COMPUTER VISION AND IMAGE UNDERSTANDING
Vol. 72, No. 2, November, pp. 185–203, 1998
ARTICLE NO. IV980721

# Extracting Buildings from Aerial Images Using Hierarchical Aggregation in 2D and 3D

André Fischer

*Computer Science Department I, University of Bonn, Römerstraße 164, 53117 Bonn, Germany*

Thomas H. Kolbe

*Institute for Environmental Sciences, University of Vechta, P.O. Box 1553, 49364 Vechta, Germany*

Felicitas Lang

*Institute for Photogrammetry, University of Bonn, Nußallee 15, 53115 Bonn, Germany*

Armin B. Cremers

*Computer Science Department III, University of Bonn, Römerstraße 164, 53117 Bonn, Germany*

Wolfgang Förstner

*Institute for Photogrammetry, University of Bonn, Nußallee 15, 53115 Bonn, Germany*

Lutz Plümer

*Institute for Environmental Sciences, University of Vechta, P.O. Box 1553, 49364 Vechta, Germany*

and

Volker Steinhage

*Computer Science Department I, University of Bonn, Römerstraße 164, 53117 Bonn, Germany*

E-mail: steinhag@informatik.uni-bonn.de

Received May 28, 1997; accepted July 7, 1998

**We propose a model-based approach to automated 3D extraction of buildings from aerial images. We focus on a reconstruction strategy that is not restricted to a small class of buildings. Therefore, we employ a generic modeling approach which relies on the well-defined combination of building part models. Building parts are classified by their roof type. Starting from low-level image features we combine data-driven and model-driven processes within a multilevel aggregation hierarchy, thereby using a tight coupling of 2D image and 3D object modeling and processing, ending up in complex 3D building estimations of shape and location. Due to the explicit representation of well-defined processing states in terms of model-based 2D and 3D descriptions at all levels of modeling and data aggregation, our approach reveals a great potential for reliable building extraction.** © 1998 Academic Press

*Key Words:* **explicit 2D modeling; coupling of 2D and 3D modeling; multilayer aggregation; building modeling; multiimage correspondence analysis; midlevel feature aggregates; aspect hierarchies, constraint logic programming.**

## 1. INTRODUCTION

Due to the fact that more than about 50% of the world population lives in urban or suburban environments the automation of 3D building extraction is an issue of great importance and shows an increasing need for various applications including geoinformation systems, town planning, or environmental related investigations.

Aerial images contain on the one hand a certain amount of information not relevant for the given task of building extraction like vegetation, cars, and building details. On the other hand, there is a loss of relevant information due to occlusions, low contrast, or disadvantageous perspective. To compensate for these properties of image data as well as for being able to handle the complexity of building types and building structures, a promising concept of automated building extraction from aerial images must incorporate a sufficiently complete model of the objects of interest and their relations within the whole process of image interpretation and object reconstruction [63].



1077-3142/98 $25.00

We propose a model-based approach to automated 3D extraction of buildings from aerial images. The knowledge about buildings is used to control and to evaluate building extraction in all stages of the process. It is encoded by means of a generic 3D object model, which describes spatial appearances of buildings and characterizations of their components. Furthermore, a 2D image model, which is capable of integrating sensor and illumination modeling, describes the projective appearances of buildings specific for the given aerial imagery.

### 1.1. Related Work

Related work on 3D building extraction—or in general on 3D scene reconstruction—reveals different modeling schemes. Polyhedral models show a long tradition as approximative object descriptions [9, 34, 36, 40, 64, 68]. Obviously, polyhedral descriptions are too general for the use within 3D building extraction and therefore move the burden of building modeling to additional representation schemes to represent and organize domain-specific heuristics and constraints, like in the MOSAIC system [33] or in the approach of Braun [5]. Parameterized models are restricted to describe the most common building types in the sense of prototypes [44, 46–48, 50, 54, 56] but represent few variations and combinations of their shapes as well as other relations. Prismatic models can describe arbitrary complex polygonal ground plans of buildings, but they reveal a strong restriction to buildings with only flat roofs [69, 70, 58]. CAD models are used to describe objects with fixed geometry and topology in object *recognition* tasks, especially for controlling industrial processes [16, 27, 37, 53]. The use of CAD models in building extraction is therefore restricted to the identification of a priori known buildings [35, 59, 60].

Generic modeling approaches promise on the one hand the greatest modeling power, but on the other hand demand effective constraints and heuristics to restrict modeling to building specific shapes. Fua and Hanson [21, 22] employ simple box-type primitives but propose an explicit representation of legal primitive combinations to more complex building aggregates. The approaches of Dickinson *et al*. [12] and of Bergevin and Levine [1] are of outstanding importance due to the integration of 3D generic object models and an explicit modeling of 2D projective object appearances within a recognition-by-components strategy [2]. Both approaches employ volumetric primitives instead of simple box types, but they neglect the description of elaborated schemes for domain dependent primitive combinations. Furthermore, the reliable extraction of their primitives from real images is the concern of current research [52, 55]. Bignone *et al*. [3] propose a generic roof model which assumes planar roof surfaces. The roof patches are extracted by combining photometric and chromatic attributes of image regions with spatial edge geometry. The 3D patches 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. Frère *et al*. [30] adopt this modeling approach, but they extract image regions with homogeneous photometric and chromatic properties by navigating through a constraint triangulation network, where each extracted line segment coincides with edges of the triangles. Henricsson *et al*. [32] present impressive results of their approach on some test data but show no explicit modeling of building types and building specific aggregation schemes. Groups of planar 3D patches that are optimized according to the criteria of simplicity, compactness, and completeness are not necessarily real roof shapes.

For a general and up-to-date overview on the topic of building reconstruction we highly recommend the proceedings of the two workshops on *Automatic Extraction of Man-Made Objects from Aerial and Space Images* [24, 25] and the proceedings of the workshop on *Semantic Modeling for the Acquisition of Topographic Information from Images and Maps* [19].

### 1.2. Overview

In earlier papers [6, 7], we proposed in detail the concepts and processes that must be taken into account for a sufficiently complete modeling framework for 3D building extraction. This modeling framework integrates interrelations between image data and model descriptions at different aggregation levels and in terms of corresponding 3D object and 2D projective object descriptions. In this paper, we present a strategy for a well-defined path from the unstructured image data to the model-based and highly structured 3D reconstruction of buildings.

The overall strategy follows the paradigm of hypotheses generation and verification and combines bottom-up (data-driven) and top-down (model-driven) processes. Domain knowledge constrains even the early stages of hypotheses generation due to an elaborated *part-of*-hierarchy of 3D building parts and their corresponding projective descriptions. The reconstruction process is already carried out for local 2D feature aggregates to allow an early domain specific classification as 3D local building feature aggregates. A step-wise and strongly model-driven aggregation process combines 3D local building feature aggregates to well-defined parameterized 3D building parts and then to more complex 3D building aggregates. The resulting complex 3D building hypotheses and their components are reprojected into the images to allow a component-based and robust hypothesis verification applying constraint solving techniques [42].

First results of our approach were presented at the SMATI workshop in Bonn [15]. This paper is a substantially extended version which describes our models and operations in more detail.

Due to the explicit representation of well-defined processing states in terms of model-based 2D and 3D descriptions at all levels of modeling and data aggregation our approach shows great potential for reliable building extraction.

## 2. CONCEPT

In this section, we present the proposed building model and discuss its implications on the developed strategy.

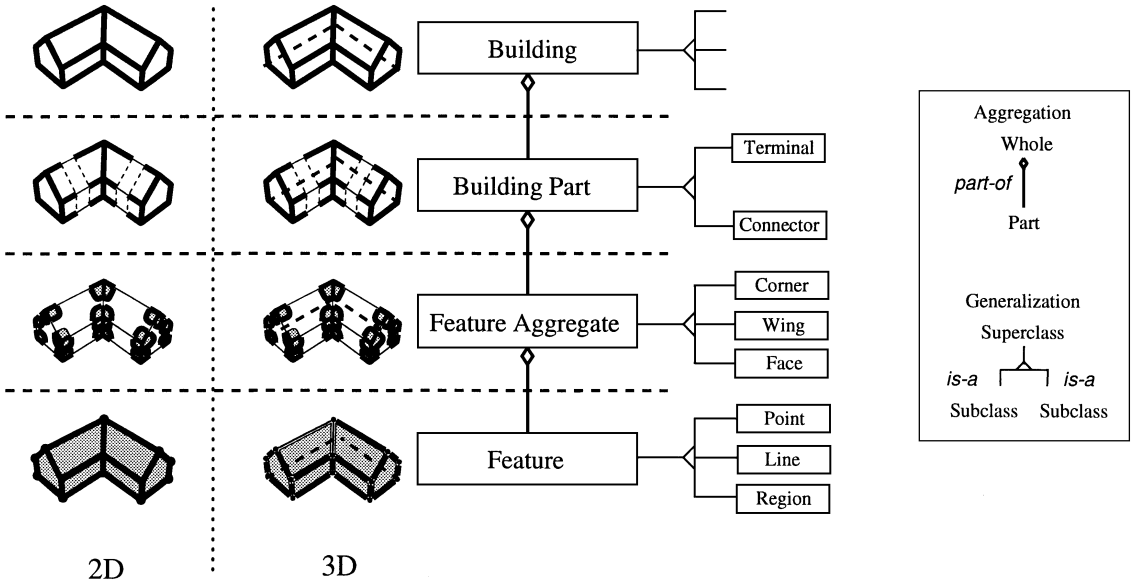



**FIG. 1.** Building model: The different semantic levels of the part-of hierarchy are shown in vertical direction, the different levels of abstraction of the is-a hierarchy in horizontal direction, which are only shown for the 3D-model. The 2D-image model describes the expected appearance of the building in the different levels of the part-of hierarchy, which is indicated by not showing the hidden lines.

### 2.1. Models

For coping with the complexity of natural scenes we propose an application-specific modeling of the domain *buildings*. Our general concept [6] contains a close interaction of bottom-up and top-down strategies within an aggregation hierarchy in 2D and 3D. In this aggregation hierarchy, the concept of building parts plays a fundamental role in our approach to a generic building model. On the one hand, this model is not based on a polyhedral scheme [32, 33], which proved to be too general in order to restrict reconstruction results to building specific shapes in terms of well-defined roof types and ground plans. On the other hand, our approach overcomes the limitations of parametrized CAD models due to the explicitly defined combinations of domain-specific volumetric and parametrized primitives in terms of building parts.

Another crucial aspect of the approach is the explicit representation of building appearances in aerial images in terms of an image model covering all levels of the aggregation hierarchy. This allows a tight coupling of 2D and 3D reasoning based on a coherent representation of object and image model. We start with the modeling of the 3D shape of buildings yielding the object model. The sensor model will transfer many of the concepts of the object model to the image model describing the expected appearance of the objects.

#### *2.1.1. Object Model*

Buildings reveal a high variability in structure, which suggests that one should represent them as an aggregation of several simple building parts. Furthermore, this modeling approach meets the problems of incomplete results of low-level feature extraction, caused by occlusions, low contrast, noise, and disturbances.

We therefore propose a multilevel *part-of hierarchy* (cf. Fig. 1). It reflects different levels of the envisaged semantic abstraction. The primitives of each aggregation level are specialized by an *is-a hierarchy* into subclasses.

Each primitive is described by its geometry, its domain specific role in terms of class memberships, and its relations to other primitives. The geometry is described by pose and shape parameters.

We currently employ four description levels for modeling complex buildings, which seems to be sufficient for a large class of buildings.

The first level (*feature level*) contains features $F$, namely attributed *points* $P$, *lines* $L$, and *regions* $R$. Attributes for lines and regions, for instance, are the orientation classifications horizontal (h), oblique (o), and vertical (v). Regions have an additional attribute describing its role: valid values among others are wall, roof, and floor. In general, the set of parameters is divided into positional parameters on the one hand, describing position and orientation, and on the other hand form parameters, like width, height, and length.

The second level (*feature aggregate level*) contains feature aggregates $A$, which are induced by points, lines, and regions and contain all their direct neighbors. Each aggregate is defined by a feature graph, given by a set $F = \{f_1, \ldots, f_k\}$ of features and adjacency relations $R \subset F \times F$. A *corner* $C$, for instance, contains one point and all its adjacent lines and regions (cf. Fig. 2).

The third level (*building part level*) contains building parts *BP*. Currently, we concentrate on the reconstruction of 3D corners. Therefore, the building part models are defined as corner

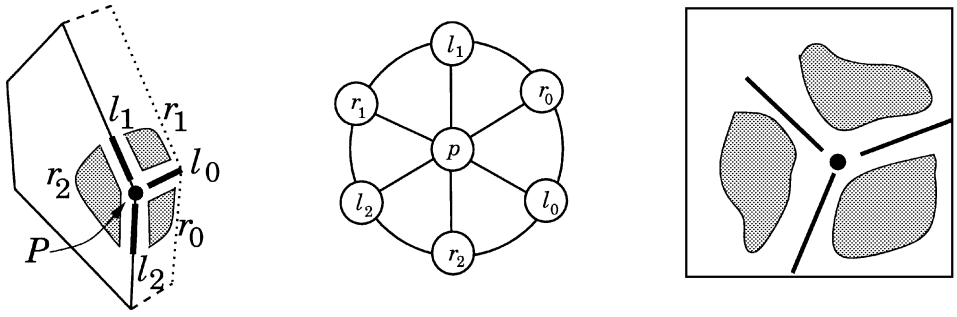



**FIG. 2.** A corner is a feature neighborhood of a point (left). Drawn as a graph, the arcs express the adjacency relation (center). Assuming no occlusions and disturbances, its expected appearance in the image reveals the same neighborhood relations between the corresponding 2D features (right).

graphs given by a set $C = \{c_1, \ldots, c_n\}$ of corners and adjacency relations $R \subset C \times C$. In future work we will describe the building part models as graph structures which will also integrate line and region induced feature aggregates (covered in the discussion in 2.2). Each building part model is also described as a parameterized volumetric object and has at least one so-called *plug face*, which is used for connecting building parts to each other. We discriminate *terminals* having exactly one plug face and *connectors* with two or more plug faces (cf. Fig. 3).

The fourth level (*building level*) contains complete buildings. Buildings are defined as graphs with building parts as nodes, the arcs representing pairs of building parts connected by corresponding plug faces. Thus, the most simple building consists of two connected terminals.

### *2.1.2. Image Model*

The 2D image model describes the expected appearance of the building at the same levels of aggregation as the corresponding 3D-structures.[1] This guarantees a maximum of coherence for representation and processing of 2D and 3D information.

The image model contains all properties that are invariant under projection and is taken to define constraints. This especially holds for all class memberships, neighborhood, and geometric relations, as far as they are not disturbed by self occlusions. These self occlusions can be predicted by an appropriate representation of the image model following the concept of an aspect-based scheme [4, 41]. For example, a 2D-corner, being a point-induced image aggregate, inherits the class membership of the corresponding 3D gable corner. The neighborhood relation between two faces of a roof in general can be expected to be transferred to a pair of image regions, whereas—assuming weak perspective—parallel 3D-roof lines map to parallel 2D-roof lines. Constraints of the higher aggregation levels are transferred to the primitives of the lower ones, if necessary.

Thus, the transformation from the object model to the image model needs at least the definition of an appropriate projective model, i.e., perspective or parallel projection, depending on the distance between the sensor and the objects of interest. In fact, to take into account all effects which contribute to the process of image formation and feature extraction, there is a need for models for sensor characteristics, illumination, and low-level feature extraction methods. The results, presented in this paper, are derived by using a pinhole camera as a sensor model and assuming weak perspective projection. The potential role of the integration and usage of an illumination model is discussed elsewhere [62] and will only be sketched in Section 3.3.

The image model is currently used for both the multiview reconstruction of 3D feature aggregates and the 2D verification of complete building hypotheses. But of course, an enhanced image model can be used at each aggregation level to utilize photometric attributes, verify intermediate aggregation results, and estimate certain parameters by monocular analysis, e.g., of shadows and wall projections [54].

[1] Actually, the image model contains the raster image as the lowest, say 0th level, from which the image features are extracted. This lowest level is not shown as we do not explicitly refer to it.

## 2.2. Strategy

Our input data is given as digital raster images with multiple overlap. Further information about the aerial image flight like exterior and interior camera orientation and time stamp are used. The starting point of our analysis is the extraction of a polymorphic image description consisting of points $P^{2D}$, lines $L^{2D}$, and regions $R^{2D}$ and their mutual relations collected in a feature adjacency graph [17, 23]. It allows to derive point, line, and region neighborhood aggregates $A^{2D}$, where the point induced vertices $V^{2D}$ are the most promising vertices for starting our analysis.

To cope with the combinatoric complexity of interpretation and reconstruction processes, a tight integration of 2D and 3D reasoning has proven to be successful [30, 32]. Therefore, we decide to carry out the 3D reconstruction process already for local feature aggregates, to meet projective ambiguities and to allow an early domain specific interpretation utilizing 3D geometry. Due to their relative stability against partial occlusions we select corners (classified vertices) as the basic class of feature aggregates for the reconstruction process. This selection should be regarded as a first choice. Future work will utilize a polymorphic reconstruction approach on the feature aggregate level, which will also include line- and region-induced feature aggregates, i.e., wings and faces [8].

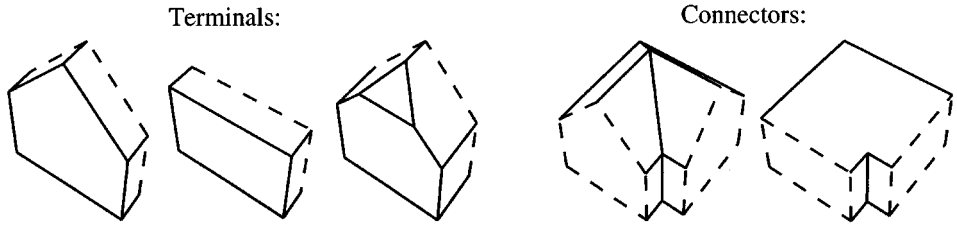



**FIG. 3.** Some examples of building parts. Plug faces, which are used to connect them, are drawn dashed.

Now, our current strategy consists of three main tasks, which are executed in sequence (cf. Fig. 4) and will be described in detail in Section 3.

*(1) 2D → 3D: Reconstruction of 3D corners.* To cope with the combinatorics of feature and building aggregation, we aim at an early integration of 2D and 3D reasoning, in this way reducing the overall number of future hypotheses. This is done by extracting feature groups, namely vertices in the images, which are evaluated as projections of building corners with the help of a 2D corner model. These 2D corner hypotheses of different images are used to derive 3D corner reconstructions by multiimage parameter estimation. The 3D reconstructions will be interpreted by a domain-specific 3D corner model.

*(2) 3D → 3D: Generation of building hypotheses.* The 3D corners are used for indexing into a library of parameterized building parts, which explain these corners. Within this indexing process, the parameter set of an indexed building part is instantiated due to the measured 3D geometry of the corner reconstructions. Generally, more than one building part is needed to explain all reconstructed 3D corners. Thus, the indexed building parts must be aggregated to complete 3D building hypotheses. This aggregation step is implemented by successive steps of merging and connecting building parts.

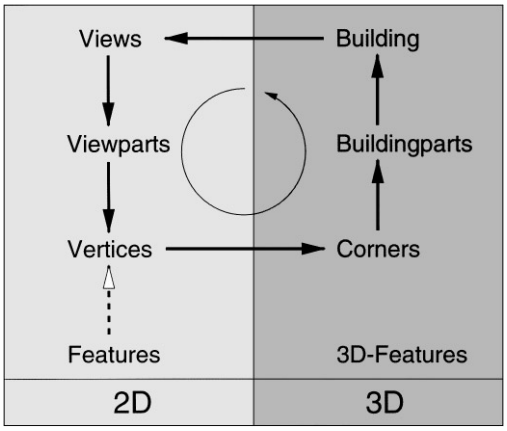



**FIG. 4.** Information transfer within the whole process. The close integration of 2D and 3D reasoning is performed by an iteration loop. The dashed arrow marks the initialization step. 3D reconstruction, generation, and verification of hypotheses for building parts and buildings is repeated. Verification is based on generated views using matching on the feature level.

*(3) 3D → 2D: Verification of building hypotheses.* For verification, the 3D building hypotheses are projected back into the original aerial images. Due to incomplete results of feature extraction, in general not all 3D corners can be reconstructed. Therefore, not all parameters of building parts and the aggregated building hypotheses may be fixed. Thus, we call the projected hypotheses *parameterized views* and implement these views in terms of a modified aspect graph notation. Hypotheses are confirmed respectively rejected by a voting process according to the degree of similarity resulting from the relational matching between the parameterized views and the originally extracted image features. Furthermore, geometric reasoning in the images by applying projective geometry may derive previously undetected image features and determine free object parameters of the building hypotheses.

The successful sequence of matching steps results in an iteratively improved gain in knowledge. These three steps are repeated until no further hypotheses can be generated. The verified building hypotheses lead to predicted unobserved 2D primitives on the lower levels, giving additional information for reconstructing previously undetected corners. This initiates a further iteration of (1) 3D-reconstruction, (2) generation of building hypotheses, and (3) verification.

## 3. MODELS AND OPERATIONS

In this section we describe in detail the three main steps of our approach to building reconstruction as proposed in Section 2.2, i.e., (1) 2D → 3D: reconstruction of 3D corners, (2) 3D → 3D: generation of building hypotheses, and (3) 3D → 2D: verification of building hypotheses (cf. Fig. 4). Each step is presented by describing the underlying models and the operations of reconstruction, aggregation, and verification.

### 3.1. 2D → 3D: Reconstruction of 3D Corners

This section describes the first two levels of our hierarchical aggregation approach, namely the extraction and reconstruction of features and feature aggregates (cf. Fig. 1). We select point-induced corner features as the first choice for feature aggregates to be processed. Corners (a) reveal a high stability against occlusions [9], (b) their projections into the images give strong restrictions during the correspondence analysis using epipolar geometry, and (c) corners have proved to be suitable components for the proposed 3D aggregation process.

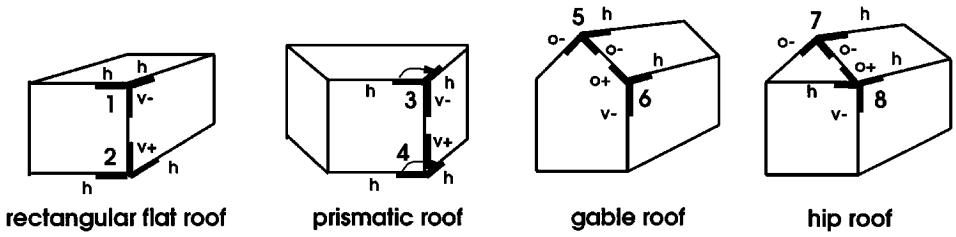



**FIG. 5.** Eight examples of corner classes $\omega_c$, which are sufficient for describing the building types rectangular flat roof, prismatic roof, gable roof, and hip roof, are shown.

Our experience with fully automatic subprocesses for the purpose of 3D reconstruction has shown a high stability while simultaneously using image patches with multiple overlap (cf. [43, 44]). Furthermore, we use the commonly available orientation data which define geometric restrictions during the correspondence analysis.

### *3.1.1. Models of Corner Features*

The corner reconstruction is guided top-down by the 3D corner model, with the corners being parts of the hierarchical building model as described in Section 2.1. For reconstruction, we perform the transition from image space to object space on the level of feature aggregates (cf. Fig. 1). Therefore, we need to model corners in object space as well as their appearance in the images by using the corner model in 3D and 2D.

*3D corner model.* The representation of corners is subdivided into the geometric information and the domain specific class labels. Each corner $c_i^{3D} = (v_i^{3D}, \omega_{c,i})$ is geometrically described by the vertex $v_i^{3D}$, which is represented by its components on the feature level, being a corner point, several lines, and planar faces (cf. Fig. 2) and a corner class label $\omega_c \in \Omega_C$ out of the set of all possible corner classes $\Omega_C$.

The corner classes are given by a two-level specialization hierarchy, which distinguishes corners into subclasses $\omega_c$. The partitioning into classes $\omega_c$ depends on different geometric class inherent constraints $\theta_{\omega_c}^{3D} \in \Theta_{\omega_c}^{3D}$ from the set of all possible constraints $\Theta_{\omega_c}^{3D}$ of the class $\omega_c$. The 3D corner $c^{3D} = (v^{3D}, \omega_c)$ results from implying the class dependent constraints $\Theta_{\omega_c}^{3D}$ onto the vertex geometry $v^{3D}$.

On the first level of the specialization hierarchy we use unary constraints for classification. They refer to single components of a corner, especially the corner components of type line. We use line attributes, given by the qualitative geometric labels horizontal (h), vertical (v+) and (v−), and oblique (o+) and (o−) (cf. [26]) depending on the slope of the lines. The sign denotes a positive (+) or negative (−) slope with respect to the corner point. We exclude meaningless corners like, e.g., a corner with line attributes {h,h,h}, which make no sense in the context of buildings and their functionality.

This description is further refined by the second level of the specialization hierarchy. It uses binary constraints which refer to the geometric relationship of pairs of corner components. Examples of binary constraints are symmetry of two lines with respect to the vertical, orthogonality of two lines, and verticality of the plane, spanned up by two lines.

If there are no constraints attached to the corner, we call it the unconstrained corner with the class label $\omega_\emptyset$.

Examples of different corner classes are given in Fig. 5. The typical corner at the ridge of a hip roof (corner no. 7) is given by the line attributes {h,o−,o−,symmetry (o−,o−)}. The distinction between corner no. 7 and corner no. 5 is due to the constraint verticality(o−,o−) on the second level of the specialization hierarchy, whereas on the first level both corner classes are described by the line attributes {h,o−,o−}.

*2D corner model.* The 2D corner model is fundamental for the access to vertices in the images, which are suitable for the corner reconstruction. Because of the higher expressiveness of the 3D vertex geometry in contrast to 2D, the assignment of a corner class label $\omega_c$ to the geometric vertex elements can be much more easily performed in object space than in image space.

Therefore, we propose using a 2D corner model without any distinction into subclasses to find point-induced vertex aggregates $V^{2D}$ that have the structural characteristics of corner projections into the images (cf. Fig. 2 (right)). These characteristics define the vertex class $\omega_v$. We statistically model the probability of a vertex being a projected corner, using the conditional probabilities $P(\omega_v \mid v^{2D})$ of the vertex class $\omega_v$ given the vertex observation $v^{2D}$. The conditional probabilities are used for classifying point-induced feature aggregates being a vertex. Please note that the transition of vertices to corners, in addition to the transition to object space, requires the assignment of a class label $\omega_c$ of a corner class.

### *3.1.2. Operations: Construction and Verification of Corner Hypotheses*

The construction of corner hypotheses starts with the analysis of 2D vertices $V^{2D}$. The vertices can be directly derived by using the feature adjacency graph (cf. Section 2.2).

By correspondence analysis using the images $i \in I$, we derive the vertex correspondence set $[v_i^{2D}] = (v_1^{2D}, \ldots, v_I^{2D})$ which is then used for the transition to 3D vertices $V^{3D}$. The vertices represent the geometry of corners $C^{3D}$ and thus serve for the subsequent interpretation by assigning the vertex $v^{3D}$ to a

corner class $\omega_c$. The results of this step are corner hypotheses $c_i^{3D} = (v_i^{3D}, \omega_{c,i})$.

In the following, we give a detailed description of the 3D vertex generation and 3D corner generation steps.

*3D vertex generation.* In the first step, the search for corresponding vertices is guided by a priority list of 2D vertices $V^{2D}$, which is built up by evaluating their suitability for the correspondence analysis and reconstruction. The evaluation is performed by statistical classification, based on the 2D corner model. The priority list is given by minimizing the information $I(\omega_v \mid v^{2D})$ for the vertex class $\omega_v$ while observing the vertex $v^{2D}$. The information $I(\omega_v \mid v^{2D})$ is derived using the probability $P(\omega_v \mid v^{2D})$ with $I(\omega_v \mid v^{2D}) = -\ln P(\omega_v \mid v^{2D})/\ln 2$.

The conditional probability results from $P(\omega_v \mid v^{2D}) = P(\omega_v \mid \boldsymbol{b}, \boldsymbol{c})$ using discrete and continuous vertex characteristics $\boldsymbol{b}$ and $\boldsymbol{c}$. Bayes' theorem breaks it down into

$$P(\omega_v \mid \boldsymbol{b}, \boldsymbol{c}) = \frac{\prod_{k=1}^{n_b} P(b_k \mid \omega_v) \cdot p(\boldsymbol{c} \mid \omega_v)}{\prod_{k=1}^{n_b} P(b_k) \cdot p(\boldsymbol{c})} \cdot P(\omega_v), \qquad (1)$$

with the discrete characteristics $b_k$ with $k = 1, \ldots, n_b$ and the continuous characteristics $c_j$ with $j = 1, \ldots, n_c$. The latter are assumed to be uncorrelated with mean $\mu_{c_j}$ and variance $\sigma_{c_j}^2$. This leads to

$$I(\omega_v \mid \boldsymbol{b}, \boldsymbol{c}) = \frac{1}{2 \ln 2} \left[ \sum_{k=1}^{n_b} -2 \ln P(b_k \mid \omega_v) + \sum_{j=1}^{n_c} \left( \frac{c_j - \mu_{c_j}}{\sigma_{c_j}} \right)^2 \right]. \qquad (2)$$

The characteristics consider the stability, uniqueness, and structural richness of the vertex. Actually, we use the number of lines and regions as discrete characteristics and the length of corner lines as continuous characteristics.

The second step during the search for corresponding vertices evaluates the structural similarity of matching candidates by cost minimization. In addition to the information of single vertices, a score value describing the similarity of vertex components using epipolar geometry is introduced into the cost function.

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 gives geometric restrictions, which facilitate the matching of the features.

The search space for the next correspondence set is defined by the neighborhood relations of the corresponding image features of the reconstructed features $F^{3D}$ using the prolongation of lines and the neighborhood of regions. For instance, we must expect further corners in the prolongation of reconstructed 3D lines. Therefore, we first follow line features of reconstructed vertices for directly selecting a set of vertex aggregates $V^{2D}$, which serves for the next reconstruction step (cf. Fig. 6). In case no appropriate vertices are found, the neighborhood of regions are used for vertex access by analyzing the region-induced aggregates.

In addition to vertices, which are directly derived by using the feature adjacency graph, we also perform a top-down prediction of additional vertices. Depending on a selected pair of possibly corresponding vertices $(v_i, v_j)$, we derive the 3D position of the vertex point by forward intersection, which we propagate to the expected vertex positions in the remaining images by reprojection. At these positions we can then perform a well-focused search for line intersections to generate additional 2D vertices. The prediction helps to bridge the incompleteness of

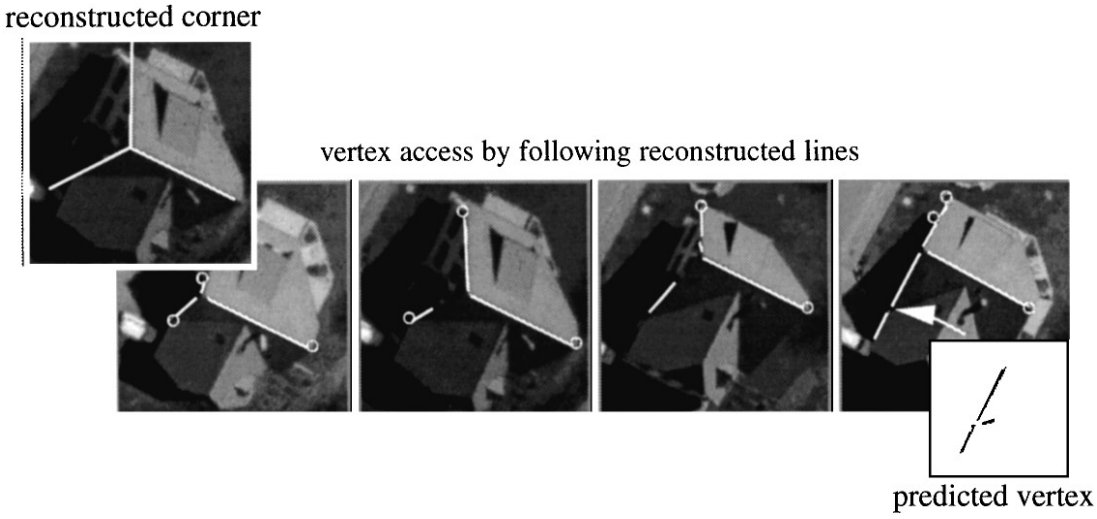



**FIG. 6.** The information propagation for selecting and predicting vertices for the next reconstruction step is shown. The upper left image visualizes one reconstructed corner. The other images show the reduced search space for vertices, which are used for the next corner reconstruction. The access to these vertices is given by the point–line adjacency relation defined by image lines, which are used for the reconstruction of the previous corners. The right image shows the top-down prediction of a vertex, generated by line intersection, where no vertex access via the feature adjacency graph was possible.

the extracted feature adjacency relations without unnecessarily enlarging the search space.

*3D corner generation.* Establishing corner hypotheses $C^{3D}$ uses the 3D corner model by following the specialization hierarchy of corners. We perform a hierarchical interpretation, starting with the set of all corner classes $\Omega_C$. It is reduced in two steps to $\Omega'_C$ and $\Omega''_C$ by testing possible class inherent constraints $\theta^{3D}_{\omega_c} \in \Theta^{3D}_{\omega_c}$. The reason for performing a two-step interpretation, using the corner specialization hierarchy, is to prevent the instantiation of mutually dependent, redundant, or contradicting constraints, which is fundamental for the subsequent verification by parameter estimation (cf. [18]).

The first interpretation step tests the generated 3D vertices $V^{3D}$ for unary constraints. Therefore, the corner components of type line are analyzed for their qualitative geometric labels, being h, v+, v−, o+ or o−. They are used to obtain the subset $\Omega'_C \subset \Omega_C$ of possible corner classes. In the second interpretation step, for each element of $\omega_c \in \Omega'_C$ possible binary constraints like, e.g., symmetry and orthogonality are tested. Thus, we yield a further reduced $\Omega''_C \subset \Omega'_C$, where each element in $\Omega''_C$ is an admissible interpretation of the vertex $v^{3D}$. The corner specialization hierarchy for the example ensures that we do not test the line pair (h, v+) for orthogonality because the orthogonality constraint is implicitly contained within the line labels and may lead to redundant or even contradicting constraints.

If the test accepts no constraints, $\Omega''_C$ is the empty set and the underlying vertex represents an unconstrained corner of class $\omega_\emptyset$.

Depending on the selected constraints, we complete the reconstructed vertices to corners by using the most probable corner class $\omega_c$, which contains the selected constraints. In the simplest case we just index into the corner classes (cf. Fig. 5). In general, we may even hypothesize additional corner components of feature type line $L$ to complete the vertex. For example, if we observe a reconstructed vertex of node degree 2 with line labels h and o− and the constraint orthogonal (h,o−), we perform a model-based prediction of a corner of node degree 3 of class {h,o−,o−,symmetry(o−,o−),verticality(o−,o−)}, as it occurs for the corner at the ridge of a gabled roof building (cf. corner no. 5, Fig. 5). For a more general approach, a priori probability distributions for the different corner classes can be learned as proposed in [13] and can be introduced to decide for the most probable corner class $\omega_c$.

The 3D corner generation may lead to several corner hypotheses $c^{3D}_j = (v^{3D}_j, \omega_{c,j})$ for each vertex reconstruction $v^{3D}$, which must be resolved in the verification step (cf. Fig. 7).

*Verification of corner hypotheses.* For verification of the corner hypotheses, we perform a second rigorous classification by statistical analysis. This is formulated as an optimization problem for finding the best interpretation $\hat{c}$ of the data $[v^{2D}_i]$ from all possible corner interpretations $c_j = (v^{3D}, \omega_{c,j})$ with $\omega_{c,j} \in \Omega''_C$. Using Bayes theorem and neglecting the denominator by normalization, we can break down the conditional probabilites, which leads to

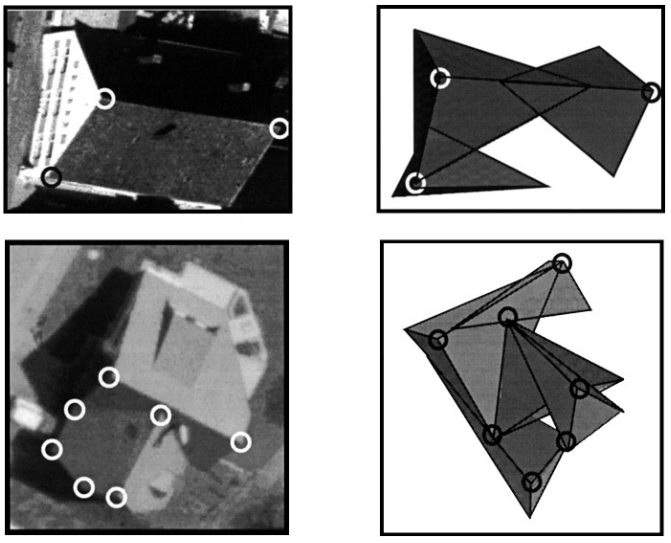


**FIG. 7.** Examples of 3D corner reconstruction. The left column shows the feature aggregates marked in one image. The right column visualizes the reconstructed 3D corners with each corner plane shown as triangle (please note that the corner planes are open ended, the drawn lines solely support the visualization).

$$\hat{c} = arg\max_{c_j} P\left(c_j \mid \left[v^{2D}_i\right]\right) \propto arg\max_{c_j} P\left(\left[v^{2D}_i\right] \mid c_j\right) P(c_j). \tag{3}$$

The a priori probability $P(c_j)$ for the corner class $\omega_{c,j}$ can be obtained empirically by learning [13] and can in principle be integrated to evaluate the model-dependent influence on the result.

The conditional probability $P([v^{2D}_i] \mid c_j)$ evaluates how good the corner class instantiation fits the observed image features points $P^{2D}$ and lines $L^{2D}$, which are contained in the vertex $[v^{2D}_i]$. Introducing the unknown class specific parameters $\boldsymbol{\beta}_j$, which depend on the geometric constraints $\Theta_{\omega_c}$ of the corner class $\omega_{c,j}$, Eq. (3) is transformed to

$$\hat{c}, \hat{\boldsymbol{\beta}} \propto arg\max_{c_j, \boldsymbol{\beta}_j} P\left(\left[v^{2D}_i\right] \mid c_j, \boldsymbol{\beta}_j\right) P(\boldsymbol{\beta}_j \mid c_j) P(c_j) \tag{4}$$

$$\propto arg\max_{c_j, \boldsymbol{\beta}_j} P\left(\left[v^{2D}_i\right] \mid \boldsymbol{\beta}_j\right) P(c_j). \tag{5}$$

As we suppose the unknowns $\boldsymbol{\beta}_j$ to be predefined by the model, the probability $P(\boldsymbol{\beta}_j \mid c_j)$ is constant and Eq. (4) can be reduced to Eq. (5).

For each hypothesis, we estimate the geometric parameters by a maximum likelihood parameter estimation using all supporting image features. The matching features are selected by backprojecting the instantiated corner model into the images. Thus, we may get access to features which were not contained in the selected vertices. Assuming the expected values $E(\boldsymbol{y})$ of the observations being a function $E(\boldsymbol{y}) = f(\boldsymbol{\beta})$ of the geometric parameters $\boldsymbol{\beta}$, the evaluation can be derived from the residuals $\boldsymbol{y} - \hat{\boldsymbol{y}}$ of the optimal estimation $\hat{\boldsymbol{y}} = f(\boldsymbol{\beta})$. We use the *probability density function* $p([v^{2D}_i] \mid \boldsymbol{\beta})$ in case the features exist and

**TABLE 1**
**The Results of the Parameter Estimation for Two Examples**

| | | $n$ | $u$ | $r$ | $\Omega[1]$ | $\sigma_0[1]$ | $\sigma_x[m]$ | $\sigma_y[m]$ | $\sigma_z[m]$ |
|---|---|---|---|---|---|---|---|---|---|
| Example 1 | no constraints | 42 | 9 | 33 | 33.65 | 5.86 | 0.037 | 0.026 | 0.063 |
| | constraints | 42 | 5 | 37 | 38.47 | 6.32 | 0.021 | 0.018 | 0.036 |
| Example 2 | no constraints | 34 | 9 | 25 | 40.80 | 8.16 | 0.074 | 0.034 | 0.128 |
| | constraints | 34 | 6 | 28 | 45.81 | 8.66 | 0.043 | 0.085 | 0.075 |

*Note.* The class depending constraints and no constraints are given for comparison. In both cases the constraints are accepted. The precision of the reconstructed vertex point is increased when using the constraints for stabilization. ($n$ is the number of observations, $u$ is the number of unknown parameters, and $r$ is the redundancy of the estimation system.)

have been successfully matched to the model.

$$p\left(\left[v_i^{2D}\right] \mid \boldsymbol{\beta}\right) = \frac{1}{(2\pi)^{u/2}(\det\Sigma_{yy})^{1/2}}\, e^{\left(-\frac{1}{2}(\boldsymbol{y}-\hat{\boldsymbol{y}})^T \Sigma_{yy}^{-1}(\boldsymbol{y}-\hat{\boldsymbol{y}})\right)}, \quad (6)$$

where $u$ is the number of unknowns in $\boldsymbol{\beta}$ and $\Sigma_{yy}$ is the covariance matrix of the observations $\boldsymbol{y}$.

To decide for the optimal interpretation and its corresponding constraints, which define the unknowns $\boldsymbol{\beta}$, we first test for acceptance of the reconstruction of the unconstrained corner of class $\omega_\emptyset$ with $\Theta^{3D}_{\omega_\emptyset} = \emptyset$. If this test is successful, we test for the class-specific constraints $\Theta^{3D}_{\omega_c}$. Both tests use a Fisher distributed test value, depending on the weighted sum of the squared residuals[2] $\Omega = (\boldsymbol{y}-\hat{\boldsymbol{y}})^T \Sigma_{yy}^{-1}(\boldsymbol{y}-\hat{\boldsymbol{y}})$ of the maximum likelihood parameter estimation.

Table 1 gives two examples of the estimation using 4 images simultaneously. Example 1 uses constraints of the corner class no. 5, example 2 of the corner class no. 6 (cf. Fig. 5) as selected during corner hypotheses generation.

According to the second test, the increase of the internally derived precision $\sigma_0$ is not significant. Thus, in both cases the class-specific constraints $\Theta^{3D}_{\omega_c}$ are accepted. The constraints are useful for geometric stabilization and for improvement of the accuracy of the reconstruction as shown with decrease of the empirical standard deviation $\sigma_x$, $\sigma_y$, and $\sigma_z$ of the three corner coordinates $(x, y, z)$ (cf. Table 1). Further details of the corner reconstruction approach can be found in [45].

The result of the estimation are evaluated corner reconstructions $c^{3D} = (v^{3D}, \omega_c)$ constrained by $\Theta^{3D}_{\omega_c}$. They form the basis for the 3D aggregation and generation of building hypotheses as presented in the next section. After the first iteration of the whole building reconstruction process, we want to use newly generated 2D corners $C^{2D}$ (cf. Section 3.3.2), in addition to the original vertex data and thus we expect an increasing number of reconstructed corners.

[2] Please note that $\Omega$ in this context is used for a statistical measure in contrast to $\Omega_c$ being the set of possible corner classes $\omega_c \in \Omega_c$.

### 3.2. 3D → 3D: Generation of Building Hypotheses

This section describes the last two levels of our hierarchical aggregation approach in 3D, namely, the reconstruction of building parts and building aggregates. The indexing into a library of parameterized volumetric building parts uses the 3D corner hypotheses derived at the feature aggregate level. Building parts are merged and combined successively to complex building aggregates describing complete building hypotheses.

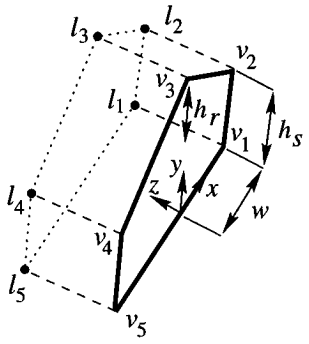
In this section we describe in detail our models of 3D building parts and building aggregates and the operations for the selection and aggregation of building parts to complete 3D building hypotheses.

#### *3.2.1. Models of Building Parts and Buildings*

Given a set $C^{3D} = \{c_1^{3D}, \ldots, c_n^{3D}\}$ of $n$ reconstructed corner aggregates, one or more building hypotheses must be generated that explain all corners in the following sense. A corner is either part of the building hypotheses or is explicitly classified as not being part of it. If a corner is part of a hypothesis, this corner must match the hypothesis geometry and structure.

We use a component-based approach for the modeling and construction of buildings. Building part models are selected and instantiated according to the reconstructed 3D corners. Subsequent aggregation operations combine these building parts of more complex ones and finally to complete buildings. In the current state of our work the set of building part models is built up manually. In the future, learning schemes as described in [13] will be used.

*Plug faces, terminals, and connectors.* Each building part contains one or more so-called *plug faces*, where it can be connected to other building parts. According to the number of plug faces the set of building part models is classified into *terminals* having one plug face and *connectors* having two or more plug faces (cf. Fig. 3). A type is assigned to each plug face, which represents its geometry and topology. Two building parts may be connected to each other via two plug faces only if they are of the same type. A building part is called *open*, if it contains one or more *plug faces* and *closed* if it contains no plug face. A final building hypothesis consists of one closed building part.




$$v_i = \begin{pmatrix} x_i \\ y_i \\ z_i \end{pmatrix}$$

| $v_i$ | $x_i$ | $y_i$ | $z_i$ |
|---|---|---|---|
| $v_1$ | $+w$ | $0$ | $0$ |
| $v_2$ | $+w$ | $+h_s$ | $0$ |
| $v_3$ | $0$ | $+h_s + h_r$ | $0$ |
| $v_4$ | $-w$ | $+h_s$ | $0$ |
| $v_5$ | $-w$ | $0$ | $0$ |

**FIG. 8.** Parameterized description of a hip roof terminal. The table on the right shows the coordinates of the vertices $v_1, \ldots, v_5$.

*Parameterization of Building Parts.* Building part models are parameterized by pose parameters determining the location and orientation and by shape parameters determining features like slope of the roof, width, and height (cf. Fig. 8). If a parameter has an assigned value it is called *fixed*, otherwise *free*. Constraints on the parameters restrict them to valid values.

Closed building parts that cover all corner observations are the requested building hypotheses. In general, there will be more than one building hypothesis for a given set of corner observations.

### 3.2.2. Operations: Selection and Aggregation of Building Parts

The generation of building hypotheses is performed with three basic types of operations: `Indexing` selects and instantiates building part models from a library of building parts. The two aggregation operations of `Merging` and `Connection` combine existing building parts to more complex ones. Closed building parts generated in this process, which cover all corner observations, are called building hypotheses.

First, we describe these three types of operations in detail. Then, we propose our strategy of organizing these operations to derive 3D building hypotheses in an efficient way.

*Indexing, merging, and connection.* `Indexing(`$c$`)` into the library of building part models yields one partially instantiated building part for the reconstructed 3D corner $c$. Location and orientation of the reconstructed corner $c$ will also fix the position and orientation of the indexed building parts. Also, its parameters of shape will be determined as far as possible. In general, there are several competing building part models and several different instances of each of these models that match a single corner. Therefore, subsequent invocations of `Indexing(`$c$`)` yield different building parts.

We say that the corner reconstruction $c$ is *covered* by the instantiated building part (cf. Fig. 9). Indexing finds for a given corner observation $c_i$ all possible mappings $\phi$ to all corners $c_j$ of all building part primitives, such that one such $\phi$ maps $c_i$'s edge indices onto $c_j$'s edge indices with the following properties:

- Every edge of $c_i$ is mapped onto one edge of $c_j$.
- The model corner $c_j$ may have more edges than $c_i$, but not less, due to possibly unobservable edges in the images.
- The labels of a model edge and an observed edge identified by $\phi$ must be the same.

If for a pair of model corner $c_j$ and corner observation $c_i$ such a mapping $\phi$ is found, it implies a set of equations, which determine location and orientation, and some form parameters of the instantiated building part template.

`Merging(`$b_1$`,` $b_2$`)` joins two building parts $b_1$ and $b_2$ of the same type, which show the same location and orientation and a

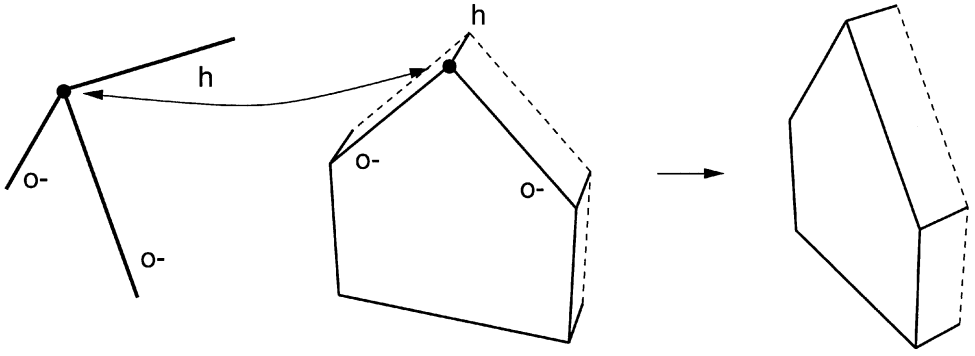



**FIG. 9.** Indexing establishes a link between corner observations (left) and building part primitives (center) resulting in an instantiated building part (right).

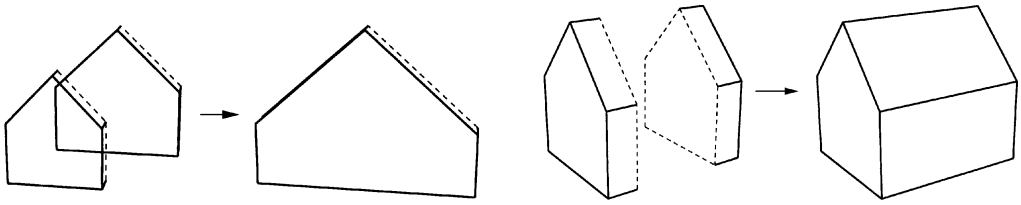


**FIG. 10.** (Left) Two saddle roof terminals are merged together. Parameters are adjusted. (right) Two saddle roof terminals are connected together. The resulting building part is closed.

consistent setting of the fixed parameters of shape (cf. Fig. 10). All corners by $b_1$ and $b_2$ are also covered by the resulting building part.

`Connection(`$b_1$`,`$b_2$`,` $f_1$`,` $f_2$`)` is the aggregation of the two building parts $b_1$ and $b_2$ via their plug faces $f_1$ and $f_2$. The two plug faces must have the same type and their orientation must be opposite to each other. The aggregation is performed by "glueing" $f_1$ to $f_2$ (cf. Fig. 10). Edges of the two building parts, which meet at $f_1$ and $f_2$, are joined together.

A new length parameter is introduced, that allows these new edges to shrink or expand. Parameters that define the shape of the plug faces are unified. If parameters which are to be unified have fixed values, they may differ only by a small amount. This condition may be used for an early rejection of the aggregation of two building parts: if for example the position and orientation of two building parts do not fit, they must not be aggregated. All corners covered by $b_1$ and $b_2$ are also covered by the resulting building part.

Each operation is followed by a maximum likelihood parameter estimation in the form of a least-squares fitting. In the case of a merging or connection operation, this parameter estimation determines whether the parameters of the two participating building parts are compatible or not. In the case of an indexing operation it determines initial values for position and orientation and some of its form parameters.

*Strategy for selection and aggregation.* The straightforward procedure would start with a set $B_0$ that contains all instantiated building parts resulting from indexing. Iterated aggregation steps generate from a set $B_i$ a new set $B_{i+1} = B_i \dot{\cup} N_i$, where $N_i$ contains all possible building parts that are constructed by merging and connection of building parts in $B_i$. Note that only building parts which are not already in $B_i$ will be constructed. Therefore, $B_i$ and $N_i$ are disjoint. The iterations stop after $n$ steps, because there are only $n$ corner observations and a building hypothesis may not contain more than one building part per reconstructed corner. The number of hypotheses contained in $B_n$ is exponential in $n$.

We use an approach that employs a heuristics that typically allows the construction of the best building hypothesis in a few steps while the construction of all closed building hypotheses covering all corner observations is still exponential. This is achieved by using a priority queue to order the operations `Indexing`, `Merging`, and `Connection`. As long as the queue is not empty, the first operation in it is performed and removed from the queue. Afterward, new operations are inserted into the queue. The process terminates if the priority queue is empty and no new operations can be inserted. All closed building parts generated in this process that cover all corner reconstructions are called building hypotheses.

*Organization and execution of the priority queue.* The strategy which leads to a fast construction of hypotheses is expressed by the following construction principles:

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

**S2.** Aggregation is done depth first. Construct building aggregates with as much covered corner observations and minimal values for the least-squares fitting as soon as possible.

**S3.** The growth of aggregations shall be as uniform as possible. This avoids having one kernel steadily growing by connecting small building parts to it and thus neglecting other possible hypotheses.

These construction principles define how the operations are ordered within the priority queue and which operation is to be performed next. Let $\pi_1$ and $\pi_2$ be two operations:

1. If $\pi_1$ and $\pi_2$ have different types, Indexing comes before Merging before Connection (S1).
2. If $\pi_1$ and $\pi_2$ are both of type Indexing, then
   (a) if the operations belong to different corner observations, the corner having less building part interpretations is preferred (S2),
   (b) else if the operations belong to the same corner observation, the operation with an a priori better expected result is performed first (S2).
3. If $\pi_1$ and $\pi_2$ are both of type Merging or Connection, with the argument building parts $b_{i,1}$ and $b_{i,2}$ of $\pi_i$, the following attributes are considered:
   (a) The operation $\pi_i$ with the higher number of corner observations covered by $b_{i,1}$ and $b_{i,2}$ is performed first: Building aggregates with as much covered corner observations as possible are generated first (S2).
   (b) The operation $\pi_i$ with the smaller difference in corner observations covered by $b_{i,1}$ and $b_{i,2}$ is performed first (S3).
   (c) Finally, the operations $\pi_i$ are sorted according to the sum of the values of the function minimized in the least-squares

fitting performed after the construction of $b_{i,1}$ and $b_{i,2}$: Building aggregates that fit the observed corners best are constructed first (S2).

*Algorithm.* The set of building hypotheses $H$ and the set of building parts $B$ are initially empty. The priority queue is initialized with all indexing operations for each corner. As long as the queue is not empty, the first operation in it is removed and executed. If its result, the building part $b$, is closed, then if it covers all corner observations it is a new building hypothesis and is added to $H$. If $b$ does not cover all observations, it is removed. If $b$ is open, new entries for the priority queue are constructed:

- If $b$ is an instance of a building part model, then for the set $\{b_1, \ldots, b_k\} \subseteq B$ of existing building parts, which are instances of the same building part model and have the same orientation (with respect to uncertainty), the operations `Merging(`$b$`,` $b_i$`)`, $1 \leq i \leq k$, are inserted into the priority queue.
- For each plug face $f$ of $b$ and for each plug face $f'$ of another building part $b' \in B$: if $f$ and $f'$ are of the same type and their orientation is opposite to each other, insert the operation `Connection(`$b$`,` $b'$`,` $f$`,` $f'$`)` into the priority queue.

An operation is inserted only if its result is not yet a member of $B$ or $H$ and if it covers every corner observation at least once. After all possible operations are inserted into the queue, $b$ is added to $B$.

If no new operations can be inserted and the priority queue is empty, the algorithm terminates. If $H$ is not empty, it contains all the generated building hypotheses. Otherwise, it has not been possible to generate a closed building hypothesis. If there exist closed building parts in $B$ which do not cover all corner observations, the one which covers the most observations is chosen as a building hypothesis. If all building parts in $B$ are open, one building part is selected from $B$ and its open plug faces are closed by connection to default terminals. The building part selected is the one which consists of the most instantiated building models and the fewest plug faces and fits the corner observations best.

*Example.* Figure 11 shows an example. Six corner reconstructions are shown on the left. After all indexing operations are performed, the best scored building parts are shown in the middle. The building part in the front covering three corner reconstructions is the result of the two merging operations. The visible three terminals and the one connector are aggregated to the building hypotheses shown on the right in three connection operations. Please note that there are no corner reconstructions at ground level, and therefore the height parameter is undetermined. Only a default value for visualization has been chosen.

### 3.3. 3D → 2D: Verification of Building Hypotheses

This section describes the last two levels of our hierarchical aggregation approach in 2D, namely the image models of building parts and building aggregates. Each aggregation level can be accompanied by corresponding aspect descriptions (cf. [14]) for intermediate hypotheses verification and parameter estimation by image analysis. Due to the high complexity of hypotheses verification for complex building shapes, we employ constraint satisfaction and propagation to verify hypotheses in an efficient way. We have currently implemented our verification procedure for complete—but also only partially instantiated—building hypotheses. Obviously, our verification procedure can also be applied for the verification of intermediate aggregation results.

Thus, in this section we describe the generation of the image models for the 3D hypotheses of buildings and building parts and the constraint-based reasoning for hypotheses verification and the derivation of new image features.

#### *3.3.1. Image Modeling by Parameterized View Hierarchies*

To generate image models of building parts and aggregates, we employ a modified version of the aspect hierarchies [11]. Aspect hierarchies describe qualitative image models of volumetric primitives in terms of aspects. Each aspect is decomposed into a hierarchical feature-based description, in order to facilitate the recognition of primitives even in the case of partial occlusion by the detection of its visible features.

While aspects represent qualitatively different object appearances due to different viewpoint positions, different appearances of our building hypotheses are caused by different settings of free shape parameters. The parameters of location and orientation are

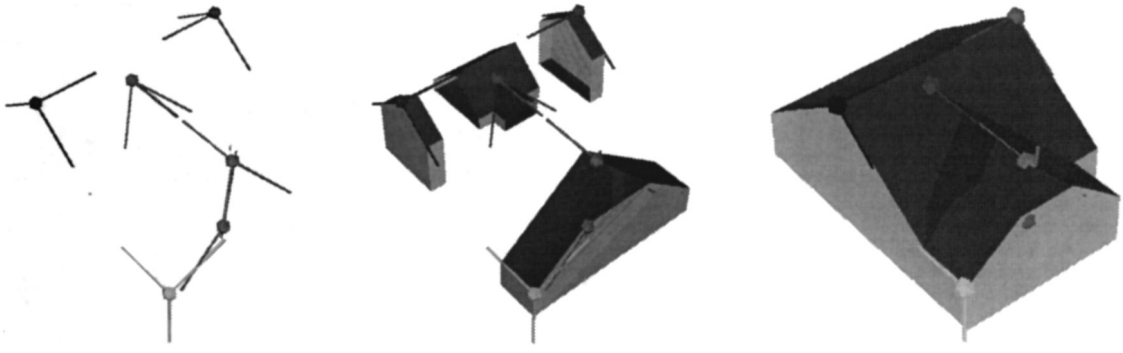


**FIG. 11.** The given corner observations are shown on the left. The figure in the middle shows the best fitting building parts after all indexing steps and some merging steps (building part in the front). These building parts are aggregated in three operations resulting in the building hypothesis shown on the right.

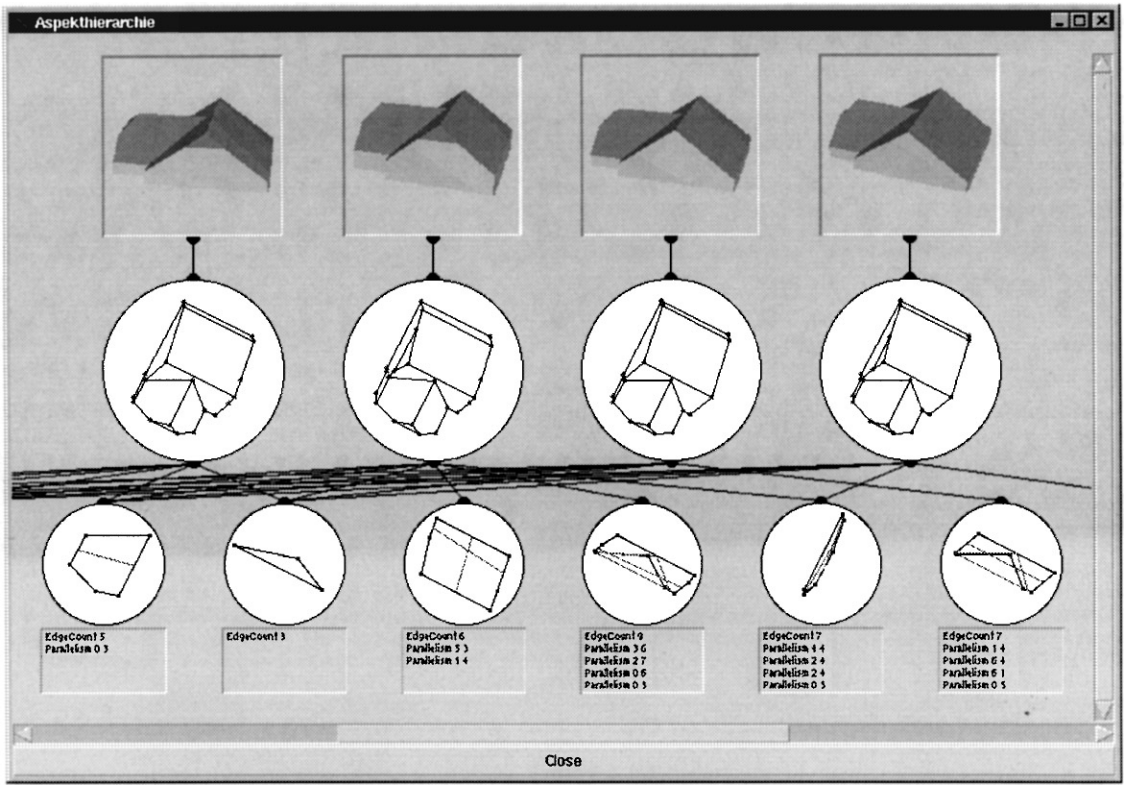



**FIG. 12.** Example of a view hierarchy. For one set of corner reconstructions four different building hypotheses have been generated due to two different indexing results into the library of building parts for the leftmost ridge corner and the free parameter of height (no ground corner was reconstructed).

usually fixed due to the fitting with the 3D corner reconstructions. If all parameters of a building hypothesis are fixed, exactly one view is generated due to the fixed camera position and orientation. If one or more parameters are still undetermined, the valid parameter space is sampled and several views are inserted (see Fig. 12). Due to the reliable processes of feature extraction and reconstruction, the number of free parameters is generally low.

To facilitate hypotheses verification, even in the case when occlusions and noise cause incomplete feature extraction, we adopt the hierarchical feature-based representation of the approach of Dickinson *et al.* (cf. [11, 12]). The image models of the hypotheses of buildings and building parts are called *parameterized view hierarchies* due to their independency from viewpoint. One parameterized view hierarchy is generated for each aerial image. It consists of three levels:

*Level 1.* The highest level contains all three–dimensional building hypotheses. Note that a hypothesis may still have undetermined parameters.

*Level 2.* The medium level contains all relevant two-dimensional views of these hypotheses.

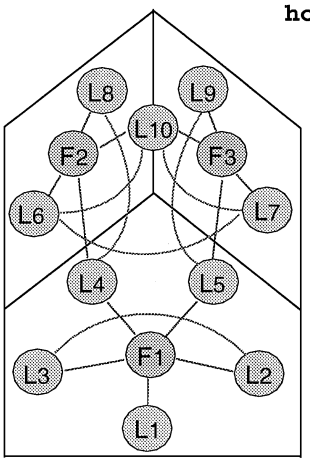
*Level 3.* Each view is decomposed into its image features and their relations, such as parallelisms and symmetries, which describe the lowest level of the hierarchy.

Figure 12 shows an example of a view hierarchy for four different building hypotheses for the same set of corner observations (cf. Fig. 11, left).

The image model currently employed for hypotheses verification utilizes a pinhole sensor model with weak perspective projection of the visible object contours. Obviously, more enhanced models of sensor characteristics, illumination, and physical properties of building materials allow further analysis for determining shape parameters and verifying intermediate and complete results of the aggregation hierarchy. Currently, we are investigating view representations that employ a standard lighting model including ambient light and diffuse reflection, where the sun vector is computed from a sensor model and a time stamp. The analysis of illumination and shadows allows one to derive 3D object parameters and to test the consistency of shadow–ground transitions and intersurface intensities (cf. [62]).

### *3.3.2. Operations for Verification and Parameter Estimation*

Hypotheses verification relies on the matching between the image models of complex building hypotheses on the one hand and extracted image features of the aerial images on the other hand. Since this matching process considers objects and their interrelationships, this task is an instance of relational matching [66]. We employ constraint-solving methods [51] to handle




```
house_aspect(F1,F2,F3,L1,...,L10,Sum) :-
    extracted_blobs(BlobDom), [F1,F2,F3] ∈ BlobDom ∪ {*},
    extracted_lines(LineDom), [L1,...,L10] ∈ LineDom ∪ {*},
    [B1,...,B19] ∈ [-1,0,1],
    all_distinct([F1,F2,F3]),
    all_distinct([L1,...,L10]),
    adjacnt(F1,L1,B1), adjacnt(F1,L2,B2), adjacnt(F1,L3,B3),
    adjacnt(F1,L4,B4), adjacnt(F1,L5,B5), adjacnt(F2,L4,B6),
    adjacnt(F2,L6,B7), adjacnt(F2,L8,B8), adjacnt(F2,L10,B9),
    adjacnt(F3,L5,B10),adjacnt(F3,L7,B11),adjacnt(F3,L9,B12),
    adjacnt(F3,L10,B13),
    line_parallel(L2,L3,B14), line_parallel(L6,L7,B15),
    line_parallel(L6,L10,B16), line_parallel(L7,L10,B17),
    line_parallel(L4,L8,B18), line_parallel(L5,L9,B19),
    Sum = B1 + B2 + ... + B18 + B19.
```

**FIG. 13.** Constraint representation of a building hypothesis for a saddleback roof house. The `adjacnt`-constraints express the topological adjacencies between lines and regions. The indicator variables $B_1, \ldots, B_{19}$ reflect the status of each constraint: $-1$ means violated, 0 relaxed, and 1 satisfied (see Section 3.3.2).

the exponential combinatorial complexity, permitting exhaustive search for the optimal match.

Matching between image features and image model yields a grouping of fragmented lines and a model-based interpretation of each matched image feature. Then, previously not extracted 2D corners are detected by geometric reasoning with robust estimations of intersections of identified lines. These 2D corners are additional information that is used to reconstruct further previously undetected 3D corners in the next iteration of the whole building reconstruction procedure.

The transformation of building hypotheses to a constraint satisfaction problem (CSP), the matching, and its implementation using constraint logic programming (CLP) [38, 65] is explained in detail in [42].

*Transformation of view hierarchies to sets of constraints.* A view hierarchy enumerates the possible views for the building hypotheses with respect to one image. To evaluate the correspondence of these image models with the originally extracted image features, the matching between the view hierarchy and the image features is done on the lowest level of our model hierarchy shown in Fig. 1. The entities considered for matching are objects of the three feature classes points, lines, and regions and different relations, e.g., `adjacency`, `line parallelism`, `region symmetry`, and `region contrast`. The line parallelism and region symmetry relations reflect the respective 3D properties in 2D. For each pair of adjacent regions the contrast relation expresses their expected intensity ratio.

Thus, we employ a representation of the image model, which describes each view as a set of 2D image features and a set of relations between them. These relations may be regarded as a set of constraints which must be satisfied simultaneously by the corresponding objects. A consistent match, also called consistent labeling [29], is an identification of a set of extracted image features that satisfies all constraints. The decomposition of the matching problem into the simultaneous satisfaction of different constraints leads to a structure CSP($\mathcal{V}, \mathcal{D}, \mathcal{C}$) which is adequately represented by the translation of hypotheses into conjunctions of constraints $\mathcal{C}$, where the features are represented by variables $\mathcal{V}$ with restricted domains $\mathcal{D}$. Figure 13 illustrates the constraint representation of views of building hypotheses using constraint logic programming. For each view of a hierarchy, one corresponding constraint representation will be employed.

*Matching of constraint sets to the image data.* The matching is done separately for every aerial image and view model by searching for a valid assignment of extracted image features to the variables of the corresponding CSP such that all constraints are satisfied. Consistency techniques, like forward checking and look ahead, as described in [10, 28, 49, 65] exploit the structured representation of the matching problem as a constraint network. Each constraint is used to restrict the domains of its variables and these reductions are propagated through the network. This causes early prunings of the search tree and in most cases dramatically reduces the computational effort.

These standard techniques for constraint solving demand that a match satisifies every constraint. Due to occlusions and disturbances, often neither every predicted model feature of a view can be found, nor are all their incident constraints satisfiable. Thus, the resulting CSPs are in general overconstrained [39]. The approaches of MaxCSP as proposed by Freuder and Wallace [20, 67] or inexact graph matching described by Haralick and Shapiro [29, 61] handle this by allowing the relaxation of constraints. The best matching then is defined as the one which satisfies the maximum number of constraints.

The main problem of this metric lies in the inappropriate representation of unobservability and is discussed in detail by Vosselman [66]. If an expected model feature is missing in the image, the corresponding variable must be assigned a wildcard value, and all its incident constraints must be relaxed in order to get a solution for the CSP. However, these costs are not correlated to the importance of the model feature.

To overcome this problem, we also permit the explicit elimination of variables. The domain of every variable is extended by the wildcard $*$, which is assigned to a variable if the respective model feature could not be found in the image. We further distinguish between the relaxation of constraints due to the unobservability of an incident feature and due to single constraint violation. Therefore, we extend every constraint $c(t_1, \ldots, t_n)$ of the model by a variable $b$ with the three possible values $-1$, 0, or 1 to a constraint $c'(t_1, \ldots, t_n, b)$, where

$$c'(t_1, \ldots, t_n, b) \Leftrightarrow (b = 0) \vee ((b = 1) \wedge c(t_1, \ldots, t_n)) \vee ((b = -1) \wedge \neg c(t_1, \ldots, t_n)). \quad (7)$$

In the framework of constraint logic programming (CLP), variable $b$ is used both as an indicator and a switch. If the original constraint $c$ is satisfied, $b$ is set to 1. If it is violated, $b$ is set to $-1$. If $b$ is set to 1 (or $-1$), $c$ must be satisfied (or violated, respectively) to fulfill $c'$, and thus $c'$ will be replaced by $c$ (or $\neg c$). If a wildcard is assigned to a variable $v_i \in \mathcal{V}$, every incident constraint $c'_j(\ldots, v_i, \ldots, b_j) \in \mathcal{C}'$ must be relaxed by setting $b_j = 0$. If the wildcard is removed from the domains of all variables $v_1, \ldots, v_n$ of a constraint $c'(v_1, \ldots, v_n, b)$, the value 0 can be removed from the domain of $b$. Vice versa, if $b$ is set to 0, the wildcard must be removed from the domains of the variables $v_1, \ldots, v_n$.

So the original constraint satisfaction problem CSP $(\mathcal{V}, \mathcal{D}, \mathcal{C})$ is transformed to a problem CSP'$(\mathcal{V}, \mathcal{D}', \mathcal{C}')$ with $\mathcal{D}' = \mathcal{D} \cup \{*\}$ and $\mathcal{C}' = \{c'(t_1, \ldots, t_n, b) \mid c(t_1, \ldots, t_n) \in \mathcal{C}, b \in \{-1, 0, 1\}\}$. To find the best mapping we maximize

$$f(\mathcal{C}) = \sum_{i=1}^{n} b_i \quad \text{with } c'_i(\ldots, b_i) \in \mathcal{C}' \quad (8)$$

using a branch-and-bound algorithm embedded in the CLP system [65]. During the search, the lower bound $\lambda$ is added as a constraint $f(\mathcal{C}) \geq \lambda$ to the constraint system. From this constraint, the CLP system can infer additional knowledge about the values of the variables $b_i$. For example, if we have eight constraints, six of which have already been fixed (four to 1 and two to 0), and $\lambda$ is 6, the final two constraints must be satisfied to reach the lower bound, and so the $b_i$ for the remaining two constraints can be set to 1, enforcing both constraints.

The solution of the constraint satisfaction problem consists of valid assignments of extracted features to the variables. On one hand, these assignments determine the free geometric parameters of the features of the image model and thus determine free parameters of the whole building hypothesis. On the other hand, every assigned image feature is identified as a specific component of the building hypothesis. These matched image features are labeled with respect to the building model. Figure 14 shows highlighted on the left the image features that were successfully matched with the image model of building hypothesis of a saddle roof house.

Since lines in general are fragmented into several line image features, corresponding line features are grouped and fitted using the labeling information to determine the lines of the image model of the building hypothesis.

*Generation of new 2D corners.* The solution of the CSP in general fixes further parameters of the building hypotheses. It should be noted that free parameters may only remain if some constraints or variables had to be relaxed. Corners of the building hypotheses with fixed positions that have no corresponding vertex structure in the image are finally detected in 2D, as illustrated by the right picture in Fig. 14, to provide new information for the 3D corner reconstruction in the next iteration of the reconstruction process.

As mentioned above, the loop of 3D reconstruction, generation, and verification of building hypotheses will be terminated if no further hypotheses can be generated. After the final verification step, the parameters of the 3D building model will be determined simultaneously based on the observed image features and taking all geometric and possibly radiometric constraints into account.

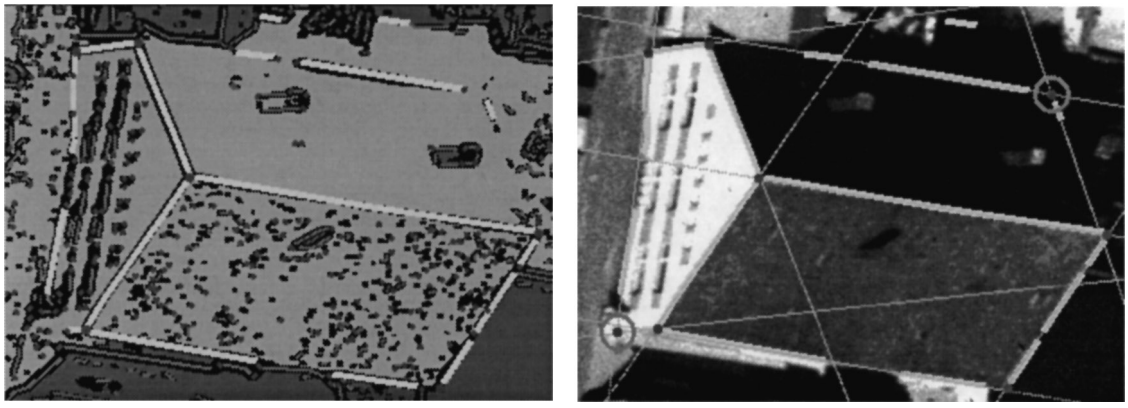


**FIG. 14.** (left) Image features that are successfully matched with the image model of the building hypothesis. (right) 2D corners, predicted by the matched image model depicted by circles and line intersections.

## 4. RESULTS

The current result of the reconstruction procedure is presented for the international test 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 Images* (cf. [25]). The image scale is 1 : 5000. Due to the fact that our feature extraction currently yields too many spurious image features (e.g., roofing tiles) at a resolution of 15 $\mu$m, we use a resolution of 30 $\mu$m pixel size. As building h12 is under construction, only 11 out of the 12 buildings which are contained in multiple images, are relevant for the analysis. To test the feasibility of the concept in a first instance the number of building primitives is reduced to some few building part terminals and their possible connectors, which occur in the data set. To reconstruct corners at the footprint of the buildings, higher image resolution is necessary. Therefore, we propose to derive the building heights by using a digital terrain model [31]. For visualization, we actually set the building heights to fixed values. Figure 15 shows the result after the first iteration loop (cf. Fig. 4). Results of the intermediate steps are listed in Table 2 and reflect the following aspects of the different reasoning steps.

*Reconstruction of 3D corners (I).* The number of reconstructed corners RC on average compounds 40% of the buildings corners (C), whereas 3 of them are incorrectly classified. We are not able to completely reconstruct every corner, because not all corners are observable in the symbolic image descriptions. An application independent grouping (as presented in [23]) may improve the initial symbolic image descriptions. Possibly, remaining unreconstructed corners must be identified during the verification step of building hypotheses. The reason for incorrect classification is given by a weak intersection geometry for the line reconstruction, which is not yet considered in the modeling process. Neglecting the influence of the uncertainty of the line reconstruction, the corner point is correctly reconstructed. Currently, the identification of wrong corner interpretations is performed by finding global inconsistencies during the generation of building hypotheses. The parameter estimation for correctly classified corners, using 4 images simultaneously, achieves an accuracy of the reconstructed corner point about $\sigma_x = \sigma_y = \pm 6$ cm and $\sigma_z = \pm 15$ cm. The accuracy of the orientation of the corners is about $\sigma_\kappa = 0.6^\circ$, the accuracy of the slope of the roof is about $\sigma_{s\ell} = 2^\circ$.

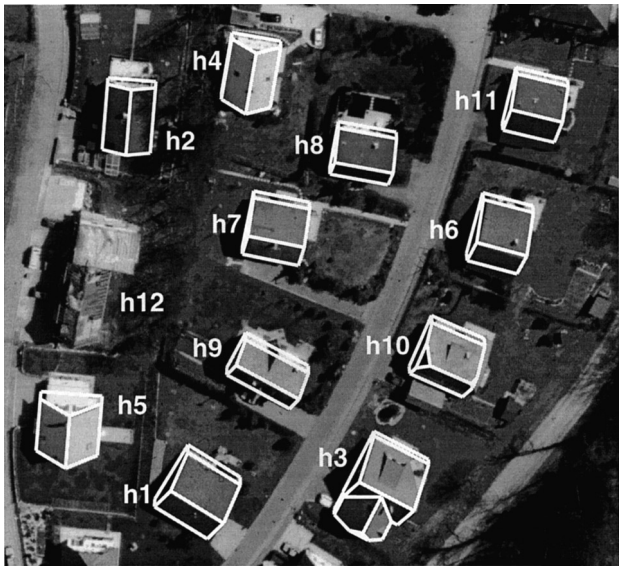



**FIG. 15.** Visualization of the reconstruction result for the Ascona data set after one iteration loop (cf. Fig. 4) is performed. The images contain 12 buildings in multiple overlap. As building h12 is under construction, we excluded it from our analysis. We are able to completely reconstruct all of the 11 buildings that were relevant for the analysis.

**TABLE 2**

**Detailed Results of the Intermediate Reconstruction Processes: Reconstruction of 3D Corners (I), Generation of Building Hypotheses (II), and Verification of Building Hypotheses (III) for the Ascona Data Set**

| | I | | | II | | III | | |
|---|---|---|---|---|---|---|---|---|
| Building | C | RC | CE | BPH | BH | UV | GV | FP |
| h1 | 6 | 3 | 0 | 3/4/3 | 4 | 3 | 3/3/3/3 | 0 |
| h2 | 6 | 1 | 0 | 3 | 2 | 5 | 5/5/5/5 | 0 |
| h3 | 12 | 7 | 0 | 3/3/2/3/3/4/0 | 4 | 5 | 3/4/3/3 | 0 |
| h4 | 6 | 3 | 0 | 4/3/3 | 4 | 3 | 3/3/3/3 | 0 |
| h5 | 6 | 3 | 0 | 3/4/4 | 4 | 3 | 3/3/3/3 | 0 |
| h6 | 6 | 3 | 1 | 3/0/4 | 4 | 4 | 4/4/4/4 | 0 |
| h7 | 6 | 2 | 1 | 3/0 | 2 | 5 | 5/5/5/5 | 0 |
| h8 | 6 | 3 | 0 | 3/3/4 | 4 | 3 | 3/3/3/3 | 0 |
| h9 | 6 | 4 | 0 | 3/3/4/8 | 2 | 2 | 2/2/2/2 | 0 |
| h10 | 10 | 6 | 1 | 1/4/4 | 2 | 5 | 3/4/3/4 | 0 |
| h11 | 6 | 2 | 0 | 3/3 | 4 | 4 | 4/4/3/4 | 0 |

*Note.* All of the 11 buildings can be reconstructed as shown in column FP. C, number of corners of the building; RC, number of reconstructed corners; CE, number of corner classification errors; BPH, number of building part hypotheses for each corner; BH, number of building hypotheses; UV, predicted corners (not previously reconstructed) before verification; GV, predicted and verified vertices per image; FP, remaining free parameters.

*Generation of building hypotheses (II).* The column BPH in Table 2 gives the number of building part hypotheses that are generated for each of the reconstructed single corners. The combination of these building parts results in the number BH of hypotheses of complete buildings that are consistent with all reconstructed corners. If only one corner is given, as it is the case for h2 and h7, the corresponding terminals are each completed to a closed building hypothesis with a second terminal of the same type. For each of the BH building hypotheses the view graph is generated. Currently, the view graph of the best building hypothesis is passed to the verification of building hypotheses. The best decision is given by using the Minimum Description Length Principle following [57].

*Verification of building hypotheses (III).* For each building in each image one view graph, which is a 2D building hypothesis, is available and matched to the extracted image features and

mutual relations. Column UV lists the number of corners which initially could not be 3D reconstructed but were predicted by the generated building hypotheses. The column GV lists for each of the 4 images the number of correctly matched vertices for the previously unknown corners. It shows that 92% of the undetermined building corners could be correctly identified by the verification procedure. Since in general the building parameters are redundantly determined by the geometrical and topological constraints between the model features, all free parameters for every building hypotheses have been determined (see column FP). Please note that for buildings h2 and h7 the matching process also determined the formerly unknown length parameter.

## 5. CONCLUSIONS AND OUTLOOK

We have proposed a model-based approach to 3D building extraction from aerial images. Our modeling concept reveals a tight coupling of generic 3D object modeling and an explicitly represented image model. Object and image models show corresponding *part-of* hierarchies describing aggregation states within a recognition-by-components strategy.

The strategy aims at a step-by-step increase of knowledge during the reasoning. As the domain-specific object models of buildings and building parts define the maximum achievable knowledge about an actual scene and its granularity determines the resolution for knowledge, our model spans hierarchically from the smallest observable features to complex building shapes and allows an adequate evaluation of hypotheses within all different reasoning steps.

We have presented first experimental results. Currently, we are investigating and developing on

- the measurement and propagation of uncertainty within the overall reconstruction process;
- the extension of the current building modeling by more sophisticated knowledge about buildings, especially functional aspects;
- the use of enhanced image models, including sensor characteristics and lighting models within hypotheses verification;
- the derivation of domain-dependent heuristics to constrain search spaces;
- the handling of incomplete observations within all processes and stages of hypotheses generation and verification.

Especially, we consider the integration of logical and statistical knowledge into the framework of constraint logic programming. Furthermore, the approach must be compared to different automatic algorithms for segmentation, 3D reconstruction, and classical photogrammetry. In order to achieve this goal, an attempt is made to find a set of standardized representations about basic building models, parts of buildings, and classes of buildings to ease such comparisons on a conceptual level.

## ACKNOWLEDGMENTS

This work was largely done within the the project “Semantic Modeling and Extraction of Spatial Objects from Images and Maps” especially in the subproject “Building Extraction” which is supported by the Deutsche Forschungsgemeinschaft, DFG. We thank the DFG for supporting our work.

## REFERENCES

1. R. Bergevin and M. D. Levine, Generic object recognition: Building and matching coarse descriptions from line drawings, *IEEE Trans. Pattern Anal. Machine Intell*. **15**(1), 1993, 19–36.
2. I. Biederman, Recognition-by-components: A theory of human image understanding, *Psychol. Rev*. **94**, 1987, 115–147.
3. F. Bignone, O. Henricsson, P. Fua, and M. Stricker, Automatic extraction of generic house roofs from high resolution aerial images, in *Computer Vision—ECCV ’96*, Lecture Notes in Computer Science, No. 1064, pp. 85–96, Springer Verlag, Cambridge, 1996.
4. K. Bowyer and C. Dyer, Aspect graphs: An introduction and survey of recent results, *Int. J. Imaging Systems Technol*. **2**, 1990, 315–328.
5. C. Braun, *Interpretation von Einzelbildern zur Gebäudeerfassung*, Ph.D. thesis, Institut für Photogrammetrie, Universität Bonn, 1994.
6. C. Braun, T. H. Kolbe, F. Lang, W. Schickler, V. Steinhage, A. B. Cremers, W. Förstner, and L. Plümer, Models for photogrammetric building reconstruction, *Comput. Graphics* **19**(1), 1995, 109–118.
7. A. Brunn, E. Gülch, F. Lang, and W. Förstner, A hybrid concept for 3d building acquisition, *ISPRS J. Photogrammetry Remote Sensing* **53**, 1998, 119–129.
8. A. Brunn, F. Lang, and W. Förstner, A procedure for segmenting surfaces by symbolic and iconic image fusion, in *Mustererkennung 1996, 18 DAGM Symposium, Heidelberg, Informatik Aktuell* (B. Jähne, Ed.), pp. 11–20, Springer Verlag, Berlin, 1996.
9. M. Clowes, On seeing things, *Artificial Intell*. **2**, 1971, 79–116.
10. R. Dechter and J. Pearl, Network-based heuristics for constraint satisfaction problems, *Artificial Intell*. **34**, 1988, 370–425.
11. S. J. Dickinson, A. P. Pentland, and A. Rosenfeld, 3-D shape recovery using distributed aspect matching, *IEEE Trans. Pattern Anal. Machine Intell*. **14**(2), 1992, 174–198.
12. S. J. Dickinson, A. P. Pentland, and A. Rosenfeld, From volumes to views: An approach to 3-D object recognition, *CVGIP: Image Understanding* **55**(2), 1992, 130–154.
13. R. Englert, Systematic acquisition of generic 3D building model knowledge, in *Semantic Modeling for the Acquisition of Topographic Information from Images and Maps* (W. Förstner and L. Plümer, Eds.), pp. 181–195, Birkhäuser Verlag, Basel, Switzerland, 1997.
14. A. Fischer and V Steinhage, On the computation of visual events in aspect graph generation, in *Mustererkennung 1997, 19 DAGM Symposium, Braunschweig, Informatik Aktuell*, pp. 156–163, Springer Verlag, Berlin, 1997.
15. A. Fischer, T. H. Kolbe, and F. Lang, Integration of 2D and 3D reasoning for building reconstruction using a generic hierarchical model, in *Semantic Modeling for the Acquisition of Topographic Information from Images and Maps* (W. Förstner and L. Plümer, Eds.), pp. 159–180, Birkhäuser Verlag, Basel, Switzerland, 1997.
16. P. J. Flynn and A. K. Jain, CAD-based computer vision: From CAD-models to relational graphs, *IEEE Trans. Pattern Anal. Machine Intell*. **13**(2), 1991, 114–132.
17. W. Förstner, A framework for low level feature extraction, in *Computer Vision, ECCV ’94, Vol. II* (J.-O. Eklundh, Ed.), Lecture Notes in Computer Science, No. 801, pp. 383–394, Springer Verlag, Berlin, 1994.
18. W. Förstner, Mid-level vision processes for automatic building extraction, in *International Workshop on Automatic Extraction of Man-Made Objects*

*from Aerial and Space Images* (A. Grün, O. Kübler, and P. Agouris, Eds.), pp. 179–188, Birkhäuser, Basel, 1995.

19. W. Förstner and L. Plümer (Eds.), *Semantic Modeling for the Acquisition of Topographic Information from Images and Maps*, Birkhäuser Verlag, Basel, 1997.
20. E. C. Freuder and R. J. Wallace, Partial constraint satisfaction, in *Over-Constrained Systems* (M. Jampel, E. Freuder, and M. Maher, Eds.), Lecture Notes in Computer Science, No. 1106, pp. 63–110, Springer-Verlag, Berlin, 1996.
21. P. Fua and A. J. Hanson, Resegmentation using generic shape: Locating general cultural objects, *Pattern Recognit. Lett.* **5**, 1987, 243–252.
22. P. Fua and A. J. Hanson, An optimization framework for feature extraction, *Pattern Recognit. Lett.* **4**, 1991, 59–87.
23. C. Fuchs and W. Förstner, Polymorphic grouping for image segmentation, in *Proceedings 5th International Conference on Computer Vision, IEEE Computer Society Press, 1995*, pp. 175–182.
24. A. Grün, E. P. Baltsavias, and O. Henricsson (Eds.), *Automatic Extraction of Man-Made Objects from Aerial and Space Images (II)*, Birkhäuser, Basel, 1997.
25. A. Grün, O. Kübler, and P. Agouris (Eds.), *Automatic Extraction of Man-Made Objects from Aerial and Space Images*, Birkhäuser, Basel, 1995.
26. E. Gülch, A knowledge-based approach to reconstruct buildings in digital aerial imagery, in *International Archives of Photogrammetry and Remote Sensing* (L. W. Fritz and J. R. Lucas, Eds.), Vol. 29, B2, pp. 410–417, 1992.
27. C. Hansen and T. C. Henderson, CAD-based computer vision, *IEEE Trans. Pattern Anal. Machine Intell.* **11**, 1993, 1187–1193.
28. R. M. Haralick and G. L. Elliott, Increasing tree search efficiency for constraint satisfaction problems, *Artificial Intell.* **14**, 1980, 263–313.
29. R. M. Haralick and L. G. Shapiro, *Computer and Robot Vision*, Vol. II, Addison–Wesley, New York, 1993.
30. 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*, Vol. 32, Part 3-4W2, pp. 87–96, ISPRS, 1997.
31. O. Henricson, F. Bignone, W. Willuhn, F. Ade, O. Kübler, E. P. Baltsavias, S. Mason, and A. Grün, Project AMOBE: Strategies, current status and future work, in *International Archives of Photogrammetry and Remote Sensing*, Vol. 31, B4, pp. 321–330, 1996.
32. O. Henricsson and E. P. Baltsavias, 3-D building reconstruction with ARUBA: A qualitative and quantitative evaluation, in *Automatic Extraction of Man-Made Objects from Aerial and Space Images (II)* (A. Grün, E. P. Baltsavias, and O. Henricsson, Eds.), pp. 139–148, Birkhäuser, Basel, 1997.
33. M. Herman and T. Kanade, The 3D MOSAIC scene understanding system: Incremental reconstruction of 3D scenes from complex images, in *Readings* in *Computer Vision: Issues, Problems, Principles, and Paradigms* (M. A. Fischler and O. Firschein, Eds.), pp. 471–482, Kaufmann, Los Altos, CA, 1987.
34. A. Heyden, Consistency and correction of line-drawings, obtained by projections of piecewise planar objects, in *Computer Vision—ECCV '94, Proc. of the 3rd Europ. Conf. on Computer Vision*, Lecture Notes in Computer Science, No. 800, pp. 411–419, Springer Verlag, Berlin, 1994.
35. A. Huertas, M. Bejanin, and R. Nevatia, Model registration and validation, in *Automatic Extraction of Man-Made Objects from Aerial and Space Images* (A. Grün, O. Kübler, and P. Agouris, Eds.), pp. 33–43, Birkhäuser, Basel, 1995.
36. D. A. Huffman, Impossible objects as nonsense sentences, *Machine Intell.* **6**, 1971, 295–323.
37. K. Ikeuchi and P. J. Flynn, Recent progress in CAD-based vision, *CVGIP: Image Understanding* **61**(3), 1995, 293–294.
38. J. Jaffar and M. J. Maher, Constraint logic programming: A survey, *J. Logic Program.* **19**/**20**, 1994, 503–581.
39. M. Jampel, E. Freuder, and M. Maher (Eds.), *Over-Constrained Systems*, Lecture Notes in Computer Science, No. 1106, Springer-Verlag, Berlin, 1996.
40. K. Kanatani, *Group-Theoretical Methods in Image Understanding*, Springer-Verlag, Berlin, 1990.
41. J. J. Koenderink and A. J. van Doorn, The internal representation of solid shape with respect to vision, *Biol. Cybernet.* **32**, 1979, 211–216.
42. T. H. Kolbe, L. Plümer, and A. B. Cremers, Using constraints for the identification of buildings in aerial images, in *Proceedings of the 2. Int. Conf. on Practical Applications of Constraint Technology PACT'96 in London*, pp. 143–154, The Practical Application Company Ltd., London, 1996.
43. T. Läbe and K.-H. Ellenbeck, 3D-wireframe models as ground control points for the automatic exterior orientation, in *International Archives of Photogrammetry and Remote Sensing*, Vol. 32, B2, pp. 218–223, 1996.
44. F. Lang and W. Förstner, 3D-city modelling with a digital one-eye-stereo system, in *International Archives of Photogrammetry and Remote Sensing*, Vol. 31, B4, pp. 261–266, 1996.
45. F. Lang and W. Förstner, Surface reconstruction of man-made objects using polymorphic mid-level features and generic scene knowledge, *Z. Photogram. Fernerkundung* **6**, 1996, 193–201.
46. C. Lin, A. Huertas, and R. Nevatia, Detection of buildings using perceptual grouping and shadows, in *Proceedings of CVPR'94*, pp. 62–69, 1994.
47. C. Lin, A. Huertas, and R. Nevatia, Detection of buildings from monocular images, in *Automatic Extraction of Man-Made Objects from Aerial and Space Images* (A. Grün, O. Kübler, and P. Agouris, Eds.), pp. 125–134, Birkhäuser, Basel, 1995.
48. C. L. Lin, Q. Zheng, R. Chelappa, L. S. Davis, and X. Zhang, Site model supported monitoring of aerial images, in *Proceedings of CVPR'94*, pp. 694–699, 1994.
49. A. K. Mackworth, Consistency in networks of relations, *Artificial Intell.* **8**, 1977, 99–118.
50. D. M. McKeown, Towards automatic cartographic feature extraction from aerial imagery, in *Digital Photogrammetric Systems* (H. Ebner, D. Fritsch, and Ch. Heipcke, Eds.), Wichmann, Karlsruhe, 1990.
51. P. Meseguer, Constraint satisfaction problems: An overview, *AICOM* **2**(1), 1989, 3–17.
52. D. Metaxas, S. J. Dickinson. R. C. Munck-Fairwood, and L. Du, Integration of quantitative and qualitative techniques for deformable model fitting from orthographic, perspective and stereo projection, in *Proceedings of the 4th Int. Conf. on Computer Vision, ICCV 1993*, pp. 364–371, 1993.
53. O. Munkelt, Aspect-trees: Generation and interpretation, *CVGIP* **61**(3), 1995, 365–386.
54. R. Nevatia, C. Lin, and A. Huertas, A system for building detection from aerial images, in *Automatic Extraction of Man-Made Objects from Aerial and Space Images (II)* (A. Grün, E. P. Baltsavias, and O. Henricsson, Eds.), pp. 77–86, Birkhäuser, Basel, 1997.
55. M. Pilu and R. B. Fisher, Snakes and splines for tracking non-rigid heart motion, in *Computer Vision–ECCV '96*, Lecture Notes in Computer Science, No. 1064, pp. 71–82, Springer-Verlag, Cambridge, 1996.
56. L. Quam and T. M. Strat, SRI image understanding research in cartographic feature extraction, in *Digital Photogrammetric Systems* (H. Ebner, D. Fritsch, and C. Heipke, Eds.), pp. 111–121, Wichmann, Karlsruhe, 1991.
57. J. Rissanen, Minimum-description-length principle, *Encyclopedia Stat. Sci.* **5**, 1987, 523–527.
58. M. Roux and H. Maitre, 3-D description of urban areas using maps and aerial images, in *Automatic Extraction of Man-Made Objects from Aerial and Space Images (II)* (A. Grün, E. P. Baltsavias, and O. Henricsson, Eds.), pp. 311–322, Birkhäuser, Basel, 1997.

59. W. Schickler, Towards automation in photogrammetry, *Geo. Inform. Mag.* **7**(4), 1993, 32–35.
60. M. Sester and W. Förstner, Object location based on uncertain models, in *Mustererkennung 1989, 11 DAGM Symposium Hamburg* (H. Burkhardt, K.-H. Höhne, and B. Neumann, Eds.), *Informatik Fachberichte*, Vol. 219, pp. 457–464, Springer-Verlag, Berlin, 1989.
61. L. G. Shapiro and R. M. Haralick, A metric for comparing relational descriptions, *IEEE Trans. Pattern Anal. Machine Intell.* **7**(1), 1985, 90–94.
62. V. Steinhage, On the integration of object modeling and image modeling in automated building extraction from aerial images, in *Automatic Extraction of Man-Made Objects from Aerial and Space Images (II)* (A. Grün, E. P. Baltsavias, and O. Henricsson, Eds.), pp. 139–148, Birkhäuser, Basel, 1997.
63. P. Suetens, P. Fua, and A. J. Hanson, Computational strategies for object recognition, *ACM Comput. Surveys* **24**(1), 1992, 5–61.
64. K. Sugihara, *Machine Interpretation of Line Drawings*, MIT Press, Cambridge, MA, 1986.
65. P. van Hentenryck, *Constraint Satisfaction in Logic Programming*, Logic Programming Series, MIT Press, Cambridge, MA, 1989.
66. G. Vosselman, *Relational Matching*, Lecture Notes in Computer Science, No. 628, Springer-Verlag, Berlin, 1992.
67. R. J. Wallace and E. C. Freuder, Heuristic methods for over-constrained constraint satisfaction problems, in *Over-Constrained Systems* (M. Jampel, E. Freuder, and M. Maher, Eds.), Lecture Notes in Computer Science No. 1106, pp. 207–216, Springer-Verlag, Berlin, 1996.
68. D. L. Waltz, Understanding line drawings of scenes with shadows, in *Psychology of Computer Vision* (P. H. Winston, Ed.), pp. 19–91, McGraw–Hill, New York, 1975.
69. U. Weidner, An approach to building extraction from digital surface models, in *International Archives of Photogrammetry and Remote Sensing*, Vol. 31, B3, pp. 924–929, 1996.
70. U. Weidner and W. Förstner, Towards automatic building extraction from high resolution digital elevation models, *ISPRS J. Photogram. Remote Sensing* **50**(4), 1995, 38–49.

Statement of ownership, management, and circulation required by the Act of October 23, 1962, Section 4369, Title 39, United States Code: of

COMPUTER VISION AND IMAGE UNDERSTANDING

Published monthly by Academic Press, 6277 Sea Harbor Drive, Orlando, FL 32887-4900. Number of issues published annually: 12. Editor: Dr. Avinash C. Kak, Robot Vision Lab, 1285 EE Building, Purdue University, West Lafayette, IN 47907.
Owned by Academic Press, 525 B Street, Suite 1900, San Diego, CA 92101-4495. Known bondholders, mortgagees, and other security holders owning or holding 1 percent or more of total amount of bonds, mortgages, and other securities: None.
Paragraphs 2 and 3 include, in cases where the stockholder or security holder appears upon the books of the company as trustee or in any other fiduciary relation, the name of the person or corporation for whom such trustee is acting, also the statements in the two paragraphs show the affiant's full knowledge and belief as to the circumstances and conditions under which stockholders and security holders who do not appear upon the books of the company as trustees, hold stock and securities in a capacity other than that of a bona fide owner. Names and addresses of individuals who are stockholders of a corporation which itself is a stockholder or holder of bonds, mortgages, or other securities of the publishing corporation have been included in paragraphs 2 and 3 when the interests of such individuals are equivalent to 1 percent or more of the total amount of the stock or securities of the publishing corporation.
Total no. copies printed: average no. copies each issue during preceding 12 months: 1668; single issue nearest to filing date: 1633. Paid circulation (a) to term subscribers by mail, carrier delivery, or by other means: average no. copies each issue during preceding 12 months: 414; single issue nearest to filing date: 406. (b) Sales through agents, news dealers, or otherwise: average no. copies each issue during preceding 12 months: 639; single issue nearest to filing date: 634. Free distribution (a) by mail: average no. copies each issue during preceding 12 months:54; single issue nearest to filing date: 54. (b) Outside the mail: average no. copies each issue during preceding 12 months: 23; single issue nearest to filing date: 23. Total no. of copies distributed: average no. copies each issue during preceding 12 months: 1130; single issue nearest to filing date: 1117. Percent paid and/or requested circulation: average percent each issue during preceding 12 months: 93%; single issue nearest to filing date: 93%.

(Signed) Patricia Butler, Director of Circulation