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

André Fischer[1], Thomas H. Kolbe[4], Felicitas Lang[2]
Armin B. Cremers[3], Wolfgang Förstner[2], Lutz Plümer[4], Volker Steinhage[1]

Computer Science Department I[1]/III[3], Institute for Photogrammetry[2], University of Bonn
Institute for Environmental Sciences[4], University of Vechta

**Abstract**

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 multi-level 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 a reliable building extraction.

**Keywords:** explicit 2D-modeling, coupling of 2D- and 3D-modeling, multilayer aggregation, building modeling, multi image correspondence analysis, mid-level feature aggregates, aspect hierarchies, constraint logic programming.

## 1 Introduction

Due to the fact that more than about 50% of the world population live in urban or suburban environments the automation of 3D building extraction is an issue of high importance and shows an increasing need for various applications including geo-information 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 overwhelming 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 (cf. [58]).

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 to integrate 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 (e.g. [7], [32], [63], [59], [36], [30]). Obviously polyhedral descriptions are too general for the use within 3D building extraction and therefore move the burden of building modeling on additional representation schemes to represent and organize domain specific heuristics and constraints like in the MOSAIC system (cf. [29]) or in the approach of [4]. Parameterized models are restricted to describe the most common building types in the sense of prototypes (cf. [45], [51], [39], [41], [43], [42], [49]) but show a lack to represent variations and combinations of their shapes as well as other relations. Prismatic models can describe arbitrary complex polygonal ground plans of buildings, but reveal the strong restriction to buildings with only flat roofs (cf. [65], [64], [53]). CAD models are used to describe objects with fixed geometry and topology in object *recognition* tasks, especially for controlling industrial processes (cf. [24], [14], [33], [48]). The use of CAD models in building extraction is therefore restricted to the identification of a priori known buildings (cf. [55], [54], [31]).

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. [19] employ simple box-type primitives but propose an explicit representation of legal primitive combinations to more complex building aggregates. The approaches of [10] and [1] are from 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 (cf. [2]). Both approaches employ volumetric primitives instead of simple box-types but neglect the description of elaborated schemes for domain dependent primitive combinations. Furthermore the reliable extraction of their primitives from real images is concern of current research (cf. [47], [50]). [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. [27] adopt the modeling approach of [3], but extracting image regions with homogeneous photometric and chromatic properties by navigating through a constraint triangulation network where each extracted line segment coincide with edges of the triangles. [28] present impressive results of their approach on some test data but obviously show no explicit modeling of building types and building specific aggregation schemes. Groups of planar 3D patches 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 recommand the proceedings of the two workshops on "Automatic Extraction of Man-Made Objects from Aerial and Space Images" (cf. [22], [21]) as well as the proceedings of the workshop on "Semantic Modeling for the Acquisition of Topographic Information from Images and Maps" (cf. [17]).

## 1.2 Overview

In [5] we proposed in detail the concepts and processes which have to be taken into account for a sufficient 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. Within this paper now 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 *is-part-of*-hierarchy of 3D building parts and their corresponding projective descriptions. The reconstruction process is carried out already 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 back projected into the images to allow a component-based and robust hypothesis verification applying constraint solving techniques (cf. [37]).

First results of our approach were presented at the SMATI workshop in Bonn (cf. [13]). 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 reveals a great potential for a reliable building extraction.

## 2 Concept

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

### 2.1 Models

For coping with the complexity of natural scenes we propose an application specific modeling of the domain *buildings*. The general concept has been presented in [5]. It 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 the fundamental role for our approach to a generic building model. This model is not based on a polyhedral scheme (cf. [29, 28]) 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 etc. 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.
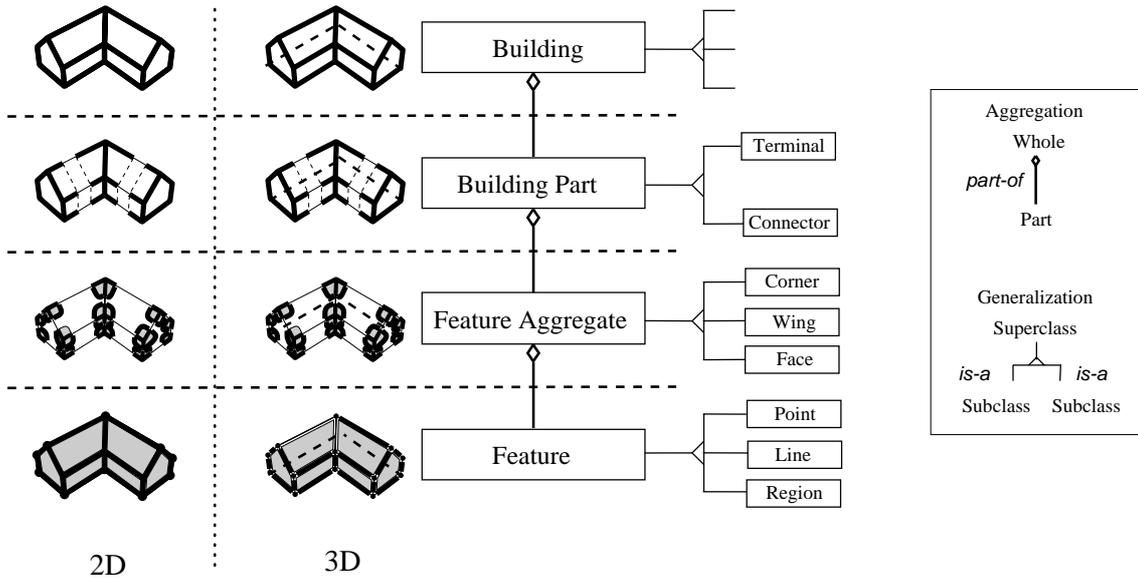
Figure 1: Building model: The different semantic levels of the part-of hierarchy are shown in vertical direction, the different levels of abstraction of the is-a hierarchy in horizontal direction, which is only shown for the 3D-model. The 2D-image model describes the expected appearance of the building in the different levels of the part-of hierarchy, which is indicated by not showing the hidden lines.

### 2.1.1 Object Model

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

We therefore propose a multi-level *part-of hierarchy* (cf. figure 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 one hand, describing position and orientation, and form parameters on the other hand 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).
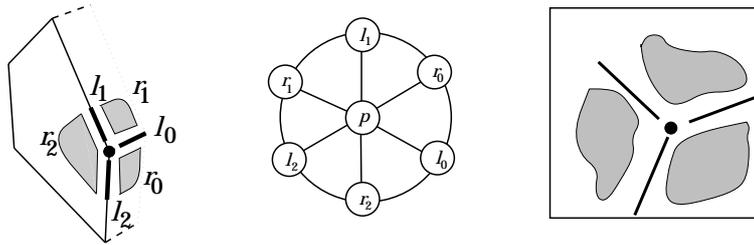
4

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

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 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 (s. 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 primitives to each other. We discriminate *terminals* having exactly one plug face and *connectors* with two or more plug faces (cf. figure 3).

Terminals:                                    Connectors:
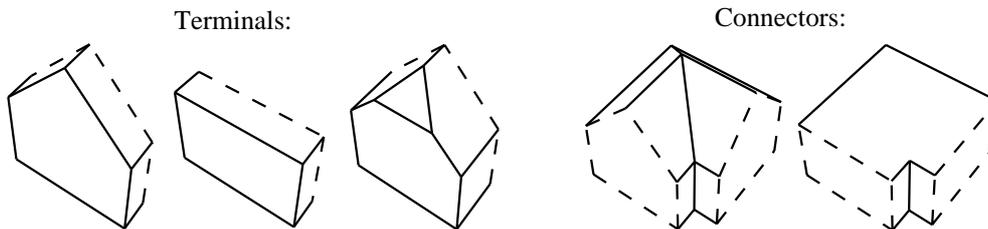


Figure 3: Some examples of building parts. Plug faces, which are used to connect them, are drawn dashed.

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 which 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. E.g. a 2D-corner, being a point induced image aggregate inherits the

---

[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.

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 region, 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 of sensor characteristics, illumination and low-level feature extraction methods. The results, presented in this paper, are derived by using a pinhole camera as sensor model and assuming weak perspective projection. The potential role of the integration and usage of an illumination model is discussed elsewhere (cf. [57]) and will only be sketched in part 3.3.

The image model is currently used for both, the multi view 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, to verify intermediate aggregation results and to estimate certain parameters by monocular analysis, e.g. of shadows and wall projections (cf. [49]).

## 2.2   Strategy

Our input data are 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 (cf. [15], [20]). It allows to derive point, line, and region neighborhood aggregates $A^{2D}$, where vertices $V^{2D}$ are the most promising ones 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 succesful (cf. [28, 27]). 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 (vertices + class label) 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 include also line and region induced feature aggregates, i.e. wings and faces.

Now, our current strategy consists of three main tasks, which are executed subsequently (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, 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 corners hypotheses of different images are used to derive 3D corner reconstructions by bundle adjustment. The 3D corner 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
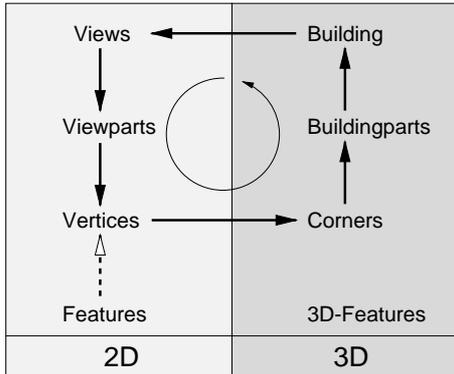
Figure 4: Information transfer within the whole process. The close integration of 2D and 3D reasoning is performed by an iteration loop. The dashed arrow marks the initialization step. 3D-reconstruction, generation and verification of hypotheses for building parts and buildings is repeated. Verification is based on generated views using matching on the feature level.

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 have to be aggregated to complete 3D building hypotheses. This aggregation step is implemented by successive steps of merging and connecting building parts.

3) **3D → 2D: verification of building hypotheses.** For the verification the 3D building hypotheses are projected back into the original aerial images, resulting in parameterized views for buildings. Due to incomplete results of feature extraction, not all parameters of building parts and the aggregated building hypotheses may be instantiated. Thus, we call the backprojection parameterized views and implement these views in terms of a modified aspect notation (cf. Fig. 1). Hypotheses are comfirmed by a voting process according how close the projected features of their parametrized views are to the originally extracted image features. Furthermore, geometric reasoning in the images and with projective geometry may derive previously undected image features and may determine free object parameters of the building hypotheses.

The successful sequence of matching steps results in an iteratively improved gain in knowledge. The 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, initiating a second iteration of 3D-reconstruction, generation of building hypotheses and verification.

## 3  Models and Operations

Within this section, we describe in detail the three main steps of our approach to building reconstruction as proposed in 2.2, i.e. (1) 2D → 3D: reconstruction of 3D corners, (2) 3D → 3D: generation of building hypotheses, (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, (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.

Our experience with fully automatic subprocesses for the purpose of 3D reconstruction has shown a high stability while using image patches with multiple overlap (cf. [39], [38]). It motivates us in using all available image data simultaneously. 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 chapter 2.1. For reconstruction we perform the transition from image space to object space on the level of feature aggregates. Therefore we need to model corners in object space as well as their appeerence in the images using the corner model in 2D and in 3D.

**3D Corner Model:** The representation of corners is subdivided into the geometric information and the domain specific class labels $\omega_c$. Each corner $c_i^{3D} = (v_i^{3D}, \omega_c)$ 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 defined corner classes $\Omega_C$ (cf. chapter 2.1).

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 depending 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 due to their slope orientation with respect to the corner point, given by the qualitative geometric labels (h), (v+), (v-), (o+) and (o-) (cf. [23]). We exclude meaningless corners as e.g. a corner with line attributes ( (h), (h), (h) ), which make no sense in the context of buildings and their functionality. For example the first specialization level compounds 21 possible classes of corners with node degree 3.

This description is further refined by the second level of the specialization hierarchy. It uses binay 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 or (orthogonality) of two lines. The explicit definition of subclasses on this level is not sensible as we cannot be capable to predefine it without restricting the variability of buildings.

If there are no constraints attached to the corner we call it the unconstrained corner with the class label $\omega_0$.

Examples of different corner classes are given in figure 5. The typical corner at the ridge of a hip roof (cf. fig. 5, corner no. 7) is given by the line attributes { (h), (o-), (o-), symmetry ((o-),(o-)) }. The distiction between corner no. 7 and corner no. 5 is due to the second level of the specialization hierarchy as on the first level they both are described by the line attributes ( (h), (o-), (o-) ).
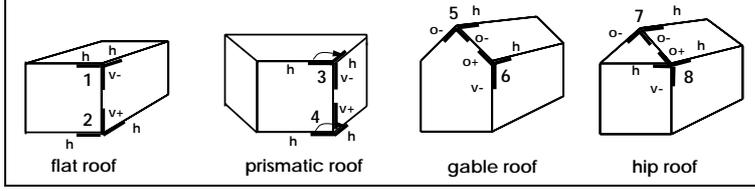
Figure 5: shows 8 examples of corner classes $\omega_c$, which are sufficient for describing the building types {`flat roof`, `prismatic roof`, `gable roof` and `hip roof`}.

**2D Corner Model:** The 2D corner model aims at giving access to 3D corners starting with the image observations. Because of the higher expressiveness of the 3D vertex geometry in contrast to 2D, the assignement of a corner class label $\omega_c$ to the geometric vertex elements can much easier be performed in object space.

Therefore we propose using a 2D corner model without any distinction into subclasses to find point induced vertex aggregates $V^{2\mathrm{D}}$ which have the structural characteristics of corner projections into the images (cf. fig. 2). These characteristics define the vertex class $\omega_v$. We statistically model the probability of a vertex being a projected corner by using the conditional probabilities $P(\omega_v|v^{2\mathrm{D}})$ of the vertex class $\omega_v$ given the vertex observation $v^{2\mathrm{D}}$. It is used for classifying point induced feature aggregates being a vertex. The vertex class $\omega_v$ describes the ideal corner projection using specific geometric and structural characteristics. Please note that the transition of vertices to corner elements additionally requires the assignment of a class label $\omega_c$.

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

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

By correspondence analysis in the images $i \in I$ we derive the vertex correspondence set $[v_i^{2\mathrm{D}}] = (v_1^{2\mathrm{D}}, \dots v_I^{2\mathrm{D}})$ which is then used for the transition to 3D vertices $V^{3\mathrm{D}}$. They represent the geometry of corners $C^{3\mathrm{D}}$ and thus serve for the subsequent interpretation by assigning the vertex $v^{3\mathrm{D}}$ to a corner class $\omega_c$. The result of this step are corner hypotheses $c_i^{3\mathrm{D}} = (v_i^{3\mathrm{D}}, \omega_{c,i})$.

In the following we give a detailed description of the 3D vertex- 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 $V^{2\mathrm{D}}$ which is build up by evaluating their suitability for the correspondence analysis and reconstruction by statistical classification which is based on the 2D corner model. The priority list is given by minimizing the information $I(\omega_v \mid v^{2\mathrm{D}})$ for the vertex class $\omega_v$ while observing the vertex $v^{2\mathrm{D}}$. The information $I(\omega_v \mid v^{2\mathrm{D}})$ is derived using the probability $P(\omega_v \mid v^{2\mathrm{D}})$ with $I(\omega_v \mid v^{2\mathrm{D}}) = -lbP(\omega_v \mid v^{2\mathrm{D}}) = -\ln P(\omega_v \mid v^{2\mathrm{D}})/\ln 2$.

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

$$P(\omega_v|\boldsymbol{b},\boldsymbol{c}) = \frac{\prod_{k=1}^{n_k} P(b_k|\omega_v) \cdot p(\boldsymbol{c}|\omega_v)}{\prod_{k=1}^{n_k} P(b_k) \cdot p(\boldsymbol{c})} \cdot P(\omega_v). \tag{1}$$

With the discrete characteristics $b_k$ with $k = 1, \dots n_k$ and the continuous characteristics $c_j$ with $j = 1, \dots n_j$ being uncorrelated with mean $\mu_j$ and variance $\sigma_j^2$ this leads to

$$I(\omega_v | \boldsymbol{b}, \boldsymbol{c}) = \frac{1}{2\ln 2}\left[\sum_{k=1}^{n_k} -2\ln P(b_k|\omega_v) + \sum_{j=1}^{n_j}\left(\frac{c_j - \mu_j}{\sigma_j}\right)^2\right] \tag{2}$$

The characteristics consider the stability, uniqueness and structural richness of the vertex and thus models the uncertainty while observing 2D corners. Actually we use the number of lines and the number of 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. The cost function introduces in addition to the information of single vertices a score value which describes the relation of vertex components to the epipolar plane.

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 facilitates the matching of the features $F^{2D}$.



reconstructed corner

vertex acces by following reconstructed lines
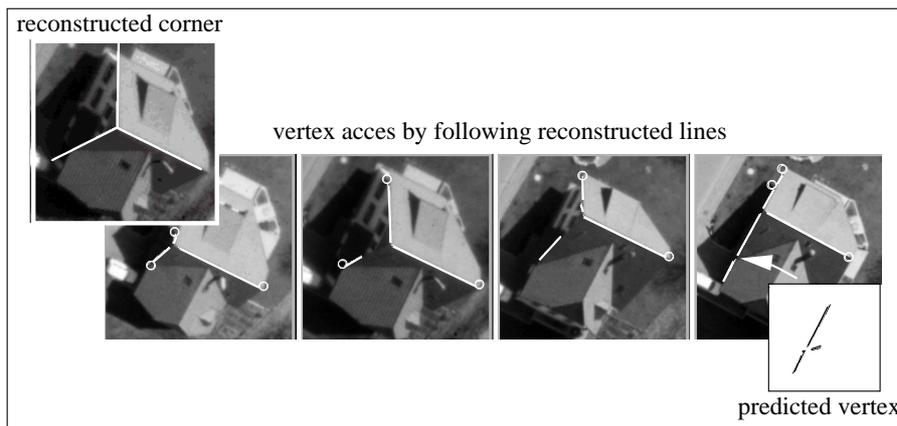
predicted vertex

Figure 6: shows the information propagation for selecting and predicting vertices for the next reconstruction step. 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 wich are used for the reconstruction of the previous corners. The right most image shows the top-down prediction of a vertex generated by line intersection where no access via the feature adjacency graph was possible.

The search space for the next correspondence set is defined by the neighbourhood relations of the corresponding image features of the reconstructed features $F^{3D}$ using the prolongation of lines and the neighbourhood of faces. For instance we have to 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. In case we thus do not find appropriate vertices, the neighbourhood of faces are used for vertex access by analysing the region-induced aggregates.

In addition to vertices which are directly derived using the feature adjacency graph we also perform a top-down prediction of additional vertices. We therefore select a pair of possibly corresponding vertices $(v_i, v_j)$. By forward intersection we derive the 3D position of the vertex point which we propagate to the expected vertex positions in the remaining images by back-projection. At these positions we then can perform a well-aimed search for line intersections to generate additional 2D vertices. The prediction helps to bridge the incompleteness of the extracted features or feature adjacency relations without unnessesarily 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 hierachical 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. [16]).

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

In case the test accepts no constraints $\Omega''_C$ is the empty set and the underlying vertex represents an unconstrained corner of class $\omega_0$.

Depending on the selected constraints we complete the reconstructed vertices to corners using the most probable corner class $\omega_c$ which contains the identified vertex constraints. Therefore we hypothesize additional corner components of feature type line $L$. In the most simple case we just index into the corner classes (cf. fig. 5) E. g. if we observe a reconstructed vertex of node degree 2 with line labels (h) and (o-) with the two lines further fulfilling the constraint `orthogonal(h,o-)` , we perform a model based prediction of a corner of node degree 3 of class (h,o-,o-) where the line pair (o-,o-) fulfills the constraint `vertical symmetry(o-,o-)` as it occurs for the corner at the ridge of a gabled roof building with the lines (o-,o-) spanning a vertical plane (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 [11] and can be introduced for selecting the most probable corner class $\omega_c$.

This way we may obtain several corner hypotheses $c_j^{3D} = (v^{3D}, \omega_{c,j})$ for each vertex reconstruction $v^{3D}$ which have to be resolved in the verification step.

**Verification of Corner Hypotheses:** For verification of the corner hypothese, 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 $[v_i^{2D}]$ 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
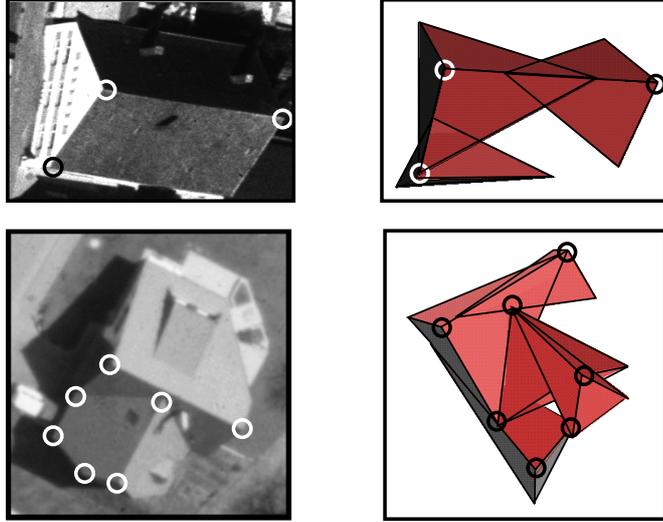
Figure 7: Examples of 3D corner reconstruction. From left to right: **a.** feature aggregates marked in one image, **b.** reconstructed 3D corners with each corner plane shown as triangle (please note that the corner planes are open ended, the drawn lines support the visualization).

$$\hat{c} = arg\max_{c_j} P(c_j \mid [v_i^{2\mathrm{D}}]) \quad \propto \quad arg\max_{c_j} P([v_i^{2\mathrm{D}}] \mid c_j)P(c_j) \tag{3}$$

The a priori probability $P(c_j)$ for the corner class $\omega_{c,j}$ can be obtained empirically by learning (cf. [11]) and can in principle be integrated to evaluate the model dependent influence on the result.

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

$$\hat{c}, \hat{\boldsymbol{\beta}} \quad \propto \quad arg\max_{c_j} P([v_i^{2\mathrm{D}}] \mid c_j, \boldsymbol{\beta}_j)P(\boldsymbol{\beta}_j \mid c_j)P(c_j) \tag{4}$$

$$\propto \quad arg\max_{c_j} P([v_i^{2\mathrm{D}}] \mid \boldsymbol{\beta}_j)P(c_j) \tag{5}$$

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* $p([v_i^{2\mathrm{D}}] \mid \boldsymbol{\beta})$ in case the features exist and have been successfully matched to the model.

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

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

To decide for the the optimal interpretation and its corresponding constraints which define the unknowns $\boldsymbol{\beta}$, we first test for acceptance of the reconstruction of the unconstained corner of class $\omega_0$ with $\Theta_{\omega_0}^{3\mathrm{D}} = \emptyset$. If this test is successful, we test for the class specific constraints $\Theta_{\omega_c}^{3\mathrm{D}}$. Both tests use a Fisher distributed testvalue depending on the residuals[2] $\Omega = (\boldsymbol{y}-\hat{\boldsymbol{y}})^T \boldsymbol{\Sigma}_{yy}^{-1}(\boldsymbol{y}-\hat{\boldsymbol{y}})$ 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 given while corner hypotheses generation.

| | | $n$ | $u$ | $r$ | $\Omega[\;]$ | $\sigma_0[\;]$ | $\sigma_x[m]$ | $\sigma_y[m]$ | $\sigma_z[m]$ |
|---|---|---|---|---|---|---|---|---|---|
| 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 |

Table 1: shows for two examples the result of parameter estimation using the class depending constraints or no constaints. The precision of the reconstructed vertex point is increased while using the constraints for stabilization. ($n$ is the number of observations, $u$ the number of unknown parameters and $r$ the redundancy of the estimation system).

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 $\Theta_{\omega_c}^{3\mathrm{D}}$ are accepted. They are useful for geometric stabilization of the reconstruction as shown with the decrease of the empirical standard deviation $\sigma_x, \sigma_y$ and $\sigma_z$ of the three corner coordinates $(x, y, z)$ while using the constraints (cf. tab. 1).

Further details of the corner reconstruction approach can be found in (cf. [40], [6]).

The result of the estimation are evaluated corner reconstructions $c^{3\mathrm{D}} = (v^{3\mathrm{D}}, \omega_c)$ constrained by $\Theta_{\omega_c}^{3\mathrm{D}}$. 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 $C^{2\mathrm{D}}$ (cf. section 3.3.2), in addition to the original vertex data and thus expect an increasing number of reconstructed corners.

## 3.2  3D $\rightarrow$ 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 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 aggregates describing complete building hypotheses.

---

[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 $\omega_c \in \Omega_c$.

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

### 3.2.1 Models of Buildings and Building Parts

Given a set $C^{3D} = \{c_1^{3D}, \ldots, c_n^{3D}\}$ of $n$ reconstructed corner aggregates, one or more building hypotheses have to 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 to more complex ones and finally to complete buildings. In the current state of our work the set of building part models is selected manually. In the future, learning schemes as described by [11] will be used.

**Plug Faces, Terminals, 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 (see Figure 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.

**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 (s. figure 8). If a parameter has an assigned value, it is called *fixed*, otherwise *free*. Constraints on the parameters restrict them to valid values.
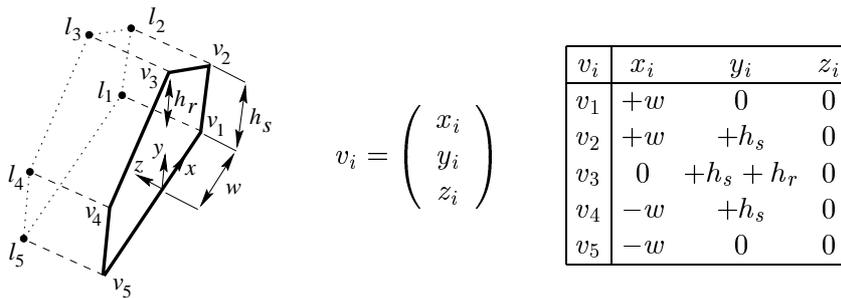


$$v_i = \begin{pmatrix} x_i \\ y_i \\ z_i \end{pmatrix}$$

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

Figure 8: Parameterised description of a hip roof terminal. The table on the right shows the coordinates of the vertices $v_1, \ldots, v_5$.

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. The result of maximum likelihood estimations used for the determination of the hypotheses parameters are used to define an order on them.

### 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 to organize 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 a one partially instantiated building part for the reconstructed 3D corner $c$. Location and orientation of the reconstructed corner $c$ will fix also position and orientation of the indexed building parts. Furthermore, 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.

Indexing finds for a given corner observation $c_i$ all possible mappings $\phi$ to all corners $c_j$ of all building part templates, 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, 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 consistent setting of the fixed parameters of shape (cf. Fig 10). All corners covered by $b_1$ and $b_2$ are also covered by the resulting building part.
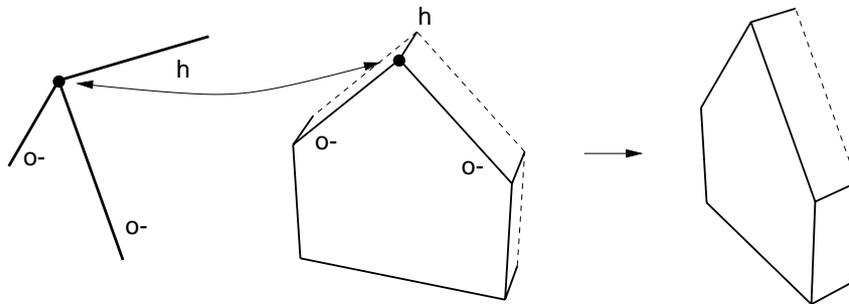


Figure 9: Indexing establishes a link between corner observations (left) and building part primitives (center) resulting in an instantiated building part (right).

`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 $f_2$ (ref. 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 does not fit, they may not be aggregated.

All corners covered by $b_1$ and $b_2$ are also covered by the resulting building part.
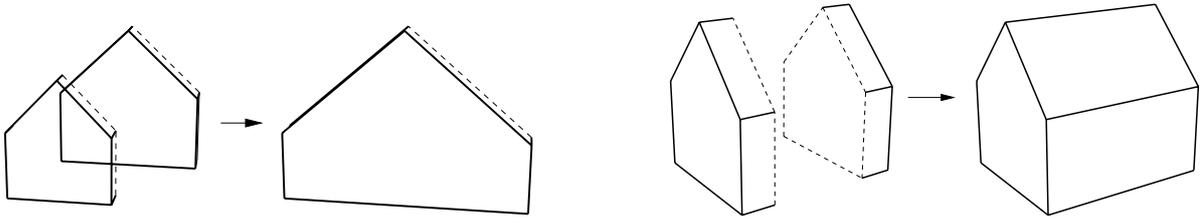


Figure 10: Left: Two saddle roof terminals are merged together. Parameters are adjusted. Right: Two saddle roof terminals are connected together. The resulting building part is closed.

Each operation is followed by a maximum likelihood parameter estimation in form of a chi-square least-squares fitting. This parameter estimation determines in case of a merging or connection operation, whether the parameters of the two participating building parts are compatible or not. In 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 way 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 get constructed, which are not already in $B_i$. Therefore $B_i$ and $N_i$ are disjunct. 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 reconsturcted corner. The number of hypotheses contained in $B_n$ is exponential in $n$.

We use an approach that employs a heuristic that typically allows the construction of the best building hypothesis in much fewer 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. Afterwards 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 that 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 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: 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 minimised in the least-squares fitting performed after the construction of $b_{i,1}$ and $b_{i,2}$: 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 initialised 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 $\mathtt{Merging}(b, b_i)$, $1 \leq i \leq k$, 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 $\mathtt{Connection}(b, b', f, f')$ into the priority queue.

An operation is inserted only if its result is not yet 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 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 and two merging 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 for visualisation a default value has been chosen.
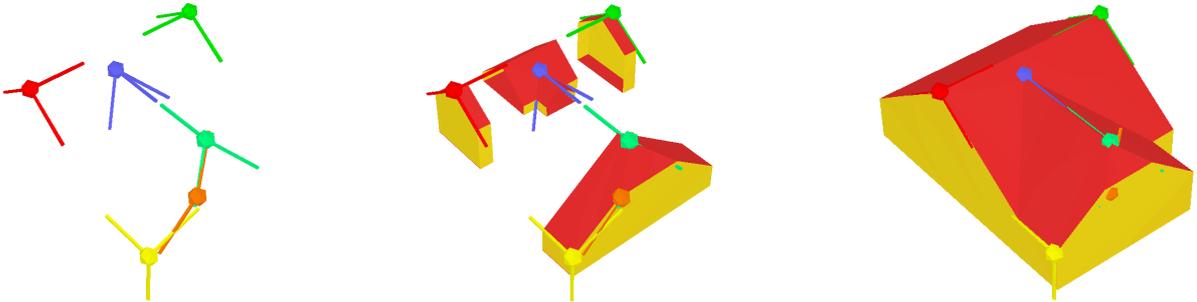


Figure 11: On the left the given corner observations are shown. The figure in the middle shows the best fitting building parts after all indexing steps and some merging steps (building part in the front). These building parts are aggregated in three operations resulting in the building hypothesis shown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

## 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 aggregates. Each aggregation level can be accompanied by corresponding aspect descriptions (cf. [12]) 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 derive hypotheses verification in an efficient way. We have currently implemented our verification procedure for complete—but also only partially instantiated—building hypotheses. But obviuously, our verification procedure can by applied also 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 proposed by [9]. Aspect hierarchies describe qualitative image models of volumetric primitives in terms of aspects. Each aspect is decomposed into a hierarchical feature-based description, 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 appearences 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 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 (s. fig. 12). Due to the reliable processes of feature extraction and reconstruction, the number of free parameters is generally low (in the worst case we have up to three free parameters of shape).

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. The image models of the hypotheses of buildings and build parts are called *parameterized view hierarchies* due to their independency from viewpoint. For each aerial image one parameterized view hierarchy is generated. It consists of three levels:

**Level 1:** On the highest level contains all three dimensional building hypotheses. Note that a hypothesis may still contain 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 like parallelisms, symmetries etc. 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 (ref. 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 which employ a standard lighting model including ambient light and diffuse reflection, where the sun vector is computed from sensor model and time stamp. The analysis of illumination and shadows allows to derive 3D object parameters and to test the consisteny of shadow-ground transitions and inter-surface intensities (cf. [57]).

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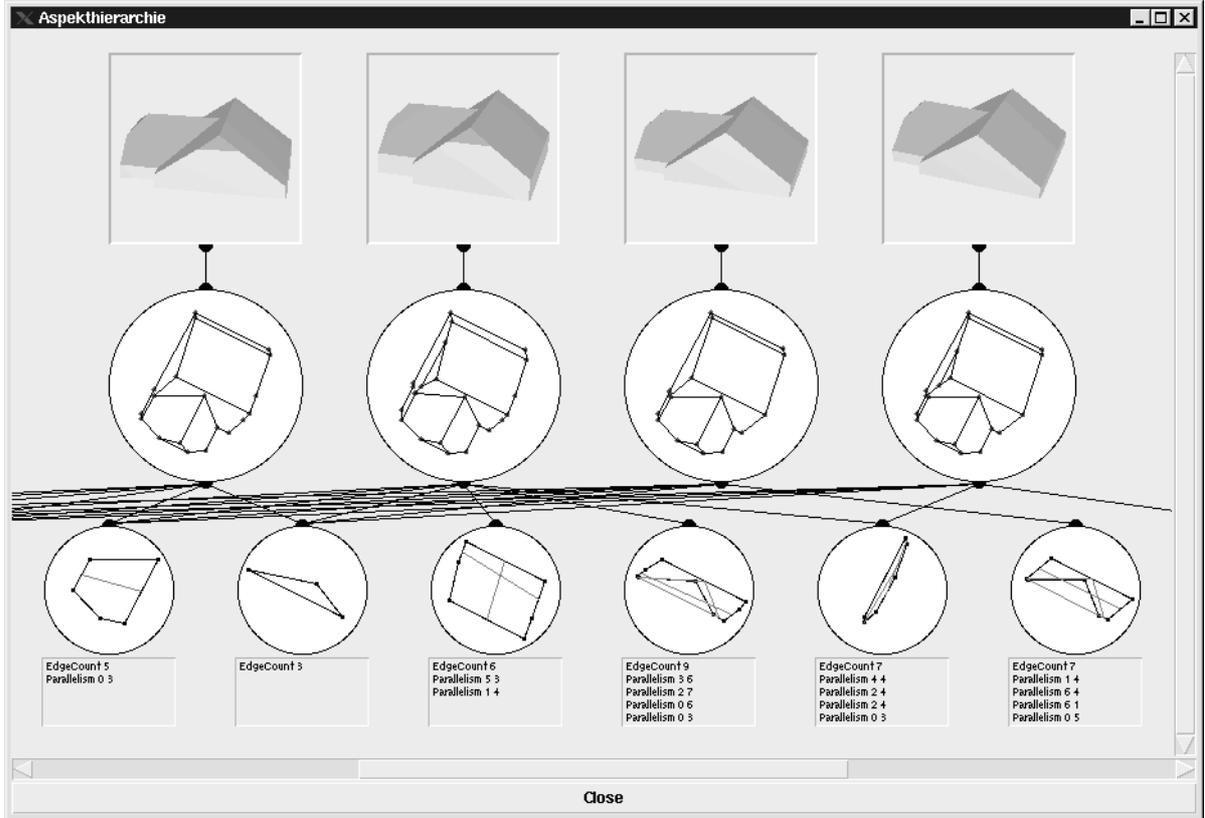
Figure 12: Example of a view hierarchy. For one set of corner reconstructions four different building hypotheses have been generated due to two different indexing results into the library of building parts for the leftmost ridge corner and the free parameter of height (no ground corner was reconstructed).

hand. Since this matching process considers objects and their interrelationships, this task is an instance of relational matching [61]. We employ constraint solving methods (cf. [46]) to handle 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 on 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) (cf. [60], [34]) is explained in detail in [37].

**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

20

our model hierarchy shown in figure 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 have to be satisfied simultaneously by the corresponding objects. A consistent match, also called consistent labeling [26], is an identification of a set of extracted image features which satisfy 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.
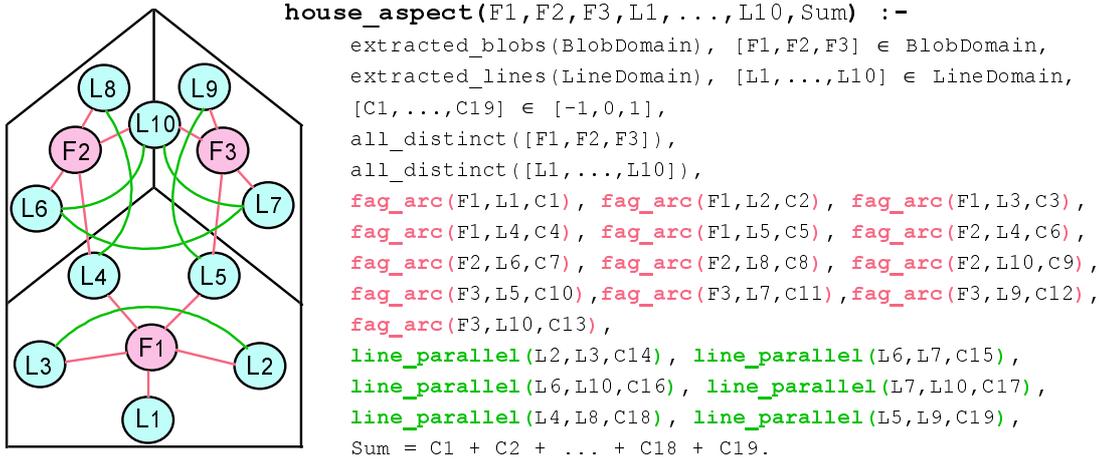


```
house_aspect(F1,F2,F3,L1,...,L10,Sum) :-
    extracted_blobs(BlobDomain), [F1,F2,F3] ∈ BlobDomain,
    extracted_lines(LineDomain), [L1,...,L10] ∈ LineDomain,
    [C1,...,C19] ∈ [-1,0,1],
    all_distinct([F1,F2,F3]),
    all_distinct([L1,...,L10]),
    fag_arc(F1,L1,C1), fag_arc(F1,L2,C2), fag_arc(F1,L3,C3),
    fag_arc(F1,L4,C4), fag_arc(F1,L5,C5), fag_arc(F2,L4,C6),
    fag_arc(F2,L6,C7), fag_arc(F2,L8,C8), fag_arc(F2,L10,C9),
    fag_arc(F3,L5,C10),fag_arc(F3,L7,C11),fag_arc(F3,L9,C12),
    fag_arc(F3,L10,C13),
    line_parallel(L2,L3,C14), line_parallel(L6,L7,C15),
    line_parallel(L6,L10,C16), line_parallel(L7,L10,C17),
    line_parallel(L4,L8,C18), line_parallel(L5,L9,C19),
    Sum = C1 + C2 + ... + C18 + C19.
```

Figure 13: Constraint representation of a building hypothesis for a saddleback roof house. The **fag_arc**-constraints express the adjacencies between lines and regions. The indicator variables $C_1, \ldots, C_{19}$ reflect the status of each constraint: $-1$ means violated, 0 relaxed, and 1 satisfied (see chapter 3.3.2).

**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 [44, 25, 8, 60] 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 all their incident constraints are satisfiable. Thus, the resulting CSPs

are in general over-constrained [35]. The approaches of MaxCSP as proposed by Freuder and Wallace [18, 62] or inexact graph matching described by Haralick and Shapiro [56, 26] 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 in [61]. If an expected model feature is missing in the image, the corresponding variable has to be assigned a wildcard value, and all its incident constraints have to 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

$$c'(t_1, \ldots, t_n, b) \Leftrightarrow (b = 0) \ \lor \ ((b = 1) \land c(t_1, \ldots, t_n)) \ \lor \ ((b = -1) \land \neg c(t_1, \ldots, t_n)) \quad (7)$$

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 $-1$), $c$ has to 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 $c'_j(\ldots, v_i, \ldots, b_j) \in \mathcal{C}'$ has to 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 has to 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

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

using a branch-and-bound algorithm embedded in the CLP context [60]. 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 have to 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.
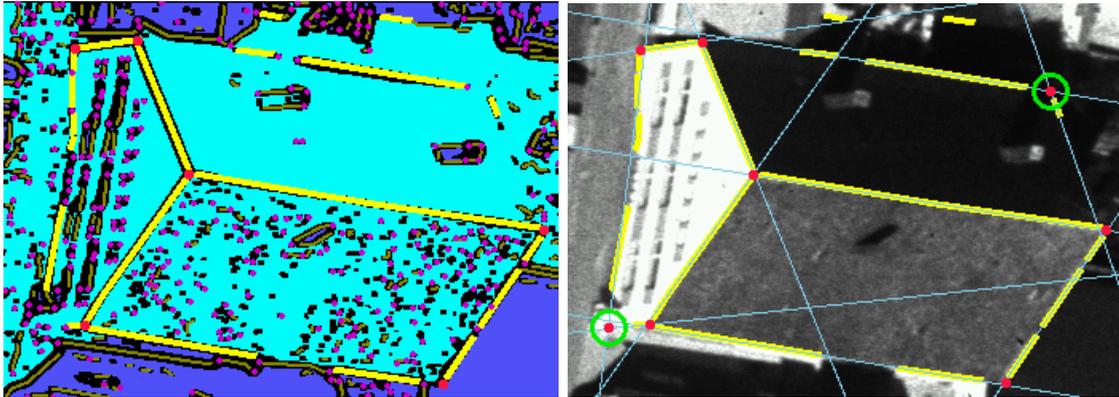
Figure 14: Left: Matched image features, right: new 2D corners

**Generation of New 2D Corners:** The solution of the CSP in general fixes further parameters of the building hypotheses. It has to 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, finally are detected in 2D as illustrated by the right picture in figure 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 and 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.

## 4 Results

The current result of the reconstruction procedure is presented for the international test data set which was distributed from the ETH Zurich for the Ascona Workshop 1995 on *Automatic Extraction of Man-Made Objects from Aerial and Space Images* (cf. [22]). 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 occurr in the data set. To resolve the footprint of the buildings, higher image resolution is necessary. Thus building heights can not be determined automatically and therefore are set 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 incorrectely classified. We are not able to completely reconstruct every corner, because not all corners are observable in the symbolic image descriptions. An application indepent grouping as presented in [20] may improve the initial symbolic image descriptions. Possibly still remaining unreconstructed
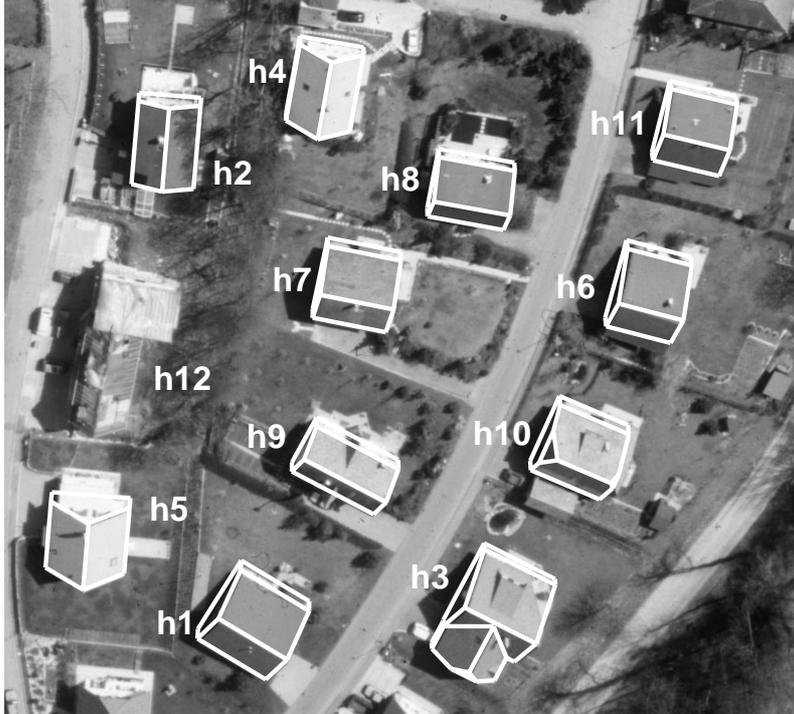
Figure 15: Visualization of the reconstruction result for the Ascona dataset after one iteration loop (cf. Fig. 4) is performed. The images contain 12 buildings in multiple overlap. As building `h12` is under construction, we excluded it from our analysis. We are able to completely reconstruct all of the 11 buildings which were relevant for the analysis.

corners have to 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 $\sigma_x = \sigma_y = \pm 6cm$ and $\sigma_z = \pm 15cm$. The accuracy of the orientation of the corners is about $\sigma_\kappa = 0.6^o$ while the reconstruction of the slope of the roof is by $\sigma_{sl} = 2^o$.

**Generation of building hypotheses (II):** The column `BPH` gives the number of building part hypotheses which 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 which 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 decision which is the best one is given by using the Minimum Description Length criterium following [52].

**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

| | 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 |

| | | | | |
|---|---|---|---|---|
| C | number of corners of the building | BH | number of building hypotheses | |
| RC | number of reconstructed corners | UV | predicted corners before verification | |
| CE | number of corner classification errors | GV | predicted and verified vertices | |
| BPH | number of building part hypotheses | | per image | |
| | for each corner | FP | remaining free parameters | |

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 Dataset. All of the 11 buildings can be reconstructed as shown in column FP.

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 model 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 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 on test data sets provided by the Landesvermessungsamt Bonn and the ETH Zürich.
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, esp. 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 have to consider the integration of logical and statistical knowledge into the framework of constraint logic programming. Furthermore the approach has to 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.

# References

[1] R. Bergevin and M.D. Levine. Generic Object Recognition: Building and Matching Coarse Descriptions from Line Drawings. *IEEE T-PAMI*, 15(1):19–36, 1993.

[2] I. Biederman. Recognition-by-Components: A Theory of Human Image Understanding. *Psychological Review*, 94:115–147, 1987.

[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*, number 1064 in Lecture Notes in Computer Science, pages 85–96. Springer Verlag, Cambridge, 1996.

[4] C. Braun. *Interpretation von Einzelbildern zur Gebäudeerfassung*. PhD thesis, Institut für Photogrammetrie, Universität Bonn, 1994.

[5] C. Braun, T. H. Kolbe, F. Lang, W. Schickler, V. Steinhage, A. B. Cremers, W. Förstner, and L. Plümer. Models for Photogrammetric Building Reconstruction. *Computer & Graphics*, 19(1):109–118, 1995.

[6] A. Brunn, F. Lang, and W. Förstner. A Procedure for Segmenting Surfaces by Symbolic and Iconic Image Fusion. In Jähne B., editor, *Mustererkennung 1996, 18. DAGM-Symposium, Heidelberg*, Informatik aktuell, pages 11–20. Springer Verlag, Berlin, 1996.

[7] M. Clowes. On Seeing Things. *Artificial Intelligence*, 2:79–116, 1971.

[8] R. Dechter and J. Pearl. Network-Based Heuristics for Constraint Satisfaction Problems. *Artificial Intelligence*, 34:370–425, 1988.

[9] S. J. Dickinson, A. P. Pentland, and A. Rosenfeld. 3-D Shape Recovery Using Distributed Aspect Matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):174–198, 1992.

[10] 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):130–154, 1992.

[11] R. Englert. Systematic Acquisition of Generic 3D Building Model Knowledge. In W. Förstner and L. Plümer, editors, *Semantic Modeling for the Acquisition of Topographic Information from Images and Maps*, pages 181–195. Birkhäuser Verlag, Basel, Switzerland, 1997.

[12] A. Fischer and V. Steinhage. On the Computation of Visual Events in Aspect Graph Generation. In *Mustererkennung 1997, 19. DAGM-Symposium, Braunschweig*, Informatik aktuell, pages 156–163. Springer Verlag, Berlin, 1997.

[13] A. Fischer, T. H. Kolbe, and F. Lang. Integration of 2D and 3D Reasoning for Building Reconstruction Using a Generic Hierarchical Model. In W. Förstner and L. Plümer, editors, *Semantic Modeling for the Acquisition of Topographic Information from Images and Maps*, pages 159–180. Birkhäuser Verlag, Basel, Switzerland, 1997.

[14] P. J. Flynn and A. K. Jain. CAD-Based Computer Vision: From CAD-Models to Relational Graphs. *IEEE T-PAMI*, 13(2):114–132, 1991.

[15] W. Förstner. A Framework for Low Level Feature Extraction. In J.-O. Eklundh, editor, *Computer Vision, ECCV '94, Vol. II*, number 801 in Lecture Notes in Computer Science, pages 383–394. Springer Verlag, Berlin, 1994.

[16] W. Förstner. Mid-Level Vision Processes for Automatic Building Extraction. In A. Grün, O. Kübler, and P. Agouris, editors, *International Workshop on Automatic Extraction of Man-Made Objects from Aerial and Space images*, pages 179–188. Birkhäuser, Basel, 1995.

[17] W. Förstner and L. Plümer, editors. *Semantic Modeling for the Acquisition of Topographic Information from Images and Maps*. Birkhäuser Verlag, Basel, 1997.

[18] E. C. Freuder and R. J. Wallace. Partial Constraint Satisfaction. In M. Jampel, E. Freuder, and M. Maher, editors, *Over-Constrained Systems*, number 1106 in Lecture Notes in Computer Science, pages 63–110. Springer Verlag, Berlin, 1996.

[19] P. Fua and A. J. Hanson. Resegmentation Using Generic Shape: Locating General Cultural Objects. *Pattern Recognition Letters*, 5:243–252, 1987.

[20] C. Fuchs and W. Förstner. Polymorphic Grouping for Image Segmentation. In *Proceedings 5th International Conference on Computer Vision*, pages 175–182. IEEE Computer Society Press, 1995.

[21] A. Grün, , E. P. Baltsavias, and O. Henricsson, editors. *Automatic Extraction of Man-Made Objects from Aerial and Space Images (II)*. Birkhäuser, Basel, 1997.

[22] A. Grün, O. Kübler, and P. Agouris, editors. *Automatic Extraction of Man-Made Objects from Aerial and Space Images*. Birkhäuser, Basel, 1995.

[23] E. Gülch. A Knowledge-based Approach to Reconstruct Buildings in Digital Aerial Imagery. In L. W. Fritz and J. R. Lucas, editors, *Proceedings of the 17th ISPRS Congress, Washington D. C.*, volume 29, pages 410–417, 1992.

[24] C. Hansen and T. C. Henderson. CAD-Based Computer Vision. *IEEE T-PAMI*, 11:1187–1193, 1993.

[25] R. M. Haralick and G. L. Elliott. Increasing Tree Search Efficiency for Constraint Satisfaction Problems. *Artificial Intelligence*, 14:263–313, 1980.

[26] R. M. Haralick and L. G. Shapiro. *Computer and Robot Vision*, volume II. Addison-Wesley Publishing Company, New York, 1993.

[27] M. Hendrickx, J. Vandekerckhove, D. Frére, T. Moons, and L. Van Gool. On the 3D Reconstruction of House Roofs from Aerial Images of Urban Areas. In *IAPRS*, volume 32, Part 3-4W2, pages 87–96. ISPRS, 1997.

[28] O. Henricsson and E. P. Baltsavias. 3-D Building Reconstruction with ARUBA: A Qualitative and Quantitative Evaluation. In A. Grün, E. P. Baltsavias, and O. Henricsson, editors, *Automatic Extraction of Man-Made Objects from Aerial and Space Images (II)*, pages 139–148. Birkhäuser, Basel, 1997.

[29] M. Herman and T. Kanade. The 3D MOSAIC Scene Understanding System: Incremental Reconstruction of 3D Scenes from Complex Images. In M. A. Fischler and O. Firschein, editors, *Readings in Computer Vision: Issues, Problems, Principles, and Paradigms*, pages 471–482. Morgan Kaufmann Publishers, Los Altos, California, 1987.

[30] 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*, number 800 in Lecture Notes in Computer Science, pages 411–419. Springer Verlag, Berlin, 1994.

[31] A. Huertas, M. Bejanin, and R. Nevatia. Model Registration and Validation. In A. Grün, O. Kübler, and P. Agouris, editors, *Automatic Extraction of Man-Made Objects from Aerial and Space Images*, pages 33–43. Birkhäuser, Basel, 1995.

[32] D. A. Huffman. Impossible Objects as Nonsense Sentences. *Machine Intelligence*, 6:295–323, 1971.

[33] K. Ikeuchi and P. J. Flynn. Recent Progress in CAD-Based Vision. *CVGIP: Image Understanding*, 61(3):293–294, 1995.

[34] J. Jaffar and M. J. Maher. Constraint Logic Programming: A Survey. *Journal of Logic Programming*, 19/20:503–581, 1994.

[35] M. Jampel, E. Freuder, and M. Maher, editors. *Over-Constrained Systems*. Number 1106 in Lecture Notes in Computer Science. Springer Verlag, Berlin, 1996.

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

[37] 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*, pages 143–154. The Practical Application Company Ltd., London, 1996.

[38] T. Läbe and K.-H. Ellenbeck. 3d-wireframe models as ground control points for the automatic exterior orientation. In *Proceedings ISPRS Congress, Comm. V, Vienna*, 1996.

[39] F. Lang and W. Förstner. 3d-city modelling with a digital one-eye-stereo system. In *ISPRS Symposium, Comm. IV, Vienna*, 1996.

[40] F. Lang and W. Förstner. Surface Reconstruction of Man-Made Objects using Polymorphic Mid-Level Features and Generic Scene Knowledge. *Zeitschrift für Photogrammetrie und Fernerkundung*, 6:193–201, 1996.

[41] C. Lin, A. Huertas, and R. Nevatia. Detection of Buildings Using Perceptual Grouping and Shadows. In *Proceedings of CVPR'94*, pages 62–69, 1994.

[42] C. Lin, A. Huertas, and R. Nevatia. Detection of Buildings from Monocular Images. In A. Grün, O. Kübler, and P. Agouris, editors, *Automatic Extraction of Man-Made Objects from Aerial and Space Images*, pages 125–134. Birkhäuser, Basel, 1995.

[43] 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*, pages 694–699, 1994.

[44] A. K. Mackworth. Consistency in Networks of Relations. *Artificial Intelligence*, 8:99–118, 1977.

[45] D. M. McKeown. Towards Automatic Cartographic Feature Extraction from Aerial Imagery. In H. Ebner, D. Fritsch, and Ch. Heipcke, editors, *Digital Photogrammetric Systems*. Wichmann, Karlsruhe, 1990.

[46] P. Meseguer. Constraint Satisfaction Problems: An Overview. *AICOM*, 2(1):3–17, 1989.

[47] 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*, pages 364–371, 1993.

[48] O. Munkelt. Aspect-Trees: Generation and Interpretation. *CVGIP*, 61(3):365–386, 1995.

[49] R. Nevatia, C. Lin, and A. Huertas. A System for Building Detection from Aerial Images. In A. Grün, E. P. Baltsavias, and O. Henricsson, editors, *Automatic Extraction of Man-Made Objects from Aerial and Space Images (II)*, pages 77–86. Birkhäuser, Basel, 1997.

[50] M. Pilu and R. B. Fisher. Snakes and Splines for Tracking Non-Rigid Heart Motion. In *Computer Vision — ECCV '96*, number 1064 in Lecture Notes in Computer Science, pages 71–82. Springer Verlag, Cambridge, 1996.

[51] L. Quam and T. M. Strat. SRI Image Understanding Research in Cartographic Feature Extraction. In H. Ebner, D. Fritsch, and C. Heipke, editors, *Digital Photogrammetric Systems*, pages 111–121. Wichmann, Karlsruhe, 1991.

[52] J. Rissanen. Minimum-Description-Length Principle. *Encyclopedia of Statistical Sciences*, 5:523–527, 1987.

[53] M. Roux and H. Maitre. 3-D Description of Urban Areas Using Maps and Aerial Images. In A. Grün, E. P. Baltsavias, and O. Henricsson, editors, *Automatic Extraction of Man-Made Objects from Aerial and Space Images (II)*, pages 311–322. Birkhäuser, Basel, 1997.

[54] W. Schickler. Towards Automation in Photogrammetry. *Geodetical Information Magazine*, 7(4):32–35, 1993.

[55] M. Sester and W. Förstner. Object Location Based on Uncertain Models. In H. Burkhardt, K.-H. Höhne, and B. Neumann, editors, *Mustererkennung 1989, 11. DAGM-Symposium, Hamburg*, volume 219 of *Informatik Fachberichte*, pages 457–464. Springer Verlag, Berlin, 1989.

[56] L. G. Shapiro and R. M. Haralick. A Metric for Comparing Relational Descriptions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 7(1):90–94, 1985.

[57] V. Steinhage. On the Integration of Object Modeling and Image Modeling in Automated Building Extraction from Aerial Images. In A. Grün, E. P. Baltsavias, and O. Henricsson, editors, *Automatic Extraction of Man-Made Objects from Aerial and Space Images (II)*, pages 139–148. Birkhäuser, Basel, 1997.

[58] P. Suetens, P. Fua, and A. J. Hanson. Computational Strategies for Object Recognition. *ACM Computing Surveys*, 24(1):5–61, 1992.

[59] K. Sugihara. *Machine Interpretation of Line Drawings*. MIT-Press, 1986.

[60] P. van Hentenryck. *Constraint Satisfaction in Logic Programming*. Logic Programming Series. MIT Press, Cambridge, MA, 1989.

[61] G. Vosselman. *Relational Matching*. Number 628 in Lecture Notes in Computer Science. Springer Verlag, Berlin, 1992.

[62] R. J. Wallace and E. C. Freuder. Heuristic Methods for Over-Constrained Constraint Satisfaction Problems. In M. Jampel, E. Freuder, and M. Maher, editors, *Over-Constrained Systems*, number 1106 in Lecture Notes in Computer Science, pages 207–216. Springer Verlag, Berlin, 1996.

[63] D. L. Waltz. Understanding Line Drawings of Scenes with Shadows. In P. H. Winston, editor, *Psychology of Computer Vision*, pages 19–91. McGraw-Hill, New York, 1975.

[64] U. Weidner. An approach to building extraction from digital surface models. *ISPRS*, 1996.

[65] U. Weidner and W. Förstner. Towards Automatic Building Extraction from High Resolution Digital Elevation Models. *ISPRS Journal of Photogrammetry and Remote Sensing*, 50(4):38–49, 1995.