for each hypothesis the optimal geometric reconstruction with the expected values $E(y)$ of the observations depending on the functional model $E(y) = X\Delta\beta$ with Σ_{yy} which defines the relation between the observations y and the geometric parameters $\Delta\beta$. The parameters β of each corner class ψ_C depend on the class-specific constraints Θ_{ψ_C} and thus are predefined by the model. Each corner C of order n in principle requires $3 + 2 \cdot n$ geometric parameters β being the three coordinates of the point vector x of the corner point P_p and two parameters for each of the n corner line L_n , being the direction angles λ_n and the azimuth ϕ_n (cf. fig 7 a.). These parameters are reduced by the class specific constraints (cf. fig 7 b.) For corner representation we actually store the $n + 1$ 3D points being the corner point P_p and virtual points $P_n(x + r_n)$ along the corner line direction which are given by the normalized direction vectors r_k , determined from the angle parameters of the corner line L_k . The estimation uses multi-image point and line observations simultaneously.

Optimization of reconstruction and interpretation: The optimization of the corner reconstruction uses eq. 1 which contains both the data-dependent part $P([V_i^{2D}]_v | C_c)$ which is responsible for the optimal geometric reconstruction and the model-dependent part

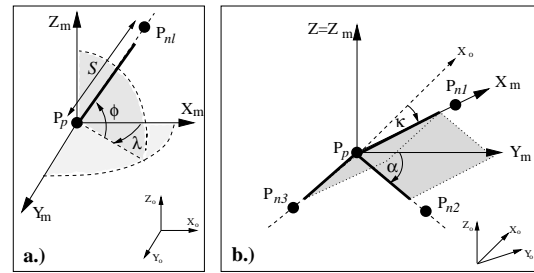


Figure 7: a. parametrization of an unconstrained corner line. b. parametrization of a corner of type gable corner gable roof. The $3 + 2 \cdot n$ parameters β of the n -corner are reduced to three coordinates of the point vector x and two angle parameters α and κ .

$P(C_c)$ which is responsible for the optimal semantic reconstruction.

- **Optimal geometric reconstruction:** For optimization of the geometric corner reconstruction, the evaluation can be derived from the residuals $\hat{e} = y - \hat{y}$ of the optimal estimation $\hat{y} = f(\hat{\beta})$ using the *probability density function* $p(y | \beta) = p([V_i^{2D}]_v | \beta)$ in case the features exist and have been successfully matched to the model.

$$p(y | \beta) = \frac{1}{(2\pi)^{n/2} (\det \Sigma_{yy})^{1/2}} e^{(-\frac{1}{2} \hat{e}^T \Sigma_{yy}^{-1} \hat{e})} \quad (2)$$

The evaluation depends on the squared sum of the residuals $\Omega = \hat{e}^T \Sigma_{yy}^{-1} \hat{e}$ of the estimation $\hat{y} = f(\hat{\beta})$. n is the number of unknowns in β and Σ_{yy} is the covariance matrix of the observations y .

- **Optimal semantic reconstruction:** The a priori probability $P(C_c)$ for the corner class ψ_C can be obtained empirically by learning (cf. [Eng97]) and can in principle be integrated to evaluate the model dependent influence on the result. Actually we prefer specialized corners ψ_C from unconstrained corners ψ_\emptyset and assume equally distributed corner classes ψ_C .

4.4 Grouping of 3D Corners

The next step in reconstruction is to perform a 3D grouping of corners. In contrast to the subsequent 3D aggregation, the grouping connects the corners solely using the knowledge of the aggregation level *feature aggregates* without using any building-specific aggregation relations. The result of the grouping process is a corner adjacency graph CAG, whereas the graph nodes are classified and attributed corners and the graph arcs denote a connection between corners, possibly evaluated by its connection probability.

The grouping is performed in two steps:

- **Qualitative compatibility:** The first step in grouping is to check the qualitative compatibility of the corner pairs which depends on the class specific constraints of the corners. Therefore we postulated

the class compatibility of the corners and their components stored in a class-connection-table. Due to their class-specific constraints a corner of type `eaves corner gable roof` can be connected to a corners of type `gable corner gable roof` but not to a corners of type `gable corner hip roof`. On the feature level a corner line of type `o+` can only be connected to a line of type `o-`. As the class compatibility of the corresponding corner components is partially expressed by the corner classes itself why we first check the corner classes and then the corresponding feature classes.

- **Quantitative compatibility:** The class-compatibility of corners is necessary but not sufficient for the grouping of corners. Therefore the second step in grouping is to check the quantitative compatibility of the class-compatible corner pairs analysing the underlying geometric corner instances. E. g. for connection two lines of type `horizontal` have to be collinear, two regions of type `horizontal` have to be coplanar. The geometry can be statistically tested exploiting the statistical properties of the features given by the covariance matrix $\Sigma_{\hat{y}\hat{y}}$ of the estimated feature geometry \hat{y} which results from the parameter estimation while corner reconstruction (cf. [HLF99]).

During the 3D aggregation the corner adjacency graph can be used to facilitate and focus the indexing into parameterized building parts (cf. chap. 5.4) or building primitives (cf. [Lan99]).

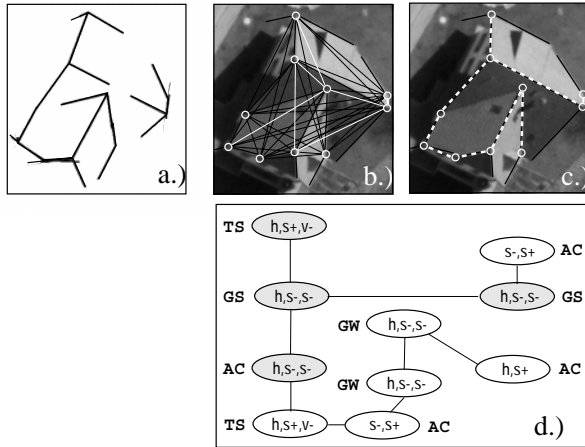


Figure 8: shows the reconstructed corners and the corner grouping. a.) wire frame representation of the reconstructed corners. b.) corner connections of class compatible corners denoted in black. c.) adjacency relations of the corners, which fulfill the class-compatibility as well as the compatibility of the geometry. d.) representation of the corner adjacency graph CAG. The graph arcs denote connectivity relations between corners. The graph nodes denote corners attributed by their class membership.

4.5 Results and Accuracy

The presented procedure for corner reconstruction and interpretation for the time being aims at delivering an appropriate basis for reconstruction of generic buildings by 3D aggregation of the corners using the proposed hierarchical building model.

The corner reconstruction is tested with stereo image data with 2–9 images overlap using image scales between 1 : 4000 up to 1 : 12000 and 10cm – 24cm pixel size on ground. The achieved accuracy of the reconstructed corner points is internally estimated with $\sigma_x = \sigma_y = \pm 3cm$ and $\sigma_z = \pm 5cm$. The orientation accuracy of the corner main direction amounts $\sigma_\kappa = 0.2 [deg]$ whereas the slope of the roof is determined with $\sigma_{sl} = 1.3 [deg]$.

For testing the complete building reconstruction procedure we used the the international distributed data set *Avenches residential* with scale 1 : 5000 which was disposed for the Ascona workshop on *Automatic Extraction of Man-Made Objects from Aerial and Space Imagery* (cf. [GKA95], [MBS94]). The corner reconstruction is performed using the grey level images with a pixel resolution of 12cm. Altogether 56% of the building corners were reconstructed and identified to have an adjacency relation to at least another corner. 82% of these are correctly reconstructed and 18% are false corners. The correct and false reconstructed corners are distributed with 72% to 10% and 4% to 14% onto the building-specific classes ψ_c and the unconstrained corners of class ψ_0 . The reconstructed corners are sufficient for aggregating the 11 buildings contained in the data set (cf. [FKL+98]).

For details concerning the 3D corner reconstruction process we refer to [Lan99].

5 GENERATION OF BUILDING HYPOTHESES

This section describes the construction of building hypotheses for a given set of reconstructed corner observations. Furthermore, it is shown how 3D building hypotheses are transformed into a parameterized 2D view hierarchy, which allows a subsequent verification.

5.1 Model

In our component-based building model, parameterized building part primitives are instantiated and aggregated to more complex building parts and finally complete buildings. Instantiated primitives and aggregations of them are referred to as *building aggregates*. Each building aggregate is represented by a graph of corners $G = (C, R)$ where $R \subset C \times C$ is a adjacency relation on the set of corners C .

Each building part has one or more *plug faces*, by which the building parts can be connected with each other. A plug face $f = (C_f, R)$ is a subgraph of G consisting of one single loop of corners $C_f \subset C$. The building model assigns a type to each plug face. Only plug faces with compatible types may be connected together. Thus the model reduces the number of connectable building aggregates considerably. Building parts with one plug face are called *terminals*, such with two or more plug faces are called *connectors* (see fig. 9).

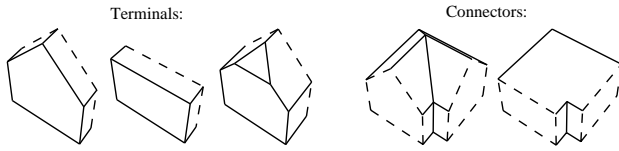


Figure 9: Some examples of building part primitives. Plug faces, which are used to connect them, are drawn dashed.

The connection of two building aggregates to a new aggregate reduces the set of plug faces. The two plug faces taking part in the connection are removed and do not belong to the newly generated aggregate. The goal of the aggregation process is the generation of a building aggregate with an empty set of plug faces. Such aggregates are called *closed*, otherwise *open*.

The coordinates of each vertex of all corners in C are parameterized by a set of pose parameters and a set of form parameters. The four pose parameters determine the location and rotation about the vertical axis. The reduction to one rotation parameter is imposed by our knowledge of buildings, because other rotations are seldomly observed. The form parameters define features like slope of the roof, width and height (see fig. 10). The coordinates may depend nonlinearly on the parameters. If a parameter has an assigned value it is called *fixed*, otherwise it is called *free*.

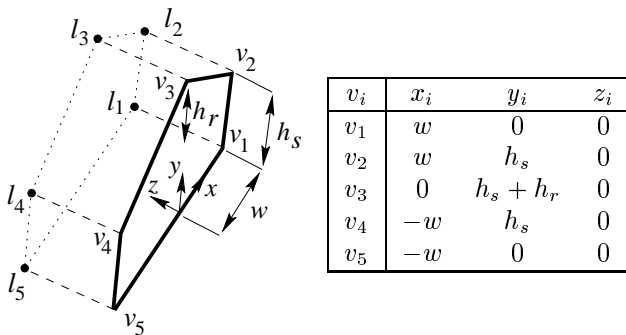


Figure 10: Parameterized description of a saddle roof terminal. The table on the right shows the coordinates of the vertices v_1, \dots, v_5 .

The parameterized coordinates are stored as symbolical expressions to allow their adaption and modification during connections and the construction of

equation systems that are needed for computing parameter estimations.

5.2 Operations

Four different types of are used for generation and aggregation of the building aggregates:

Indexing provides the link between the corner observations and the building part primitives. For a given corner observation an indexing operation selects a building part primitive from a library that has a matching corner, instantiates that primitive and generates by this a new building aggregate. Subsequent invocations of indexing operations for the same corner observation yield different building aggregates as long as there are corners of building part primitives that have not yet been used in this process.

The classification of the corner observation as described in section 4.2 is used to limit the number of potential candidates for instantiation. Model corners with a different classification from that of the given corner observation are ignored.

Merging unifies two building aggregates A_1 and A_2 which are instantiations of the same building part primitive. This operation is necessary, because for n corner observations that belong to the same building part indexing operations generate n instantiations of the same primitive instead of one. The set of plug faces of the resulting aggregate is identical to that of A_1 and A_2 .

Connection aggregates two building aggregates A_1 and A_2 to a more complex aggregate A . The set of plug faces is reduced by the two elements F_1 and F_2 whose corners are to be connected with each other. F_1 and F_2 must have the same type. With this condition the building model restricts the number of connectable building aggregates.

During the connection process each corner of F_1 is identified with one corner of F_2 . In a gluing process, both building aggregates are combined into the new aggregate, such that each pair of identified corners is replaced by a new corner having all of their edges. By design of the plug faces these are exactly two edges lying on a straight line. In a second step each of these newly created corners is removed and their adjoining edges are replaced by one new edge.

The following six steps describe the processing of the parameters in detail. Pose and form parameters are independent of each other so the pose parameters can be ignored until the last step. *For an example refer to figure 11.*

1. The connection is carried out in the coordinate system of A_1 . Both A_1 and A_2 are translated and rotated such that the normal vector of the F_i are aligned with the x -axis, the origin of F_1 lies in the origin of the coordinate system and the origin of F_2 lies on the positive x -Axis.

2. The set of form parameters P_i of A_i is divided into the disjoint subsets $P_i = P_{i,F} \dot{\cup} P_{i,\bar{F}}$. The set $P_{i,F}$ contains all parameters that are referenced by vertices of the plug face F_i . The remaining parameters of A_i make up the set $P_{i,\bar{F}}$.

3. A new length parameter l is introduced into $P_{2,\bar{F}}$ by translation of A_2 by the vector $(l, 0, 0)^T$. If P_2 already contains a parameter with that name l has to be renamed. This length parameter describes the distance between the plug faces F_1 and F_2 . In fig. 11 the origins of the two plug faces are depicted as two discs. It is not chosen as a fixed value because different observations might be contradictory about its value.

4. In order to build the set of form parameters P of the resulting aggregate A the four sets of parameters $P_{1,\bar{F}}$, $P_{1,F}$, $P_{2,F}$ and $P_{2,\bar{F}}$ are processed by the following steps:

a) Though being of the same type F_1 and F_2 may have different parameterizations. To cope with that each plug face has two translation tables. One translates its parameters into expressions over the set of parameters of a generic parameterization of the plug face type of the F_i . The second translates in the other direction.

With these tables the parameters of $P_{2,F}$ are translated to expressions over the set P_F of parameters of the plug face type. The parameters of P_F are then mapped to expressions over the set $P_{1,F}$. In fig. 11 the parameters b , h_1 and h_2 of A_2 are first mapped to b , h_1 and h_2 because F and A_2 have the same parameterization. They are then mapped to b , h and $b \tan \alpha$ of A_1 since the roof of A_1 is described by its slope α instead of its height like h_2 in A_2 .

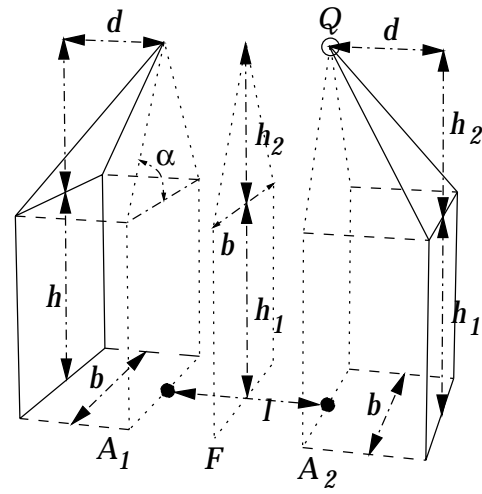
b) To avoid duplicate names, some of the parameters in $P_{2,\bar{F}}$ may have to be renamed. In fig. 11, parameter d of A_2 is renamed to d' in A_1 . The resulting set is named $P'_{2,\bar{F}}$.

The set of parameters of the new building aggregate A is $P = P_{1,\bar{F}} \dot{\cup} P_{1,F} \dot{\cup} P'_{2,\bar{F}}$.

5. All the renamings and translations of form parameters in step 4 have to be propagated to all parameter references of A_1 and A_2 , especially the vertex coordinates. In fig. 11 the origin of the coordinate system coincides, as stated above, with the origin of the plug face of A_1 , which is depicted by the left disc. Then the point Q of A_2 , marked by a circle, has after the connection the coordinates $(l, h + b \tan \alpha, 0)^T$.

6. The transformation back into the world coordinate system reintroduces the pose parameters. It is done by a following parameter estimation of form and pose parameters which is described in section 5.3.

Prediction operations instantiate building part primitives for which no observations exist, but which are needed for the construction of a closed building hy-



$b \tan \alpha$	\longleftarrow	h_1	\longleftarrow	h_1	Translation
h	\longleftarrow	h_2	\longleftarrow	h_2	
b	\longleftarrow	b	\longleftarrow	b	
d'	\longleftarrow	d			Renaming
l					New

Figure 11: Connection of two hip roof terminals A_1 and A_2 . The plug face type F is shown between them. The diagram shows how parameters are mapped, renamed or introduced.

pothesis. It only instantiates building part primitives if no other operation is executable. The new building part is chosen so that two existing building aggregates can be joined by it or that one open building aggregate can be closed by it. This ensures, that a prediction operation is carried out only as a last resort and that some information of one or two other building aggregates can be used to guide the prediction.

The connection itself is not done by this operation, but by one or two subsequent connection operations.

5.3 Parameter Estimation

Each operation except the prediction operations is followed by a parameter estimation which is also used to verify that the newly created building aggregate matches its associated corner observations well enough. If a verification is not possible, the new building aggregate is rejected.

In the previous section we have shown how building aggregates are constructed at runtime. The drawback of this is, that it is not possible to set up the equation system beforehand that represents the least square optimization problem. Instead it is constructed on demand every time a parameter estimation is required. The symbolic representation of the vertex coordinates makes this possible.

Each vertex of a building aggregate has a x , y and z -coordinate which is parameterized with respect to the aggregates form parameters. The multiplication of these vectors with the matrix that transforms the aggregate into the world coordinate system introduces the four pose parameters. Expressions of edge directions can be derived from the edges end points.

Therefore we have for every corner observation which is assigned to one model corner three equations of the corner point and three equations for every edge of the corner. Equations that do not contain a parameter are removed. The resulting nonlinear equation system describes a least squares parameter estimation problem. It is solved with a standard Levenberg-Marquardt algorithm with the only difference that it operates symbolically on the equation system. Please note, that also the derivatives used in the algorithm are computed symbolically.

5.4 Aggregation Strategy

Building aggregates are connected to more complex aggregates and finally to closed building hypotheses in an iterated process. A priority queue holds all operations not yet carried out. Initially it is filled with all indexing operations. If after the execution of the first operation in the queue the resulting building aggregate is not rejected, new operations are created which have the new aggregate as one argument and are inserted into the queue. Because the execution of the operations and the parameter estimations are computationally expensive, and there exist an exponential number of combinations of the building parts with respect to the number of corner observations we employ a heuristic to define the order of the operations inside the queue. The goal is to use as few operations as possible to generate a building hypothesis that explains the reconstructed corners.

Our construction principles for this ordering are:

S1 Indexing is performed before any aggregations or mergings take place. This provides the starting point for the algorithm.

S2 Aggregation is done depth first. This constructs building aggregates with as much covered corner observations as soon as possible.

S3 The growth of competing building aggregates shall be as uniform as possible. This avoids having one aggregate growing while others are neglected.

S4 Predictions take place only if necessary. Because a prediction operation can not use a corner observation directly it has less information than the other operations and therefore its results are more uncertain.

These construction principles result in the following ordering of the operations. Let π_1 and π_2 be two operations.

1. If π_1 and π_2 have different operation types, indexing operations come first, then merging, connection and finally prediction operations (S1 and S4).

2. So π_1 and π_2 have the same operation type.

Indexing operations. If the operations belong to different corner observations,

a) then the corner with less building part interpretations is preferred (S2),

b) else the operation with an a priori better expected result is performed first (S2).

Merging or connection operations. Let the building aggregates $b_{i,1}$ and $b_{i,2}$ be the arguments of π_1 resp. π_2 .

a) The operation π_i with the higher number of corner observations covered by $b_{i,1}$ and $b_{i,2}$ is performed first (S2).

b) The operation π_i with the smaller difference in corner observations covered by $b_{i,1}$ and $b_{i,2}$ is performed first (S3).

c) Finally, the operations π_i are sorted according to the sum of the evaluations of $b_{i,1}$ and $b_{i,2}$. Smaller values first (S2).

Prediction operations. Prefer the operation with the larger number of corner observations covered by their arguments.

5.5 Example

For an example of the aggregation process refer to figure 12. Six corners have been reconstructed from the images (a). The indexing operations assign several instances of building part primitives to each corner. Those with the best scores are shown in the figure (b). Two merging operations combine the three terminals in the front to one aggregate (c). The three terminals and the one connector in the center are aggregated in three connection operations to a closed building hypotheses (d). Please note that there are no corner observations at ground level, and therefore the height parameter is undetermined. For visualization a default value has been chosen.

5.6 Parameterized View Hierarchies

In order to verify the generated building hypothesis, they are transformed into parameterized view hierarchies. For each aerial image one parameterized view hierarchy is generated. This hierarchical image model is a modified aspect hierarchy [DPR92]. It consists of four levels:

Level 1: This level contains all of the three dimensional building hypotheses generated by the algorithm described in section 5.4. Note that a building hypothesis may still have undetermined parameters, hence the name *parameterized* view hierarchy.

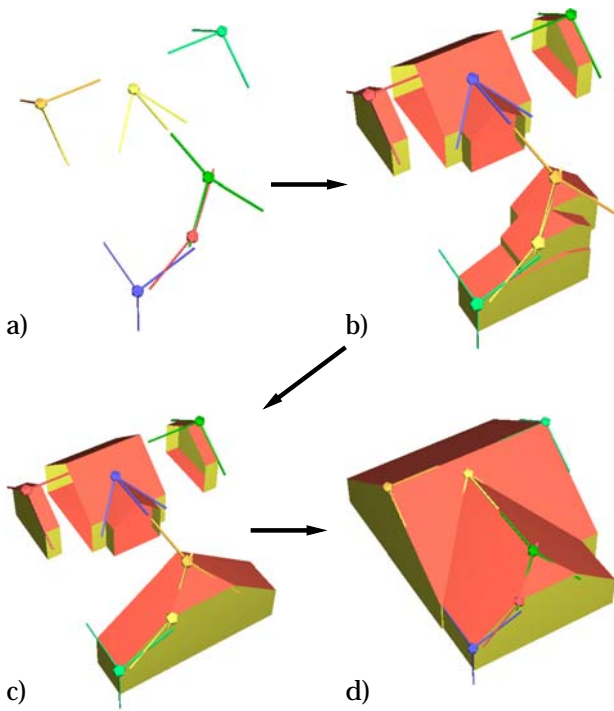


Figure 12: Aggregation: a) The reconstructed corners; b) Best fitting building aggregates after all indexing steps; c) The three building aggregates in the front are merged; d) Connection of these building aggregates results in a closed building hypothesis.

Level 2: For each building hypothesis all topologically different views are stored on this level. In contrast to an aspect hierarchy, we assume a fixed eye position defined by the known camera parameters of the aerial images. Ideally this would lead to one view per building hypothesis and image. But uncertainty of the camera parameters and free parameters of the building hypotheses make it necessary to do a sampling of this parameter space, which may result in more than one view.

Level 3: Each view is decomposed into its regions. A region is either a (part of a) face projection or a (part of a) shadow region.

Level 4: The lowest level consists of image features, namely points, lines and regions, and their interrelationships e.g. line parallelism.

The image model currently employed for hypothesis verification utilizes a pinhole sensor model with weak perspective projection of the visible object contours. Knowledge about the date and time when the aerial images were taken, the geographical position of the building hypotheses, their geometrical appearance and physical material of the different parts of a building allow the inclusion of lighting information. Currently we are investigating view representations which employ a standard lighting model including ambient and directed light and diffuse reflection. The analysis of illumination and shadows allows the derivation of

3D object parameters and to test the consistency of shadow-ground transitions and inter-surface intensity ratios [Ste97].

6 VERIFICATION OF BUILDING HYPOTHESES

The previous processing step can produce different possible 3D building hypotheses depending on the ambiguity and the quality of the 3D corner observations. This is illustrated in figure 13. Furthermore, the

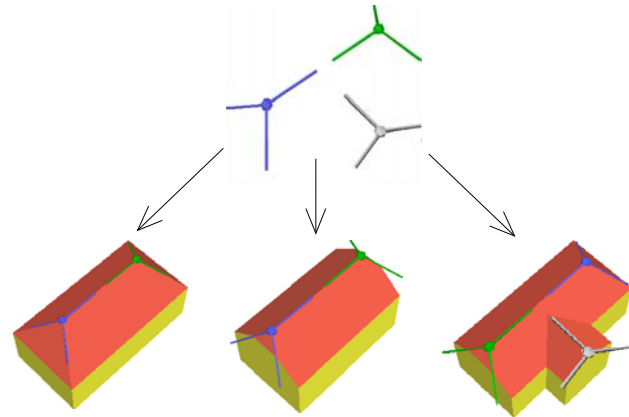


Figure 13: Generated building hypotheses for three reconstructed roof corners. The corner on the bottom right does not fit in completely with the T-shaped building on the right. Thus, two more hypotheses were generated neglecting the corner.

3D hypotheses are projected into the different possible 2D building views. Since free parameters may remain in the hypotheses (e.g. the building width and height in figure 13), the verification has to accomplish two tasks: 1) From the different hypotheses select the one which has most support in the image resp. reject all hypotheses, if there cannot be found enough evidence for any of them in the image. 2) Determine the actual geometric extent of the 2D building views in each image and estimate the unknown 3D building parameters.

For the identification of the appropriate building view the following circumstances have to be considered:

- 1.) Due to occlusions and noise it is likely that not every part of a building view can be observed in the image. For this reason, the verification will be done on the lowest level of our building model hierarchy. Building views are decomposed into point, line, and region features.
- 2.) Because of possibly free parameters of the building views, the absolute coordinates and geometric properties of image features (like e.g. line lengths, area of regions) are not fixed in general. Above, due to noise, shadows and low contrast the extracted image features often are fragmented into smaller parts.

Therefore, no simple feature based matching, comparing the attributes of building view and image features, can be done. Geometrical and topological relations, however, are stable wrt. free building parameters. These relations are also more robust in case of disturbances (for example, fragmented lines remain parallel). Therefore, the identification of an instance of a building view in the image is done using relational matching (cf. [HS93, Vos92]).

3.) The selection of the best fitting hypothesis resp. the rejection of all hypotheses should be done in a consistent and conclusive way. Therefore, the employed evaluation function for measuring the similarity between a building view and the image data is based on probability theory, which also provides an objective interpretation of the reconstruction results.

4.) The evaluation function has to take into account that building hypotheses both of the same and of different complexity have to be compared (see figure 13). This is achieved by incorporating the complexity of the building views into the evaluation function using the Minimum Description Length Principle (MDL, cf. [Ris87]). The application of the MDL principle requires optimally coded hypotheses. As an approximation, a compact encoding scheme for the relational representation of arbitrary building views had to be developed.

The verification process is done in two steps: 1.) For every building view find the best matching to the extracted image features (maximum likelihood estimation). Reject every hypothesis for which too little evidence was found in the image data. 2.) From all remaining hypotheses determine the most probable one (maximum a posteriori classification) and finally, estimate the building parameters.

In the following, we first show which types of relations are used for the relational representation of building views. Then we describe the evaluation function for measuring the similarity of relational structures and explain how the needed probabilities are acquired. Finally, it is shown how the evaluation function is extended to realize the MDL criterion.

6.1 Relational Representation of Building Views

The relational representation for 2D building views is defined by a set of relations. The selection of the appropriate relations is of utmost importance, because of the following two considerations:

1.) The search for an instance of a building view is done by finding the image structures having the maximum relational similarity with the building view. For this task, the set of relations representing a building view has to be specific enough to discriminate the image features that correspond to a building view from image features of the background.

2.) Different building views must have different relational representations to allow unambiguous classification. This becomes particularly important for resembling building views.

Furthermore, each type of relation has to satisfy the following condition: As explained above, the geometry of 2D building views may remain unfixed. The relational representation of a building view, however, should abstract from this variability and must remain constant for every possible geometrical extent of this building view. Therefore, the employed relations must be stable within all possible geometric instantiations of a given building view.

We have found out, that the following set of relations meets these requirements (cf. [Kol99]):

Feature Adjacency: The feature adjacency relation states that two image features have to be neighbored. Due to the applied image model (cf. section 4.1) only features of different type are assumed to be neighbored, namely (point,line), (point,region), and (line,region).

Line Parallelism: states that two line segments have to be parallel. It is realized by computing the angle difference between both lines and the test whether it is smaller than a given threshold value.

Collinearity: This relation defines that a line and a point feature will be considered collinear, iff the distance between the point and the straight line going through the line segment is below a given threshold.

Lines on the same side: The Same_Side_Lines (SSL) relation constrains two line segments to lie in the same halfplane that is defined by the straight line going through a third line segment. Because this is a rather unusual relation, it is illustrated in figure 14.

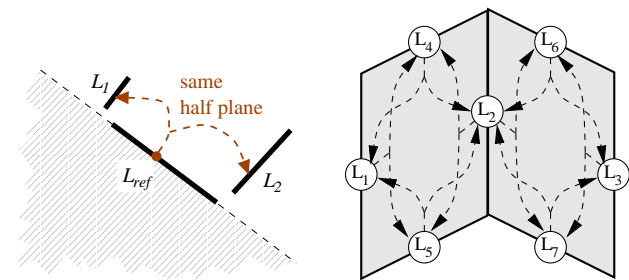


Figure 14: Definition of the (ternary) Same_Side_Line relation. Left: Three line segments L_1 , L_2 and L_{ref} satisfy the Same_Side_Lines relation, iff L_1 and L_2 lie in the same half plane that is spanned by the line running through L_{ref} . Right: All Same_Side_Lines relations of the top view of a saddle roof (L_1, \dots, L_7 denote the lines).

Distinctness of Features: This relation states that two different building view features have to be assigned different extracted image features. This relation is generated for all pairs of features of a building view.

In order to search for an instance of a building view in the image, each view is transformed to its relational description. Since building hypotheses are generated at runtime, the relational representation for each building view is derived dynamically, too. For a more detailed explanation refer to [Kol98, Kol99].

6.2 Relational Matching of a Building View with the Extracted Image Features

Finding a relational matching between a building view and the extracted image features amounts to the search for an assignment of appropriate image features to the features of the building view, such that the image features satisfy the same relations that are given between the building view's features.

This task defines a combinatorial problem and is computationally hard to solve (see [HS79]). Thus, efficient search techniques have to be applied to find the best matching. We employ constraint solving techniques — but with a special extension: Standard constraint solving methods [Mes89] demand that every object part is observable and extracted by the image segmentation, and that every constraint can be satisfied. They fail, if only one of these conditions does not hold. Due to occlusions, noise and segmentation errors this happens rather often. Therefore, the standard constraint techniques were extended to allow the explicit handling of unobserved features and constraint violations. These operational aspects are explained in detail in [Kol98, KPC99]. In the following, we will focus on the description of the evaluation function, being used to measure the similarity of an inexact matching between a building view hypothesis and the extracted image features.

In literature several evaluation functions for measuring the similarity resp. difference between two relational descriptions have been proposed (see [SH85, BK88, Vos92]). Following the line of Boyer and Kak [BK88] and Vosselman [Vos92] we employ an evaluation function that is based on information theory. Relational matching can be regarded as a communication problem, where the first relational description D_1 is transmitted over a discrete, noisy communication channel and is received as a somewhat distorted description D_2 . The similarity between the transmitted structure D_1 and the received structure D_2 is measured by the mutual information $I(D_1 ; D_2)$ of both structures. The mutual information is a measure for the true information content that is transmitted over the noisy channel (cf. [Ing71]).

Since a matching h assigns image features (on the transmitter side) to features of the building view (on the receiver side), it defines a channel. Vosselman showed that finding the assignment \hat{h} (the best channel) with the maximum mutual information is equal to maximizing $P(h|D_1, D_2)$, which means that \hat{h} is

the most likely matching for the given relational structures.

Building views are represented by sets of relations. Therefore, the mutual information between the building view and the image features is defined by the sum of the mutual information of every single relation:

$$I_h(Hypoth. ; Image) = \sum_{r_m \in Hyp.} I_h(r_m ; h(r_m)) \quad (3)$$

$h(r_m)$ means relation $r_m(h(a_1), \dots, h(a_n))$ between the image features that are assigned by the matching h . a_1, \dots, a_n denote the features of the building view that are related by r_m . In the following $h(r_m)$ will be referred to as r_i , denoting that this is a relation between extracted image features. Using this notation, the mutual information for a relation on features of the building view and the image is defined as

$$I(r_i ; r_m) = \log_2 \frac{P(r_i|r_m)}{P(r_i)} \quad [\text{Bit}]. \quad (4)$$

The conditional probability $P(r_i|r_m)$ models the characteristics of the observation process. For instance, probability $P(r_i = true|r_m = true)$ expresses the chances that if relation r_m is true in the building view it can also be observed between the corresponding features in the image. Actually, we estimate these probabilities from manually trained matchings. The a priori probability $P(r_i)$ for observing image relation r_i can be derived from the a priori probability $P(r_m)$ by applying Jeffrey's rule of total probability, getting

$$I(r_i ; r_m) = \log_2 \frac{P(r_i|r_m)}{\sum_{r'_m} P(r_i|r'_m) \cdot P(r'_m)} \quad [\text{Bit}]. \quad (5)$$

Since every relation can only be true or false, the a priori probability $P(r_m)$ has the two values $P(r_m = true)$ and $P(r_m = false)$. Hence, the conditional probability $P(r_i|r_m)$ is given by a 2×2 table. Tables 1 and 2 show an example for the computation of the mutual information.

r_m	$P(r_m)$	$P(r_i r_m)$	$r_m = true$	$r_m = false$
<i>true</i>	0.17	$r_i = true$	0.67	0.01
<i>false</i>	0.83	$r_i = false$	0.33	0.99

Table 1: Example for a priori probabilities for building view relation r_m (left) and conditional probabilities of image relation r_i wrt. building view relation r_m (right).

$I(r_i ; r_m)$	$r_m = true$	$r_m = false$
$r_i = true$	2.45	-3.61
$r_i = false$	-1.41	0.17

Table 2: Resulting mutual information calculated from the probabilities shown in table 1 using equation 5.

From this example one can see that if the relation is true in the building view and also holds for the corresponding image features, the matching will be supported by 2.45 bit. Otherwise, if the relation cannot be observed on the assigned image features, the relation contradicts the matching by 1.41 bit. In case

that one incident feature cannot be observed in the image the relation is rated 0, because then it is undefined and cannot support or contradict a matching.

In the following two sections it is described how the a priori probabilities $P(r_m)$ and the conditional probabilities $P(r_i|r_m)$ are derived. Because of the different likely visibility of roof parts and walls, relations are subdivided into roof and wall relations having their own probability distributions. A roof feature is either a roof region or a line resp. a point that is incident to a roof region. The information, which regions are roof regions, is derived from the building model. Every relation that is incident exclusively to roof features belongs to the set of roof relations, otherwise to the set of wall relations. For the sake of simplicity this distinction is omitted in the following.

Computation of the a priori probabilities. $P(r_m)$ expresses the probability that a relation between its incident features is true for a given building view. It is computed for each relation type from the ratio of the number $|R_m^+|$ of existing relations $r_m \in R_m^+$ and the number of possible relations $|R_m|$ of the building view:

$$\begin{aligned} P(r_m = true) &= \frac{|R_m^+|}{|R_m|} \\ P(r_m = false) &= 1 - P(r_m = true). \end{aligned} \quad (6)$$

The number of possible relations is bounded by the cardinality of the relation type. For a given set of model features M the cardinality of an n -ary relation R_m over M is in the general case $|R_m| = |M|^n$, because every feature can be in relation to every other (including itself). The number of possible relations reduces, if the relation is symmetric or reflexive. For example, if one line L_1 is parallel to another line L_2 , $|(L_1, L_2)|$ represents the same relation as $|(L_2, L_1)|$. Table 3 shows the properties and cardinalities of the different employed relations.

Relation R_m	Arity	Properties	Cardinality $ R_m $
FAG(Line,Point)	2	irrefl., asymm.	$ L \cdot P $
FAG(Region,Line)	2	irrefl., asymm.	$ F \cdot L $
FAG(Region,Point)	2	irrefl., asymm.	$ F \cdot P $
Line_parallelism	2	refl., sym., transitive	$\frac{ L (L -1)}{2}$
Collinearity	2	irrefl., asymm.	$ L \cdot P $
Same_Side_Lines	3	irrefl., arg. 2 & 3 sym.	$\frac{ L (L -1)(L -2)}{2}$
\neq (Points)	2	irrefl., symm.	$\frac{ P (P -1)}{2}$
\neq (Lines)	2	irrefl., symm.	$\frac{ L (L -1)}{2}$
\neq (Regions)	2	irrefl., symm.	$\frac{ F (F -1)}{2}$

Table 3: Properties and cardinalities of the employed relations. $|P|$, $|L|$ and $|F|$ denote the number of points, lines, and regions of the building view. FAG means the feature adjacency relation and \neq the distinctness relation.

Training of the conditional probabilities. The conditional probability $P(r_i|r_m)$ expresses the chances that a model relation r_m can be observed between corresponding features in the image. It is a measure of the stability resp. robustness for each type of relations wrt. to noise and disturbing effects of the image segmentation.

The probabilities are estimated from training matches by building the ratios of the number of relations $|R_i^+|$ that hold in the image and the number of given relations in the corresponding building views $|R_m^+|$, where $|R_i^+| \leq |R_m^+|$. For instance, $P(r_i = true|r_m = true)$ is estimated by

$$P(r_i = true|r_m = true) = \frac{|R_i^+|}{|R_m^+|} \quad (7)$$

Since relations are only distinguished into true or false, $P(r_i|r_m)$ is given as a 2×2 -matrix for every relation type (cf. table 1).

6.3 Finding the Most Probable Model

In the previous section it was shown how the most likely matching between one building view and the extracted image features is determined. The maximization of the mutual information always results in a matching — independent of the true resemblance of the hypothesis and the image data. Therefore, a statistical test is defined on the expected amount of information needed for a reliable matching, ruling out all definitely wrong hypotheses (for details see [Kol99, Vos92]).

To conclude the verification step, it has to be determined which of the remaining building views will be the most probable one. This is problematical, because in general the building views have not the same complexity and thus may be represented by a different number of relations. The maximum achievable mutual information between the building views and the image data increases with the complexity of a hypothesis. Therefore, the matching scores (maximum mutual information) of different hypotheses are not directly comparable.

To overcome this difficulty we employ the Minimum Description Length Principle (MDL) which was introduced by Rissanen in [Ris87]. The MDL criterion balances between the goodness of fit between hypothesis and data on the one hand and the complexity of the hypothesis on the other hand. In [Kol99] we show that the matching scores between building views and image data can be normalized, if from each score the complexity of the corresponding building view is subtracted. The complexity of a building view is measured by the size of the bit string that is needed to fully encode the view. If the encoding scheme is optimal in the sense that it achieves the minimum length, due to the MDL principle the hypothesis with the highest score after subtraction of its coding length is the most probable explanation of the data.

The mutual information between building view and image data is defined by the mutual information of the relations. Thus, a compact coding scheme for the relational representation of building views is needed as an approximation for the optimal encoding.

Coding of relational building views. Relational structures can be represented by (hyper) graphs. Instead of using standard data structures for the encoding of graphs like adjacency matrices and adjacency lists (cf. [AHU87]), we apply a more efficient encoding scheme that is inspired by the work of Cook and Holder from the field of machine learning about automatic substructure discovery [CH94]. Permutation series are used to encode the corresponding adjacency matrix of a given graph. Each row of the adjacency matrix can be seen as a bit string of fixed length n . When all rows are appended, we get one string of length n^2 consisting of k 1's and $n^2 - k$ 0's. Since there exist exactly $\binom{n^2}{k}$ different possibilities to place the k 1's on the bit string, and these combinations may be enumerated by an algorithm, we simply store the number of 1's and the number of the permutation to represent the graph. Thus, the coding length is given by

$$L(\text{Graph}) = \log_2(n + 1) + \log_2 \binom{n^2}{k} \quad [\text{Bit}]. \quad (8)$$

Graphs can be encoded even more compact if the underlying relations possess structural properties like symmetry, reflexivity, and transitivity (cf. table 3 on page 15). In case of a symmetrical relation the adjacency matrix of the corresponding graph is mirrored at the main diagonal. Therefore, it is sufficient to encode only the entries on one side of the diagonal. For transitive relations, adjacent nodes in the corresponding graph form cliques, that can be sparsed to minimal spanning trees, reducing the number of 1's. To restore the original structure, one only has to compute the transitive closure of the encoded graph.

For the encoding of the relational representation of building views not only the structural properties of the different relations have to be considered, but also the interdependencies of the relations. For instance, the feature adjacency relation between regions and points can be derived from the feature adjacency relations between regions and lines in conjunction with the feature adjacency relation between lines and points. Thus, the former must not be included in the calculation of the overall coding length.

Furthermore, only structures that lead to codes with variable length have to be considered. Model information with a constant coding length such as the number of points, lines, and regions of a building view are the same for all different building views and can be neglected in the classification. Table 4 shows the different types of coding lengths that are taken into

account for the computation of the total code length of building views. A more detailed explanation of the coding scheme is given in [Kol99].


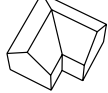
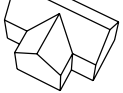
			
No. of points $ P $	9	14	19
No. of lines $ L $	12	21	27
No. of regions $ F $	4	8	9
No. of relations			
$ FAG(\text{region, line}) $	17	34	42
$ FAG(\text{line, point}) $	24	42	54
$ Line_parallelism $	11	30	53
$ Same_Side_Lines $	17	34	38
$\Omega_{FAG(\text{region, line})}$	46.00	123.00	160.14
$\Omega_{FAG(\text{line, point})}$	76.08	159.91	229.39
Ω_{parallel}	35.58	70.99	107.50
Ω_{SSL}	4.09	5.09	22.16
$\Omega_{\text{roof regions}}$	2.00	3.00	3.17
Total coding length Ω	163.75	361.99	522.36

Table 4: Coding lengths for three building views of different complexity seen from the same perspective. All coding lengths are given in Bit. As expected, the coding length increases with the complexity of the building view.

Example. Figure 15 on the next page shows an example for the relational matching and classification applied to a portion of an aerial image showing a L-shaped hip roof house. Three hypotheses were generated (simple, L-shaped, and T-shaped hip roof house). The mutual information and the assigned image features of the best matchings for all three building views are shown. As expected, the mutual information increase as the building views become more complex. On the bottom row one can see that the subtraction of the coding length correctly normalises the matching scores. The minimal distance between the highest total score and the others is about 40 bits, indicating a clear decision for the (correct) L-shaped hip roof building hypothesis.

6.4 Final Parameter Estimation

A building hypothesis is successfully verified, when not all generated hypotheses were rejected. In this case the most probable building view is selected from each image. The determined matchings assign extracted image features to the features of the building views, fixing the possibly unknown geometric extents of the 2D views. In order to estimate the parameters of the underlying 3D building hypothesis, the observations (assigned extracted image features) for each image are inserted into the building view specific projection equations, resulting in one nonlinear equation system. Finally, the 3D building parameters are estimated simultaneously using the same procedure as described in section 5.3 on page 10.


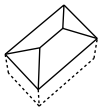
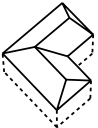
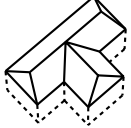
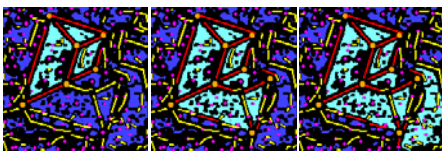
Image	Hypotheses (2D building views)		
			
best matching \hat{h}			
$I_{\hat{h}}(\text{Hypot.}; \text{Image})$	85.2	224.24	237.56
Ω	107.31	206.09	325.19
$I_{\hat{h}} - \Omega$	-22.11	18.15	-87.63

Figure 15: Identification of the most probable model for the given image. Each column shows a building hypothesis, the extracted image features (highlighted wrt. the best matching), the mutual information of the best matching, the coding length of the building view, and the final rating. As can be seen from the bottom row, even from very noisy data the L-shaped hypothesis correctly is selected.

7 CONCLUSIONS

In this paper we have shown how complex buildings are automatically reconstructed from aerial images using a component-based building model. The reconstructed buildings are represented by parameterized components, which beside the determined geometric shape and location contain also the information about the building type and the corresponding parameters.

The hierarchical structure of the building model reflects the four levels of semantic abstraction. It enables the stepwise interpretation of the initially extracted image features. The coherent modeling in 2D and 3D allows the verification of hypotheses at any aggregation level.

Building specific knowledge is used in all stages of the analysis to reduce the search space for hypothesis generation, to rate the matchings between data and model, and to increase the accuracy of the geometric reconstruction. The most important benefits wrt. the three main steps of the reconstruction process are:

- The transition from image space to object space is restricted to building corner types. Only corners that actually appear as a part of an existing building part primitive will be reconstructed.
- The construction of aggregates is restricted to sensible building structures and sizes by the building specific aggregation relations resp. parameter constraints.
- Building specific constraints of generated building hypotheses are propagated down to the lowest level

of our hierarchy and are used to perform the relational matching of building view features with the image features during the final verification. The knowledge about the interdependence of different relation types is taken into account to accomplish a compact encoding of relational representations of building hypotheses, thus making the application of the MDL principle for classification feasible.

The presented concept has been implemented and shows promising results in suburban areas. It was successfully applied to the international data set which was distributed from the ETH Zürich for the Ascona Workshop 1995 on *Automatic Extraction of Man-Made Objects from Aerial and Space Imagery* (cf. [GKA95, MBS94]). All buildings in the images could be reconstructed. In contrast to other approaches, the type and the specific parameters like height, roof height, width and length even of the complex buildings were determined. The detailed results are given in [FKL⁺98].

ACKNOWLEDGEMENTS

This work was done within the project "Semantic Modeling and Extraction of Spatial Objects from Images and Maps", especially in the subproject "Building Extraction" funded by the German Research Council (DFG). Thanks go to Wolfgang Förstner, Lutz Plümer, Volker Steinhage, and Armin B. Cremers who helped with many discussions and critical comments to develop these ideas. We thank the DFG for supporting our work.

REFERENCES

- A. Aho, J. E. Hopcroft, and J. D. Ullman. *Data Structures and Algorithms*. Addison-Wesley, London, 1987.
- F. Bignone, O. Henricsson, P. Fua, and M. Stricker. Automatic Extraction of Generic House Roofs from High Resolution Aerial Images. In *Computer Vision — Proceedings of ECCV '96*, Lecture Notes in Computer Science No. 1064, pages 85–96. Springer-Verlag, Berlin, 1996.
- K. L. Boyer and A. C. Kak. Structural Stereopsis for 3D-Vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10(2):144–166, March 1988.
- C. Braun, T. H. Kolbe, F. Lang, W. Schickler, V. Steinhage, A. B. Cremers, W. Förstner, and L. Plümer. Models for Photogrammetric Building Reconstruction. *Computer & Graphics*, 19(1):109–118, 1995.

- D. J. Cook and L. B. Holder. Substructure Discovery Using Minimum Description Length and Background Knowledge. *Journal of Artificial Intelligence Research*, 1:231–255, 1994.
- S. J. Dickinson, A. P. Pentland, and A. Rosenfeld. 3-D Shape Recovery Using Distributed Aspect Matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):174–198, 1992.
- R. Englert and E. Gülch. A One-Eye Stereo System for the Acquisition of Complex 3D-Building Structures. *GEO-INFORMATION-SYSTEMS, Journal for Spatial Information and Decision Making*, 9(4):16–21, August 1996.
- R. Englert. Systematic Acquisition of Generic 3D Building Model Knowledge. In W. Förstner and L. Plümer, editors, *Semantic Modeling for the Acquisition of Topographic Information from Images and Maps*, pages 181–195. Birkhäuser Verlag, Basel, Switzerland, 1997.
- P. Fua and A.J. Hanson. Resegmentation using generic shape locating general cultural objects. *Pattern Recognition Letters*, 5:243–252, 1987.
- A. Fischer, T. H. Kolbe, and F. Lang. Integration of 2D and 3D Reasoning for Building Reconstruction Using a Generic Hierarchical Model. In W. Förstner and L. Plümer, editors, *Semantic Modeling for the Acquisition of Topographic Information from Images and Maps*, pages 159–180. Birkhäuser Verlag, Basel, Switzerland, 1997.
- A. Fischer, T. H. Kolbe, F. Lang, A. B. Cremers, W. Förstner, L. Plümer, and V. Steinhage. Extracting Buildings from Aerial Images using Hierarchical Aggregation in 2D and 3D. *Computer Vision & Image Understanding*, 72(2), 1998.
- C. Fuchs, F. Lang, and W. Förstner. On the Noise and Scale Behaviour of Relational Descriptions. In Ebner, Heipke, and Eder, editors, *ISPRS*, volume 30, pages 256–267. SPIE, 1994.
- W. Förstner and L. Plümer, editors. *Semantic Modeling for the Acquisition of Topographic Information from Images and Maps*. Birkhäuser Verlag, Basel, Switzerland, 1997.
- C. Fuchs. *Extraktion polymorpher Bildstrukturen und ihre topologische und geometrische Gruppierung*. PhD thesis, Deutsche Geodätische Kommission, München, Vol. C 502, 1998.
- A. Grün, E. P. Baltsavias, and O. Henricsson, editors. *Automatic Extraction of Man-Made Objects from Aerial and Space Images (II)*. Birkhäuser, Basel, 1997.
- A. Grün, O. Kübler, and P. Agouris, editors. *Automatic Extraction of Man-Made Objects from Aerial and Space Images*. Birkhäuser, Basel, 1995.
- E. Gülch. A Knowledge-based Approach to Reconstruct Buildings in Digital Aerial Imagery. In L. W. Fritz and J. R. Lucas, editors, *International Archives of Photogrammetry and Remote Sensing*, volume 29, B2, pages 410–417, 1992.
- O. Henricsson and E. P. Baltsavias. 3-D Building Reconstruction with ARUBA: A Qualitative and Quantitative Evaluation. In A. Grün, E. P. Baltsavias, and O. Henricsson, editors, *Automatic Extraction of Man-Made Objects from Aerial and Space Images (II)*, pages 139–148. Birkhäuser, Basel, 1997.
- S. Heuel, F. Lang, and W. Förstner. Topological and geometrical reasoning in 3d grouping for reconstructing polyhedral surfaces. Internal report, Institute for Photogrammetry, Bonn University, 1999.
- R. M. Haralick and L. G. Shapiro. The Consistent Labeling Problem, Part I. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1:173–184, 1979.
- R. M. Haralick and L. G. Shapiro. *Computer and Robot Vision*, volume II. Addison-Wesley Publishing Company, New York, 1993.
- M. Hendrickx, J. Vandekerckhove, D. Frère, T. Moons, and L. Van Gool. On the 3D Reconstruction of House Roofs from Aerial Images of Urban Areas. In *International Archives of Photogrammetry and Remote Sensing*, volume 32, Part 3-4W2, pages 87–96. ISPRS, 1997.
- F. M. Ingels. *Information and Coding Theory*. Intext Educational Publishers, San Francisco, Toronto, London, 1971.
- C. Jaynes, A. Hanson, and E. Riseman. Model-Based Surface Recovery of Buildings in Optical and Range Images. In W. Förstner and L. Plümer, editors, *Semantic Modeling for the Acquisition of Topographic Information from Images and Maps*, pages 211–227. Birkhäuser Verlag, Basel, Switzerland, 1997.
- T. H. Kolbe. Constraints for Object Recognition in Aerial Images — Handling of Unobserved Features. In M. Maher and J.-F. Puget, editors, *Principles and Practice of Constraint Programming – Proceedings of CP'98*, Lecture Notes in Computer Science No. 1520. Springer-Verlag, Berlin, 1998.
- T. H. Kolbe. *Identifikation und Rekonstruktion von Gebäuden in Luftbildern mittels unscharfer Constraints*. PhD thesis, Institut für Umweltwissenschaften, Hochschule Vechta, 1999.
- T. H. Kolbe, L. Plümer, and A. B. Cremers. Identification of Buildings in Aerial Images Using Constraint

- Relaxation and Variable Elimination. *IEEE Intelligent Systems*, to appear, 1999.
- F. Lang. *Geometrische und semantische Rekonstruktion von Gebäuden durch Ableitung von 3D-Gebäudeecken*. PhD thesis, Institut für Photogrammetrie, Universität Bonn, 1999.
- C. Lin, A. Huertas, and R. Nevatia. Detection of Buildings from Monocular Images. In A. Grün, O. Kübler, and P. Agouris, editors, *Automatic Extraction of Man-Made Objects from Aerial and Space Images*, pages 125–134. Birkhäuser, Basel, 1995.
- F. Lang, T. Löcherbach, and W. Schickler. A one-eye stereo system for semi-automatic 3d-building extraction. *Geomatics Info Magazine*, 1995.
- S. Mason, E. Baltsavias, and D. Stallman. High precision photogrammetric data set for building reconstruction and terrain modelling. Internal report, Institute of Geodesy and Photogrammetry, ETH Zürich, 1994.
- P. Meseguer. Constraint Satisfaction Problems: An Overview. *AICOM*, 2(1):3–17, 1989.
- R. Mohan and R. Nevatia. Perceptual Grouping for the Detection and Description of Structures in Aerial Images. In *Proceedings of the Image Understanding Workshop 1988*, 1988.
- R. Mohan. *Perceptual Organization for Computer Vision*. PhD thesis, Institute for Robotics and Intelligent Systems, University of Southern California, 1989.
- J.C. McGlone and J.A. Shufelt. Projective and object space geometry for monocular building extraction. In *Proceedings Computer Vision and Pattern Recognition*, pages 54–61, 1994.
- R. Nevatia, C. Lin, and A. Huertas. A System for Building Detection from Aerial Images. In A. Grün, E. P. Baltsavias, and O. Henricsson, editors, *Automatic Extraction of Man-Made Objects from Aerial and Space Images (II)*, pages 77–86. Birkhäuser, Basel, 1997.
- R. Nevatia and K. E. Price. Locating Structures in Aerial Images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 4(5):476–484, 1982.
- J. Rissanen. Minimum-Description-Length Principle. *Encyclopedia of Statistical Sciences*, 5:523–527, 1987.
- K. Schutte. Recognition of buildings from aerial images. In J. A. C. Bernsen, J. J. Gerbrands, A. A. Hoeve, A. W. M. Smeulders, M. A. Viergever, and A. M. Vossepoel, editors, *3rd Quinquennial Review 1991-1996 Dutch Society for Pattern Recognition and Image Processing*, pages 211–225. NVPBV, Delft, 1996.
- L. G. Shapiro and R. M. Haralick. A Metric for Comparing Relational Descriptions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 7(1):90–94, 1985.
- L. Spreeuwers, K. Schutte, and Z. Houkes. A model driven approach to extract buildings from multi-view aerial imagery. In A. Gruen, E. P. Baltsavias, and O. Henricsson, editors, *Automatic Extraction of Man-Made Objects from Aerial and Space Images (II)*, pages 109–118. Birkhäuser Verlag, Basel, 1997.
- V. Steinhage. On the Integration of Object Modeling and Image Modeling in Automated Building Extraction from Aerial Images. In A. Grün, E. P. Baltsavias, and O. Henricsson, editors, *Automatic Extraction of Man-Made Objects from Aerial and Space Images (II)*, pages 139–148. Birkhäuser, Basel, 1997.
- V. Steinhage. Zur automatischen Gebäuderekonstruktion aus Luftbildern. Habilitationsschrift, Institut für Informatik, Universität Bonn, 1998.
- C.J. Taylor, P.E. Debevec, and J. Malik. Reconstructing polyhedral models of architectural scenes from photographs. In B. Buxton B. and R. Cipolla, editors, *Computer Vision - ECCV '96, Vol. II, Proc. of the 4th Europ. Conf. on Computer Vision*, pages 659–668. Lecture Notes in Computer Science, 1065, Springer-Verlag, 1996.
- G. Vosselman. *Relational Matching*. Lecture Notes in Computer Science No. 628. Springer Verlag, Berlin, 1992.
- U. Weidner. Digital Surface Models for Building Extraction. In Grün et al. [GBH97], pages 193–202.
- U. Weidner and W. Förstner. Towards Automatic Building Extraction from High Resolution Digital Elevation Models. *ISPRS Journal of Photogrammetry and Remote Sensing*, 50(4):38–49, 1995.