Towards Automatic Large-Scale 3D Building Reconstruction: Primitive Decomposition and Assembly

Hai Huang and Helmut Mayer

Abstract In this paper we propose a pipeline for highly automatic building reconstruction based on 3D point clouds. 3D building models are of great interest for many applications including city planning, navigation, emergency response and tourism and their reconstruction has been intensively studied. It is, however, still a challenge to minimize manual intervention and to achieve highly automated processing in practical applications. The main reason lies in the variability and complexity of urban scenes. We believe that one possible key to tackle this is a reliable primitive-based decomposition of urban scenes as well as their constituent buildings. It links scene interpretation with model reconstruction and, thus, naturally completes an automatic reconstruction pipeline. We propose an effective scheme for the decomposition of the whole scene straight into individual building components, i.e., primitives. The primitives are reconstructed via statistical generative modeling and assembled into individual watertight building models. An experiment has been performed on a dataset of a complete central European village demonstrating the potential of the proposed approach.

Keywords 3D reconstruction · Point cloud · Building · Statistical modeling

1 Introduction

The automatic generation of 3D building models from remote sensing data is of great interest for many applications including city planning, navigation, emergency response and tourism. Many approaches have been reported in the past decades. Overviews are given in Brenner (2005), Schnabel et al. (2008), Vosselman (2009). Current work includes (Sampath and Shan 2010), which segments and reconstructs

H. Huang (✉) · H. Mayer
Institute for Applied Computer Science, Bundeswehr University Munich,
Werner-Heisenberg-Weg 39, 85577 Neubiberg, Germany
e-mail: hai.huang@unibw.de

H. Mayer
e-mail: helmut.mayer@unibw.de

complicated buildings from airborne LIDAR point clouds based on polyhedral models. Starting from planar roof segments, Zhou and Neumann (2012) try to organize them using "global regularities" in the form of orientation and placement constraints. Lafarge et al. (2010) present building reconstruction from a Digital Surface Model (DSM) combining generic and parametric methods. For more sophisticated buildings, basic geometric primitives, e.g., planes, cylinders and cones, are combined with mesh-patches to present irregular roof forms (Lafarge and Mallet 2012). Huang et al. (2013a) propose a statistical approach for building model reconstruction from LIDAR data via generative models. Partovi et al. (2015) present an extension of a hybrid framework for data from stereo satellite imagery with ridge-line-based building mask decomposition.

Approaches for building footprint decomposition can be divided into two categories: With or without primitive overlap. Brenner and Haala (2000) propose a flexible decomposition scheme resulting in overlapping rectangular primitives. Lafarge et al. (2010) present a decomposition of given building footprints as adjacent primitives, which are not limited to rectangular shapes. Current work includes (Wang et al. 2015), which presents building decomposition based on the adjacency graph of detected planar roof patches and a primitive-based reconstruction.

In recent years, quality and availability of 3D point clouds from LIDAR and image matching have been significantly improved and some approaches reach a high level of automation (Lafarge and Mallet 2012; Huang et al. 2013a) and cover large suburban areas. There are, however, still several challenges towards fully automatic city model reconstruction. One of them is the parsing of complex scenes as well as buildings. This limits the reconstruction of larger urban areas and, thus, renders a pipeline to build a whole city model unreliable. We believe, that the key to tackle this deficit is a reasonable decomposition subdividing the whole scene as well as heterogeneous buildings into regular components.

As shown in Fig. 1, this paper extends and links our approaches to scene classification (Huang and Mayer 2015) and primitive-based reconstruction (Huang et al. 2011, 2013a) with a pre-processing—decomposition and a post-processing—assembly of primitives to complete a practical reconstruction pipeline. A reasonable and reliable decomposition of the whole scene as well as of complicated buildings into regular primitives precedes model reconstruction. The assembly of the primitives performs a true model merging in CAD (Computer Aided Design) style. The decomposition works as an intermediate step linking the preceding scene interpretation with the following model reconstruction and is, therefore, a crucial part to complete an automatic reconstruction pipeline. The building mask from scene classification is used as input along with the 3D point cloud derived from dense image matching. Scene decomposition splits individual buildings from the building mask while building decomposition further disassembles building complexes into simple building components. The latter are represented by predefined 3D primitives and reconstructed via statistical generative modeling. The primitives are subsequently assembled into individual watertight building models via CSG (Constructive Solid Geometry) modeling.
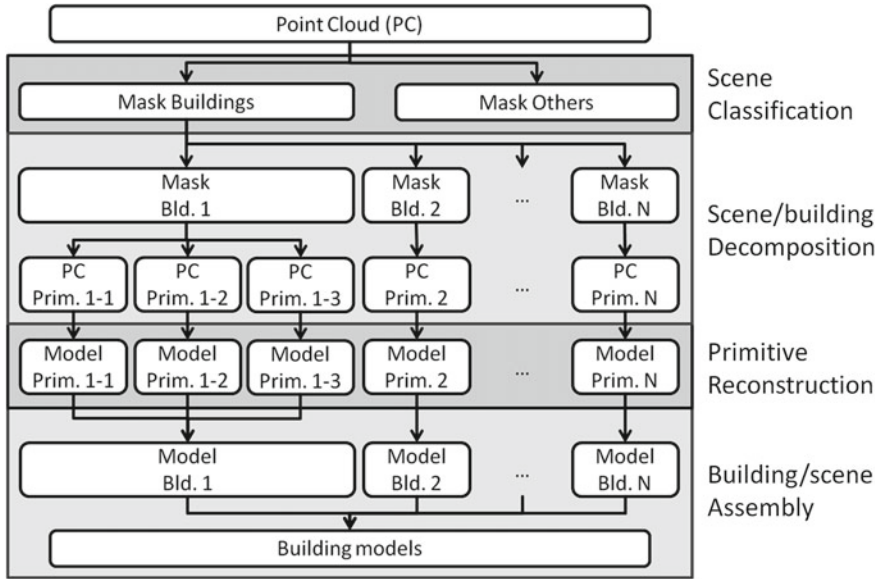
Fig. 1   Pipeline for automatic building reconstruction

This paper is organized as follows: Sects. 2 and 3 present scene and building decomposition, respectively. The statistical generative modeling of building primitives is given in Sect. 4. Section 5 describes the assembly of primitives into complete building models. Experimental results are given in Sect. 6. The paper ends up with the conclusion in Sect. 7.

2   Scene Decomposition

The goal of scene decomposition is to extract individual buildings from a (binary) building mask of the whole scene. The building mask (Fig. 2, left) is derived by previous scene classification, which may contain labeling errors often affecting the separation of buildings (middle). A mathematical morphological "opening" operation is conducted, as shown in Fig. 2 (right), to remove trivial areas and to better isolate the buildings. The radius of the disk-shaped structuring element is determined based on data quality as well as resolution. For the presented dataset with 0.2 m resolution, a radius of 1 m is employed for the structuring element.

Individual buildings are found via "blob" detection. As shown in Fig. 3, the input data are then correspondingly segmented into rectangular tiles. The segmentation is conducted with a buffer, so that certain classification errors can be tolerated. Overlapping between tiles is allowed to make sure buildings are completely included in
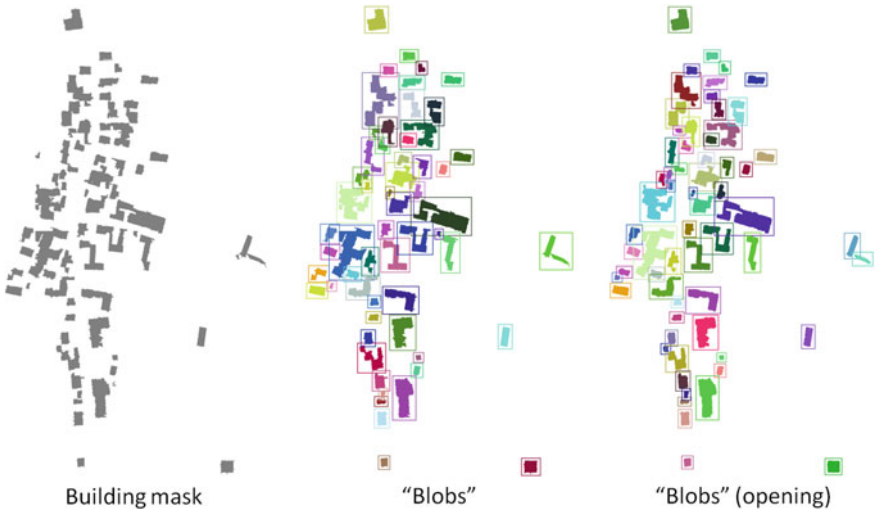
Fig. 2 Scene decomposition and individual building detection: comparison of building detection (as colorful "blobs") without (*middle*) and with (*right*) morphological opening on the input building mask (*left*)
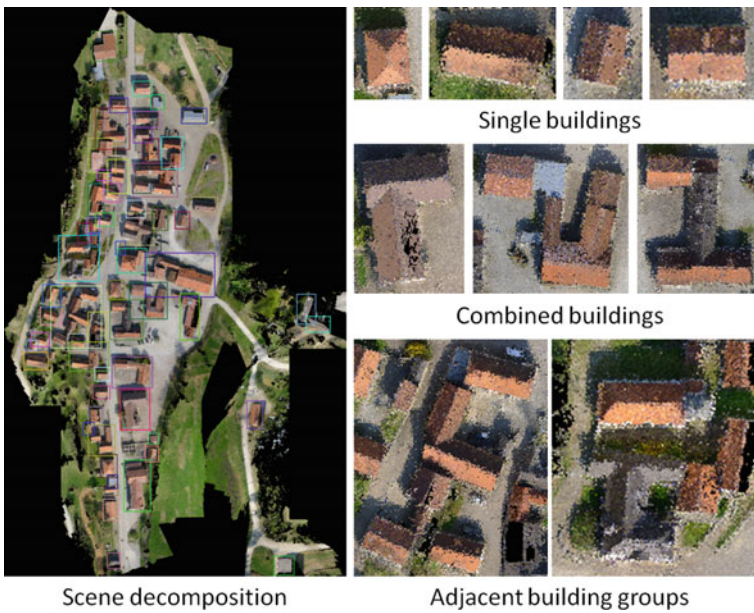

Fig. 3 Scene decomposition into tiles, which may contain individual or multiple buildings

the tiles. Global coordinates including the height (for undulating areas, cf. Sect. 6) are attributed to each tile for the final model assembly of the whole scene.

It, however, cannot be guaranteed that each "blob" contains exactly one individual building. This is due to the complexity of the scene and the imperfection of the classification i.e., after the decomposition, a data tile may contain a single building, but also a combined building (a building consisting of multiple building components), or multiple buildings which are closely adjacent to each other (cf. Fig. 3). The last case can often be found in densely inhabited areas. In this paper we call both, combined buildings and adjacent building groups, "building complexes". Further parsing of building complexes is described in the following section.

3   Building Decomposition

Generally, the goal of building decomposition is to divide building complexes into simple standard building components making the following model reconstruction easier and more efficient. This is especially true for primitive-based reconstruction methods, where the building components are directly represented by predefined building primitives.

Please note that building decomposition actually does not work on building models but directly on the input data, because the former does not exist yet. Conventionally, the decomposition is conducted bottom-up based on 2D footprints that are either already available (Lafarge et al. 2010; Kada and McKinley 2009) or extracted from the data (Partovi et al. 2015). This becomes, as shown in Fig. 4c, infeasible when 3D information has to be taken into account. Adjacent building components with different 3D geometry cannot be separated if they have similar width. This error will affect the following model reconstruction.

We propose a combined bottom-up and top-down scheme for the decomposition of building complexes. It works based on 3D geometry parsing with the support of
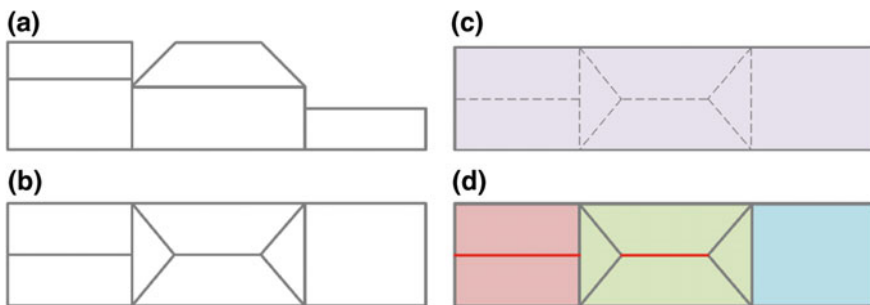

Fig. 4 Footprint- and ridge-based building decomposition: Side-view (**a**) and top-view (**b**) of a building complex and the decomposition based on footprint (**c**) and 3D geometry (**d**) using ridge (*red*) parsing and height differences

a predefined primitive library. As shown in Fig. 4d, the decomposition is improved by using 3D information including height differences and roof shapes.

We are aware that the strategies of the decomposition and the following model reconstruction have to match i.e., they should share the same construction concept for buildings and be adapted to each other, so that the pipeline works smoothly and the degree of automation is improved. There are two basic strategies for building (footprint) decomposition. The key difference is if the final building components are allowed to overlap (Brenner and Haala 2000; Huang et al. 2011) or not (Lafarge et al. 2010; Kada and McKinley 2009). Different primitive definitions are correspondingly employed. The concept allowing for overlaps fits better to the generative modeling process described in Huang et al (2013a). It is more flexible and has the potential to keep the model complete and regular with a reasonable size for the primitive library. No special primitives such as additional joint parts (Lafarge et al. 2010; Kada and McKinley 2009) are required. Besides, allowing primitive overlaps can tolerate a certain extent of uncertainty as result of the stochastic search during model reconstruction (cf. Sect. 4).

3.1   Ridge Lines

The ridge line is one of the key geometric features of many buildings. In approaches for roof interpretation ridge lines often play an important role, especially for those that use adjacency graph models (Elberink and Vosselman 2009; Huang and Brenner 2011). It has been demonstrated that the ridge line is not only the most significant, but also the most stable feature for 3D building roof model reconstruction.

Ridge lines are also used to improve the bottom-up decomposition performance (Arefi et al. 2010; Partovi 2015). The ridges are mostly treated as a building's central line and extracted from 2D contours or simple height information (line fitting to the points with a given height limit for ridges). Alternatively, "skeletons" derived from 2D footprints are often employed for building structure analysis (Haunert and Sester 2008, 2013b). Decomposition, however, becomes more difficult when the complexity of buildings increases.

We propose a top-down primitive-based decomposition scheme with emphasis on complicated building structures. The ridge lines are extracted and divided into straight line segments, which can be seen as ridges of individual primitives and, thus, guide the decomposition. In comparison with related work, the ridge lines are much more precisely extracted fully 3D by plane detection and intersection instead of an approximation by means of fitting lines to candidate ridge points (Arefi et al. 2010) or a morphological operation (Partovi et al. 2015). With the focus on the degree of automation, we concentrate on buildings with regularly structured components, which make up the majority of urban areas, instead of atypical or landmark buildings. The latter are rare and manual intervention in the reconstruction is mostly unavoidable anyway.
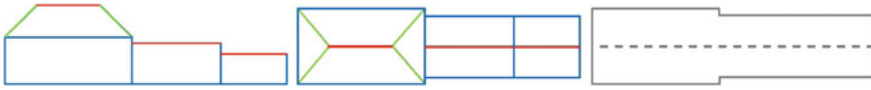
Fig. 5 Ridges and building skeleton—*Horizontal* ridge (*red*), diagonal ridges (*green*), eaves (*blue*), and skeletons (*gray dashed*) derived from 2D footprint (*gray*)

We define, as shown in Fig. 5, the following types of edges on the roof: (1) Horizontal ridge line (red), which connects two apexes of the roof, (2) diagonal ridge line (green) connecting one apex and one eave corner, and (3) eave line (blue), which links two eave corners. The roof contour consisting of eave lines can be used to approximate the building footprint when the overhang of the eave is ignored.

The ridge lines proposed in this paper have the following advantages as basis of building decomposition:

1. No additional footprint data is required. Ridge lines are derived directly from the input point cloud by means of plane intersection, which is much more accurate than conventional approximation methods.
2. With full 3D parsing more specific and accurate (e.g., asymmetric roofs, different roof heights and types) geometrical information is available.

For flat roofs, which do not have ridges, central lines are used instead. In comparison with building skeletons, central lines of roofs are still 3D i.e., as shown in Fig. 5, they have a height value and in the case of adjacent flat roofs they can differentiate multiple building components by means of their heights.

For non-flat roofs, ridge lines are found in the form of the intersection lines of the individual planes of the roofs. Planes of a building complex are detected by RANSAC. By this means, all intersection lines are actually determined via a consensus of all data points of both planes i.e., the intersection lines are much more reliable and precise.

To separate ridges from other kind of intersection lines, we employ the "relation matrix" (Huang and Brenner 2011) shown in Fig. 6. After the planes have been detected, the points that lie in the intersection area are counted and the numbers are entered in the relation matrix. The intersection area is defined based on the intersection line along with a buffer range, which is empirically determined proportionally to the point resolution.

"False" intersection lines are often found (cf. Fig. 6, dashed gray lines, planes 2–5 and 3–4), because the planes are actually infinite with no boundaries defined. An intersection line is verified by checking the normal directions of the planes on its both sides. Similar normal directions imply that the intersection line lies inside one slope of the roof and is,therefore, "false". The corresponding cell in the relation matrix is then set to null. The relation matrix is symmetric. For the purpose of illustration we, as shown in Fig. 6 (right), use the lower triangle (gray) to show the original numbers of intersection points while the upper triangle (colored) presents the verification of horizontal (red) and diagonal (green) ridges.
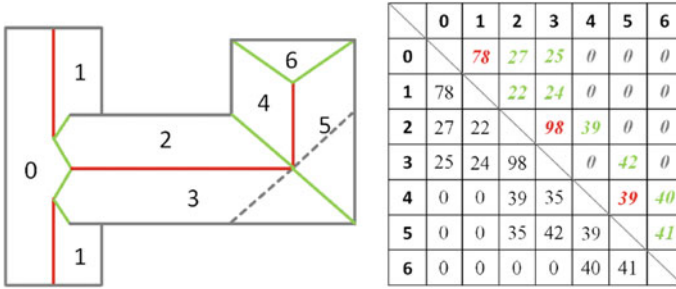
	0	1	2	3	4	5	6
0		78	27	25	0	0	0
1	78		22	24	0	0	0
2	27	22		98	39	0	0
3	25	24	98		0	42	0
4	0	0	39	35		39	40
5	0	0	35	42	39		41
6	0	0	0	0	40	41	

Fig. 6 Ridge line determination with relation matrix: *Horizontal* ridges (*red*) and diagonal ridges (*green*)

3.2 Primitive-Based Decomposition

A top-down building decomposition is proposed based on a predefined primitive library given in Huang et al. (2013b). Figure 7 presents the process of building decomposition guided by the ridges. The detected horizontal ridge lines (Fig. 7b, red) are decomposed into straight line segments (Fig. 7c, bold). The primitives have a rectangular contour as well as a single straight horizontal ridge line (except flat
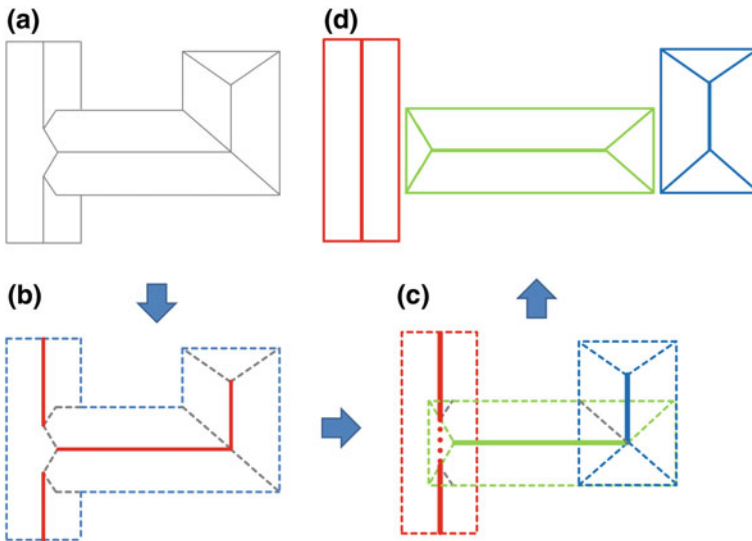

Fig. 7 Building decomposition: **a** Underlying model, **b** detected ridge lines (*red*), **c** decomposition based on ridge segments (*solid lines*) with completion using edges of the primitives (*dashed lines*), and **d** the decomposed primitives

roof and shed roof). The end points of the segments are determined by intersection with diagonal ridges or the boundary of the building mask.

Using the horizontal ridges as bases (cf. Fig. 7c), the most appropriate primitives are statistically selected from the library. The goal is a fit (1) to the already extracted diagonal ridges and (2) to the rest of the edges, without conflicts with the known planes and the boundary of the building mask. Again, this step decomposes no actual building model but underlying models, because the former does not yet exist. The goal is to define primitives that compose the building instead of modeling them i.e., the decomposition determines the number and types of primitives and the way of their combination (Fig. 7d). The concrete parameters of the primitives are calculated in the following primitive reconstruction (Sect. 4).

The primitives are, however, not the only information that can be derived from the building decomposition. The ridge line is a key component and plays an important role for the roof geometry. Initial values of the following primitive parameters can be derived solely from known horizontal ridges:

- Centroid coordinates (x and y) are defined by the center of the ridge line.
- Orientation (*azimuth*) is that of the ridge line.
- Ridge height (z_2) of the roof.
- Length is approximated proportionally to that of the ridge line.

Furthermore, the following initial values are obtained in combination with the known building mask. In the case that the building mask is not available, e.g., for building decomposition without previous scene decomposition, the diagonal ridges can be used instead:

- Area is approximately proportional to the mask area or the number of data points (for raster data).
- Width is derived from the area and length.
- Depth of hips (hip_{l1}, hip_{l2}, hip_{d2}, and hip_{d2}) are the longitudinal and radial distances from the end points of the horizontal ridge to the boundary of the building mask.

Since the ridge lines are precisely determined (cf. Sect. 3.1), the derived initial values are, to a certain extent, reliable and specific. They can, therefore, significantly improve the performance of the statistical search (cf. Sect. 4).

4 Primitive Reconstruction

We propose a generative primitive-based reconstruction, which extends the approach described in Huang et al. (2013a). The primitive parameters θ are defined as:

$$\theta \in \Theta; \Theta = \{\mathcal{P}, \mathcal{C}, \mathcal{S}\}, \tag{1}$$

where the parameter space Θ consists of position parameters $\mathcal{P} = \{x, y, azimuth\}$, contour parameters $C = \{length, width\}$ (rectangle footprint), and S containing shape parameters, i.e., ridge/eave height and the depths of hips.

The goal of statistical reconstruction is to optimize the parameters fitting the primitive to the data. The Maximum A Posteriori (MAP) estimate of Θ can be expressed as

$$\hat{\Theta}_{MAP} = \underset{\Theta}{argmax}\left\{ \frac{L(\mathcal{D}|\Theta)p(\Theta)}{P(\mathcal{D})} \right\} = \underset{\Theta}{argmax}\left\{ L(\mathcal{D}|\Theta)p(\Theta) \right\}, \qquad (2)$$

where $L(\mathcal{D}|\Theta)$ is the likelihood function presenting the goodness of fit of the model to the data $\mathcal{D}$ and $p(\Theta)$ presents the prior for Θ, which is derived from empirical knowledge and incrementally improved during the reconstruction i.e., the parameter values of the already found building components or of adjacent buildings are used to update the priors. $P(\mathcal{D})$ is the marginal probability, which is regarded as a constant in the optimization as it does not depend on Θ.

The statistical optimization of the parameters is driven by reversible jump Markov Chain Monte Carlo with model selection in the transition kernel. Multiple hypothetic models ("candidates") are generated via statistical sampling of the primitive type as well as the corresponding parameters and evaluated based on the given 3D point cloud. The final model is the verified candidate model with the best goodness of fit to the data. Markov Chain Monte Carlo is employed for an efficient exploration of the high-dimensional (determined by the number of parameters) search space and the reversible jump mechanism is used for switching between different search spaces, i.e., different types of primitives. By these means, the statistical optimization including the change of primitive types is fully automatic. In comparison to Huang et al. (2013a), the search spaces are of the same size, but due to the more reliable initial values (cf. Sect. 3) the search entropy is much lower i.e., the computational effort is significantly reduced (cf. Sect. 6) by the proposed building decomposition before the reconstruction.

3D point clouds from image matching may contain data flaws such as false color and false positions of points. One important issue are gaps in the data, which often occur on surfaces with homogeneous color/texture (i.e., no matching points) and in case of occlusions. It leads to (cf. Fig. 8, left) missing points—holes on the roofs. Please note that such data gaps also exist in LIDAR data in areas where reflections, e.g., on glass surfaces and water bodies, occur. Conventional bottom-up methods may encounter difficulties in this case resulting in irregular and/or incomplete building components. Complete and watertight building models can, therefore, not be guaranteed. As shown in Fig. 8, the proposed method is robust despite such data flaws and ensures plausible results.
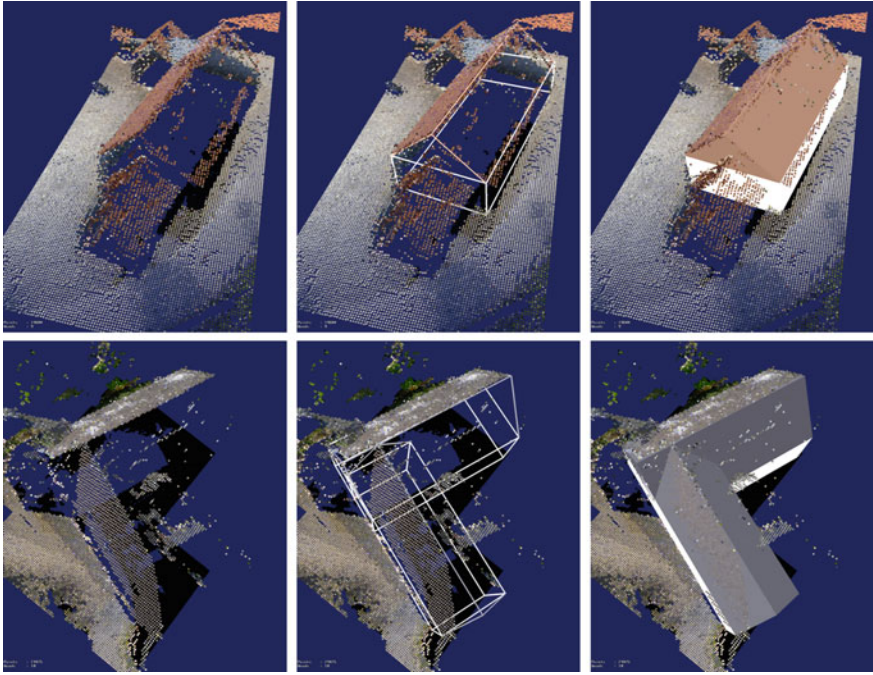
Fig. 8 Robust reconstruction despite data flaws: the input point clouds (*left*), detected primitives shown as wireframes (*middle*), and final building models (*right*)

5   Assembly of Buildings

The reconstructed primitives of a building complex are assembled into a single model. The modeling methods used for buildings can be categorized in two concepts (Brenner 2004): Surface modeling, also known as boundary representation (B-Rep) and solid body modeling, which is mainly based on CSG (Mäntylä 1987). In B-Rep modeling buildings are described by their bounding surfaces and their relationship of intersection. CSG is widely employed in CAD tools. Complicated models are represented by multiple volumetric primitives combined using Boolean operators. Both of them are employed in this paper for building assembly, which consists of two consecutive parts: (1) Joint parametric adjustment and (2) geometrical model merging.

Joint parametric adjustment helps to remove trivial conflicts between primitives and compensates for small deviations (cf. Fig. 9a), which may happen during the reconstruction driven by a stochastic process (cf. Sect. 4). The parameters of all building components are jointly adjusted using two rules. In the joint adjustment the change of each side of a primitive is proportional to its size, i.e., footprint area.
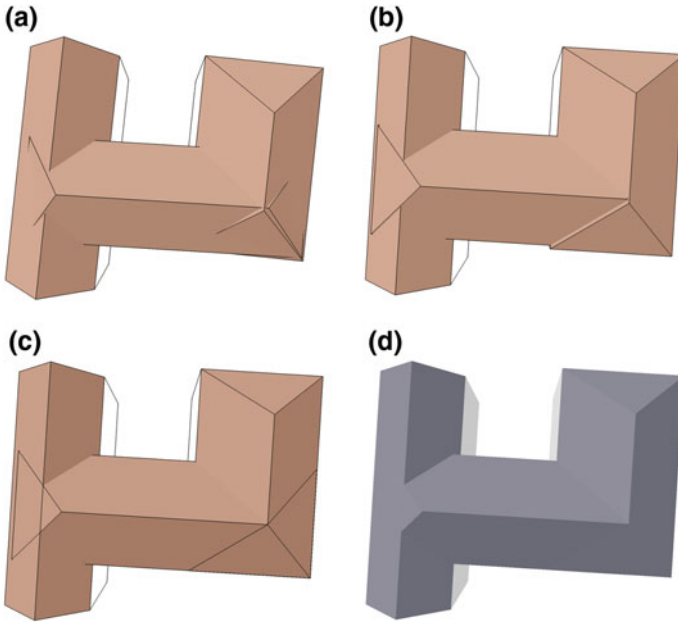
Fig. 9 Primitive adjustment: Reconstructed primitives (**a**), joint parametric adjustment (**b**), vertices-shifting adjustment (**c**), and rendered model (**d**)
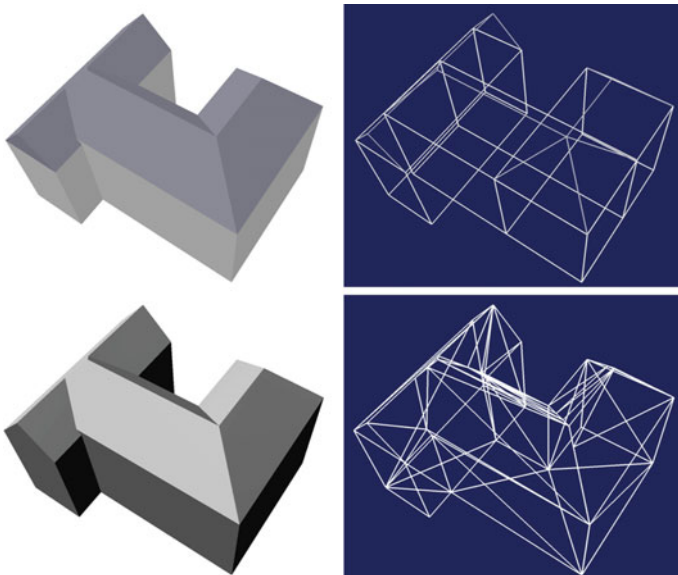

Fig. 10 Primitive merging: from multiple primitives (*top*) to single watertight model (*bottom*)

- Rule 1: The intersection angles of the primitives are jointly regularized to 0° or 90° if the deviation is less than a threshold of 5°. An exception exists for merging a flat roof with a roof with a ridge line (e.g., gable roof): The flat roof is aligned to the latter instead of adjusting both, as the orientation of a ridge is much more reliable than that of a flat roof.
- Rule 2: Heights of flat roofs or ridge- and eave-heights of other roofs are harmonized if the deviation is less than 0.2 m.

Figure 9b shows that the parameter adjustment cannot remove all mismatching positions of the primitives. The mismatching is the result of deviations caused by stochastic processes and data uncertainty, which in principle cannot be corrected in this step. Therefore, further geometrical adjustment is required.

Geometrical model merging generates the final single model of the building complex. Inspired by Huang et al. (2013a), we conduct a simple vertices-shifting (Fig. 9c) to correct the geometrical mismatching and all the primitives are merged into a watertight model. The primitives are originally generated as B-Rep models, as shown in Fig. 10 (top) and simply placed together as two separate models which overlap. Although in the rendered model (left) the intersected part is hidden and does not affect the appearance, the model is ontologically not a single "subject" and geometrically not watertight. Our model merging employs CSG modeling. As shown in Fig. 10, the B-Rep primitives are first converted into CSG models and merged with a "union" operation into a single solid body. The latter is then converted back to a single and watertight B-Rep model, i.e., the final model.

6 Experiments

The experiment is performed for a complete and typical central European village with a mixture of detached buildings and building complexes, a church, and a small castle on a hill. The 3D point cloud has been reconstructed by dense matching of UAS (unmanned aerial system) imagery taken in Bonnland, Germany (Kuhn et al. 2014). The data are generated from 822 images that cover about 0.12 km^2 of undulating terrain. We use a reduced and rasterized version with a resolution of 0.2 m (Fig. 11a, b). The building mask (Fig. 11c, red) is provided by a previous scene classification (Huang and Mayer 2015).

The building mask is decomposed into 62 data tiles (cf. Fig. 2), which are processed in parallel and the reconstructed building models are assembled in the global system. Bird's-eye views of input point cloud (top) and the reconstructed model (bottom) are given in Fig. 12. Along with the watertight building models a mesh model is generated from the non-building points to model the ground.

The total runtime of the presented scene with 33 single buildings and 29 building complexes, which is composed of 112 primitives, is about 14 min on a laptop with a 4 cores/8 threads CPU at 2.3 GHz. The overall reconstruction error (for each individual building) is defined as average deviation from the data points to the model surface
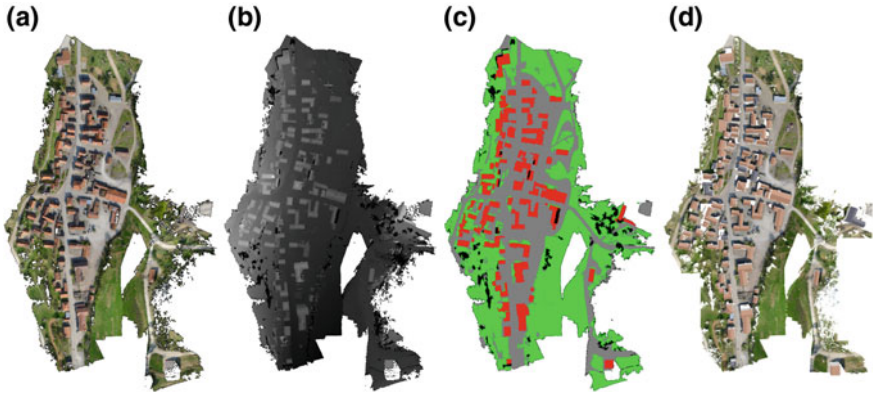
Fig. 11 Experiment on Bonnland data: **a** Input point cloud with color, **b** input point cloud with height presented as *gray* values, **c** the result of scene classification with the building mask in *red*, and **d** the final vector model of the whole scene
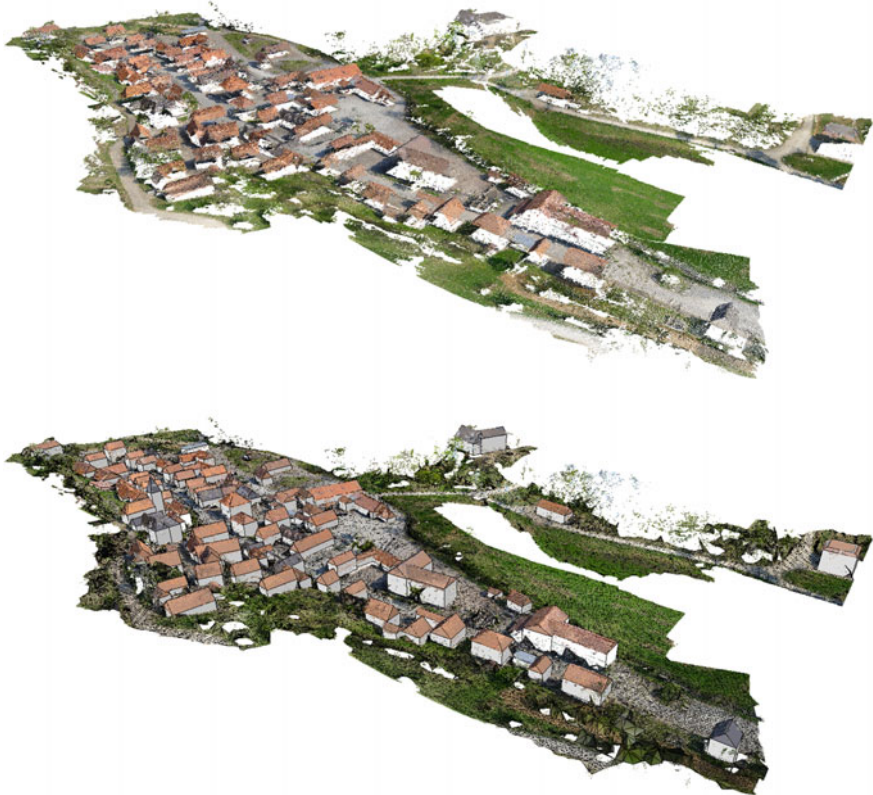

Fig. 12 Building reconstruction for Bonnland: input point cloud (*top*) and reconstructed models (*bottom*)

Table 1 Comparison of the runtimes (in seconds) with and without building decomposition

Methods	Prim. 1	Prim. 2	Prim. 3	Merging	Total	In %
w/o Decomp. (global + local)	104 + 36	89 + 28	65 + 32	25	379	100
with Decomp. (local only)	22	20	25	25	92	24.2

in nadir direction (cf. also Huang et al. 2013a). Except for buildings with large data flaws (cf. Fig. 8) or occlusion, where the average data deviation does not reflect the reconstruction accuracy, the reconstruction errors of the majority of the buildings are less than the half of the data resolution, i.e., 10 cm.

To demonstrate the performance improvement with building decomposition (cf. Sect. 3) in comparison to a direct reconstruction of a building complex we list in Table 1 the detailed runtime analysis for the example model presented in Fig. 10 with three primitives. With the initial values provides by building decomposition computational effort has been saved not only for the time-consuming global searching, but also for the local optimization of other parameters.

7 Conclusion

This paper presents an automatic pipeline for Level of Detail 2—LOD2 building model reconstruction focusing on a reliable scene as well as building decomposition into regular primitives and their subsequent assembly. We proposed:

1. Decomposition of the whole scene into data tiles containing individual buildings or building complexes
2. Decomposition of building complexes into standard primitives with fully 3D geometrical parsing
3. CSG primitive assembly into a single watertight model.

The primitive decomposition links the scene interpretation with the model reconstruction and, thus, completes the automatic modeling pipeline. Additionally, it significantly improves the reconstruction efficiency concerning the following aspects:

1. The time-consuming global search for buildings in a large scene is avoided.
2. The data tiles can be processed independently in parallel.
3. The initial values derived for building decomposition are more precise and reliable than that derived from a simple building mask making the parameter optimization of the primitives much more efficient (cf. Sect. 6).

Concerning future work, we consider to extend the proposed reconstruction pipeline to LOD3 building models. To this end, an approach to object detection on the facades and roofs will be included. CSG modeling should be advantageous for the consistent integration of windows, doors, balconies, chimneys, and dormers into

the 3D model. Furthermore, the library of primitives will be improved with non-rectangular bases, e.g., ellipses or general polygons. The texturing of the buildings is of great interest for certain applications such as tourism. B-Rep models can be much more easily derived from CSG models than the other way around. We assume that a watertight B-Rep model with regular shapes, which we generate with our approach, is a good basis for texturing.

References

Arefi H, Hahn M, Reinartz P (2010) Ridge based decomposition of complex buildings for 3D model generation from high resolution digital surface models. In: The international archives of the photogrammetry, remote sensing and spatial information sciences, vol 38(1/W17)

Brenner C (2004) Modelling 3D objects using weak CSG primitives. In: international archives of photogrammetry and remote sensing, vol 35

Brenner C (2005) Building reconstruction from images and laser scanning. Int J Appl Earth Obs Geoinf, Theme Issue Data Qual Earth Obs Tech 6(3–4):187–198

Brenner C, Haala N (2000) Erfassung von 3D Stadtmodellen. Photogrammetrie - Fernerkundung - Geoinformation 2:109–117

Elberink SO, Vosselman G (2009) Building reconstruction by target based graph matching on incomplete laser data: analysis and limitations. Sensors 9(8):6101. doi:10.3390/s90806101, http://www.mdpi.com/1424-8220/9/8/6101

Haunert JH, Sester M (2008) Area collapse and road centerlines based on straight skeletons. GeoInformatica 12(2):169–191. doi:10.1007/s10707-007-0028-x

Huang H, Brenner C (2011) Rule-based roof plane detection and segmentation from laser point clouds. In: Joint urban remote sensing event (JURSE) 2011, 11–13 April. IEEE, Munich, Germany, pp 293–296. doi:10.1109/JURSE.2011.5764777

Huang H, Brenner C, Sester M (2011) 3D building roof reconstruction from point clouds via generative models. 19th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems (GIS), 1–4 November. ACM Press, Chicago, IL, USA, pp 16–24

Huang H, Mayer H (2015) Robust and efficient urban scene classification using relative features. In: 23rd ACM SIGSPATIAL international conference on advances in geographic information systems. ACM, New York, NY, USA, GIS'15, pp 81:1–81:4. doi:10.1145/2820783.2820872

Huang H, Brenner C, Sester M (2013a) A generative statistical approach to automatic 3D building roof reconstruction from laser scanning data. ISPRS J Photogrammetry Remote Sens 79:29–43. doi:10.1016/j.isprsjprs.2013.02.004, http://www.sciencedirect.com/science/article/pii/S0924271613000476

Huang H, Kieler B, Sester M (2013b) Urban building usage labeling by geometric and context analyses of the footprint data. In: 26th international cartographic conference (ICC)

Kada M, McKinley L (2009) 3D building reconstruction from LiDAR based on a cell decomposition approach. In: The international archives of the photogrammetry, remote sensing and spatial information sciences, vol 38(3/W4), pp 47–52

Kuhn A, Mayer H, Hirschmüller H, Scharstein D (2014) A TV prior for high-quality local multi-view stereo reconstruction. In: 2nd international conference on 3D vision (3DV), pp 65–72

Lafarge F, Mallet C (2012) Creating large-scale city models from 3d-point clouds: a robust approach with hybrid representation. Int J Comput Vision 99(1):69–85

Lafarge F, Descombes X, Zerubia J, Pierrot-Deseilligny M (2010) Structural approach for building reconstruction from a single DSM. IEEE Transactions on Pattern Analysis and Machine Intelligence 32(1):135–147

Mäntylä M (1987) An introduction to solid modeling. Computer Science Press Inc, New York, NY, USA

Partovi T, Huang H, Krauß T, Mayer H, Reinartz P (2015) Statistical Building Roof Reconstruction from WORLDVIEW-2 Stereo Imagery. ISPRS—International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences pp 161–167. doi:10.5194/isprsarchives-XL-3-W2-161-2015

Sampath A, Shan J (2010) Segmentation and reconstruction of polyhedral building roofs from aerial lidar point clouds. IEEE Trans Geosci Remote Sens 48(3):1554–1567. doi:10.1109/TGRS.2009.2030180

Schnabel R, Wessel R, Wahl R, Klein R (2008) Shape recognition in 3D point-clouds. In: The 16th international conference in central europe on computer graphics, visualization and computer vision, 4–7 February, Plzen-Bory, Czech Republic

Vosselman G (2009) Advanced point cloud processing. In: Fritsch D (ed) Photogrammetric Week '09. Heidelberg, Germany, pp 137–146

Wang H, Zhang W, Chen Y, Chen M, Yan K (2015) Semantic decomposition and reconstruction of compound buildings with symmetric roofs from lidar data and aerial imagery. Remote Sens 7(10):13,945. doi:10.3390/rs71013945, http://www.mdpi.com/2072-4292/7/10/13945

Zhou QY, Neumann U (2012) 2.5D building modeling by discovering global regularities. In: The IEEE computer society conference on computer vision and pattern recognition, 16–21 June. IEEE Computer Society, Providence, RI, USA, pp 326–333