Contents lists available at ScienceDirect

ISPRS Journal of Photogrammetry and Remote Sensing

journal homepage: www.elsevier.com/locate/isprsjprs

Knowledge-based building reconstruction from terrestrial video sequences

Yixiang Tian [a,b,*], Markus Gerke [a], George Vosselman [a], Qing Zhu [b]

[a] University of Twente, Faculty of Geo-Information Science and Earth Observation (ITC), Hengelosestraat 99, P.O. Box 6, 7500AA, Enschede, The Netherlands

[b] State Key Lab of Information Engineering in Surveying, Mapping and Remote Sensing, Wuhan University, 129 Luo Yu Road, Wuhan, Hubei, 430079, PR China

ARTICLE INFO

Article history:
Received 29 July 2009
Received in revised form
13 April 2010
Accepted 11 May 2010
Available online 7 June 2010

Keywords:
Façade
Reconstruction
Knowledge
Image sequence

ABSTRACT

The paper presents an automatic method for the reconstruction of building models from video image sequences. These videos may be recorded using a hand-held camera or a camera mounted on a moving car. Such terrestrial video sequences are economic and flexible. Presenting buildings as geometric models – rather than for instance a representation from a simple meshing of 3D points – enables one to perform a wide range of analyses. However, sparse 3D points and 3D edges do not contain topological relations. Therefore, integrating building structure knowledge into the reconstruction steps plays an important role in our method. First, some rules are applied to reasonably group the extracted features. Then, a suitable outline and normal direction are specified for each surface patch. Based on these surface patches, a hybrid model- and data-driven method is used to recover a building model from both the extracted surface patches and hypothesized parts. Using the building structure knowledge leads to a simple and fast reconstruction method, and also enables one to obtain the main structures of buildings. The results show that this method correctly sets up topological relationships between generated surface patches and also obtains reasonable structure models in occluded areas. Therefore, the reconstructed models satisfy requirements for both visualization and analysis.

1. Introduction and related work

Applications such as Google Earth and Microsoft Bing maps (Virtual Earth) are very successful in delivering effective visualizations of the earth's surface based on aerial and satellite images to a broad audience. However, various fields demand realistic 3D city models, such as urban planning, virtual tourism, navigation and emergency response. Extraction and reconstruction of man-made structures from aerial images has been a topic of intense research for many years (Remondino and EL-Hakim, 2006). Because only roofs are well observed from aerial images, researchers focus on roof structures recovering from features (points, edges or segments) with or without other data, such as the building ground plane (Baillard and Zisserman, 1999; Henricsson and Baltsavias, 1997; Suveg and Vosselman, 2004). Although a lot of research is still devoted to this topic, it is still far from the goal of a fully automatic system. Recently 3D city models constructed from ground based data are becoming interesting as they represent realistic façades, which contain more details than models constructed from aerial data.

Commonly, ground based object extraction has mainly relied on manual operations with the support of image based modeling software, such as Image Modeler or Photo Modeler (Autodesk, 2010; Eos Systems, 2010). Due to the huge number of urban objects in a city and the variety of shapes, manual reconstruction of a city is a time-consuming and expensive procedure (Brenner, 2005). According to different input data and aims, some semi-automated and fully automated reconstruction methods have been presented (Debevec et al., 1996; Dick et al., 2004; Pollefeys et al., 2008; Pu and Vosselman, 2009; Werner and Zisserman, 2002).

Image sequences play an important role in many close-range applications in computer vision. Economic and flexible data acquisition procedures together with the automatic structure from motion approach are advantages of using video image sequences as the source data for object reconstruction. In recent years, the topic of 3D reconstructions of buildings and other landmarks from image sequences and video data has received much attention (Cornelis et al., 2008; Mayer and Reznik, 2007; Pollefeys et al., 2008; Snavely et al., 2008). Some of these systems use video data together with GPS and IMU information and produce detailed 3D models in the form of textured polygonal meshes in real time (Pollefeys et al., 2008). Other systems produce sparse reconstruction of various landmarks from internet photo collections, which can

[*] Corresponding author at: University of Twente, Faculty of Geo-Information Science and Earth Observation (ITC), Hengelosestraat 99, P.O. Box 6, 7500AA, Enschede, The Netherlands.

E-mail addresses: yixiangtian@yahoo.com, ytian@itc.nl (Y. Tian), gerke@itc.nl (M. Gerke), vosselman@itc.nl (G. Vosselman), zhuq66@263.net (Q. Zhu).

0924-2716/$ – see front matter © 2010 International Society for Photogrammetry and Remote Sensing, Inc. (ISPRS). Published by Elsevier B.V. All rights reserved.
doi:10.1016/j.isprsjprs.2010.05.001

be used for visualization (Snavely et al., 2008). Based on sparse 3D points and edges recovered from images, buildings can be shown as textured 3D models, and be compared with images from different periods for change detection (Schindler et al., 2007). A problem for these systems is that with an increasing demand for details the required computation time is also increasing. Some systems use sideways looking video to create multi-perspective images for visualizing roughly planar scenes or employ a simple model for geometry (Cornelis et al., 2008; Roman et al., 2004). However, these approaches aim at a visualization of buildings, and do not explicitly recover a boundary and face representation for individual buildings. Therefore, analyses of buildings for instance in a CAD or GIS environment is not possible. Werner and Zisserman (2002) proposed an approach that uses sparse points and edges to reconstruct ground and building planes from wide baseline images. From a vanishing point computation, three principal directions are reconstructed. They sweep vertical planes through space to determine the position which best matches the images. Then two generic models are used to fit some details. However, the two orthogonal horizontal directions that are required in their method are usually not available from video image sequences.

Image sequences have been used for building reconstruction for several years. If buildings are described by triangular networks (mesh), the constructed 3D geometry from the result of triangulating the dense point clouds is very useful to recover the details. However, it requires a lot of memory space and is greatly affected by depth errors. The geometry of buildings can also be described by parametric or polyhedral models. Parametric models are used for simple buildings, which can be described using a few parameters. However, they are not easy to extend for the description of complex objects. As the majority of buildings satisfy the assumption that they can geometrically be modeled as an ensemble of planar polygonal surface patches, using polyhedral models seems to be a relatively simple and efficient way to present building structures (Werner and Zisserman, 2002). Such representations with detailed roof structures and planar façades are for example sufficient for simulations or visualizations at small or medium scale and satisfy the requirement of LOD2 and LOD3 being defined by CityGML (Open Geospatial Consortium, 2008).

In principle, object reconstruction methods can be divided into data-driven and model-driven. Occlusions of buildings or building parts cause failure in the extraction of complete features. Therefore, some unreconstructed areas can be left or only the actually observed/detected features are connected if using a data-driven method only. A more improved method could be to integrate the structure's knowledge into the reconstruction in order to define reasonable hypotheses (Baltsavias, 2004). Model-driven methods ensure the plausibility and the topological correctness of the reconstructed objects. On the other hand, the enormous variations in the structure and shape of building façades prevent to use too tight constraints to recover the structure. A good compromise can be to take advantage of the flexibility of a data-driven approach and of the robustness of a model-driven one.

In this paper, knowledge of the building's structure is integrated into the reconstruction of buildings from video sequences. The term *knowledge* is widely used in many image analysis methods and it may describe any kind of information (Baltsavias, 2004). In this paper, it refers to rules or constraints that are deduced from a general building façade structure. Our approach takes automatically extracted 3D points and 3D edges as the basic features. Surface patches are obtained by grouping the basic features and a verification step. The basic façade shape knowledge is used to estimate their outlines. Topological properties within the planar building structure are considered during the surface patch connecting and model recovery step.

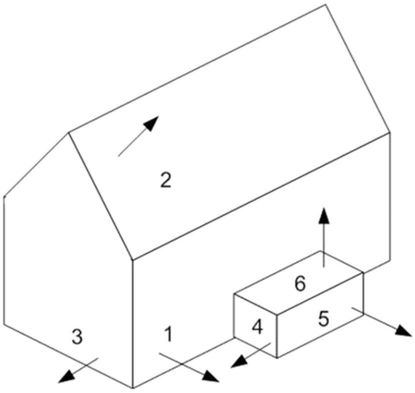


Fig. 1. Surface patches with their normal vectors.

The paper is organized as follows: Section 2 is an overview of our recent research in this field. In Section 3, the method of generating surface patches from sparse points and edges is described. Section 4 is about reconstructing building models from surface patches. Section 5 presents and discusses some building reconstruction results. Some conclusions are given in Section 6.

2. Method overview

2.1. Related concepts

In this section we present the proposed building model and the prior knowledge which is used in the developed strategy. In order to cope with the complexity of real scenes we propose an application specific modeling of buildings with planar structures, and not specific architectures. Some steps have already been introduced in Tian et al. (2009a,b).

Buildings reveal a high variability in structure. However the main façades contain most important geometrical, topological and texture information. We therefore propose a hybrid data-/model-driven method, which represents buildings as an aggregation of several simple building parts. This enables one to cope with the problems caused by occlusions, low contrast, noise and disturbances. Some concepts need to be specified before describing our reconstruction method.

Surface patch: A plane has a closed outline, represented by a polygon. The normal vector of the surface patch points to the outside of building. For example, if there is a building containing six surface patches (s_i, $i = 1 \ldots 6$) as shown in Fig. 1, the normal vectors of these surface patches must be in the same direction as indicated in the drawing.

Surface patch neighborhood relation: There are two kinds of neighborhood relations between surface patches. One is that two surface patches can be part of the same volume, such as surface (1, 2), (1, 3), (2, 3), (4, 5), (4, 6) and (5, 6) in Fig. 1. The other one is the edge from one surface which is contained in another surface, e.g., surface 4 and 6 are attached to surface 1.

Local model: A local model is constructed by three adjacent surface patches, and defines a part of a volume. Each local model has exactly one corner point where the three planes intersect, i.e. when representing the model as a graph, the corner is a node with degree three (Fig. 2).

Knowledge: Rules and constraints are retrieved from a generic building façade structure. In the next section, the knowledge about buildings is collected and presented as rules.

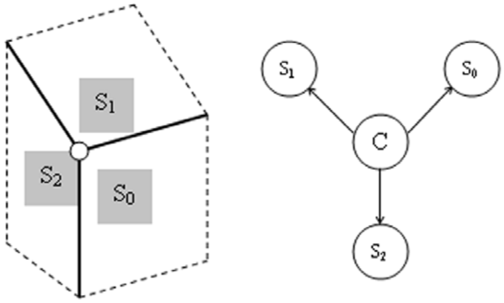


Fig. 2. Local model shows the topological relation of three surface patches (left). Drawn as graph the arrow express the neighborhood relation (right).

2.2. Knowledge presentation

In our opinion, a surface patch is the basic element for humans to recognize building structures, because surface patches implicitly reflect semantic information and relations between different building parts. 2D topological relationships have been intensively studied by Egenhofer and Franzosa (1991). As the extension of 2D topology, 3D topological relationships are much more complex, because 3D space has a number of quite difficult and unexpected situations and 3D–2D, 2D–2D, even 3D–1D relations need to be considered (Pigot, 1991; Pilouk, 1996). A general classification to satisfy all requirements is very difficult (Ellul and Haklay, 2006). We specify some useful subdivision classifications according to our application, which are based on relationships as proposed in Egenhofer and Herring (1990) and Pigot (1991). Knowledge is presented by different kinds of classifications.

Each surface patch s has a number of attributes, such as color, or material. In the context of this paper only geometric attributes as extracted from images are regarded further. In particular these are position, orientation and shape.

Semantic information: The main surface patches belong to walls, roofs, ground planes or extrusions. Extrusions include balconies and overhanging parts on the wall or roof. So, $\forall s, \mathit{Semantic}(s) = \mathit{Wall} \vee \mathit{Roof} \vee \mathit{Ground} \vee \mathit{Extrusion}$.

Position: Some surface patches can be expected at relative positions inside a scene. For example, the ground is usually the lowest part and other types might be close to the ground as well. From point and edge features, a 3D boundary box of the area of interest can be calculated. Surfaces exclusively located in the lower half part of the buildings' vertical extension are considered to be "low".

$$\forall s, \mathit{Position}(s) = \mathit{Low} \vee \mathit{NotLow}.$$

Orientation: The orientation of a semantic surface patch is predictable. For example, the ground is horizontal, walls and extrusions are usually vertical and roofs are not vertical. And they can be presented as $\forall s, \mathit{Orientation}(s) = \mathit{Horizontal} \vee \mathit{Vertical} \vee \mathit{Oblique}$.

Shape: Building surface patches have a regular and common shape, especially when buildings are presented in a generalized form. Triangle, rectangular, parallelogram trapezoid are basic shapes for surface patches. Building surface patches may also show as a combination of these basic shapes. For the ground, a convex hull is chosen to present it.

$$\forall s, \mathit{Shape}(s) = \mathit{Rectangular} \vee \mathit{Triangular} \vee \mathit{Trapezoid} \\ \vee \mathit{Parallelogram} \vee \mathit{Others}.$$

At this stage we want to restrict ourselves to some standard buildings, thus buildings that do not comply with the following specifications will not be regarded further:

- All building faces are planar.
- Walls are vertical.
- Roofs intersect with walls.

From the structural information of buildings we can also predict some *preferential* knowledge, which can be used to enhance the modeling, but this is not obligatory. To present this kind of knowledge, spatial relations between surface patches need to be defined. Some relations can have a numbered value, such as the *angle* between two surface patches and the *distance* between two surface patches. Other attributes are:

Adjacency: When the distance between two surface patches is zero, they are adjacent.

$$\forall s_1, s_2, \mathit{Adjacent}(s_1, s_2) = \mathit{True} \vee \mathit{False}.$$

Intersection: Intersection is a basic spatial relation between two surface patches, which is different from *through* (Pigot, 1991). We specify different intersection types according to the location of common edges on each surface.

$$\forall s_1, s_2, \mathit{Intersection}(s_1, s_2) = \mathit{ShareBoundaryEdge} \\ \vee \mathit{NotShareBoundaryEdge}.$$

Coherence: Coherence is the relation between normal vectors of two adjacent surface patches. Surface coherence means the normal vectors of surface patches should be consistent to a point interior or exterior to the volume. As shown in Fig. 3(a) and (b), two adjacent surface patches are coherent along their common edge if:

$$[l, v_1, n_1] \cdot [l, v_2, n_2] < 0 \tag{1}$$

where l presents the direction of the common edge,

v_i presents the orthogonal direction from the common edge to the interior of surface patch,

n_i presents the normal vector of the surface patch,

$[l, v_i, n_i]$ stands for the scalar triple product.

Fig. 3 also shows some counter examples in (c) and (d). So,

$$\forall s_1, s_2, \mathit{Adjacent}(s_1, s_2) = \mathit{True} \\ \Rightarrow \mathit{Coherent}(s_1, s_2) = \mathit{True} \vee \mathit{False}.$$

Concluding, surface patch neighborhood relations can be defined as: $\mathit{Relation}(s_1, s_2) = \mathit{Meet} \vee \mathit{Attach}$.

They are subdivisions of the original *meet* relationship (Egenhofer and Herring, 1990) and can be presented by other attributes:

$$\exists s_1, s_2, \mathit{Relation}(s_1, s_2) = \mathit{Meet} \\ \Leftrightarrow \mathit{Adjacent}(s_1, s_2) = \mathit{True} \wedge \mathit{Intersection}(s_1, s_2) \\ = \mathit{ShareBoundaryEdge} \wedge \mathit{Coherent}(s_1, s_2) = \mathit{True}$$

$$\exists s_1, s_2, \mathit{Relation}(s_1, s_2) = \mathit{Attach} \\ \Leftrightarrow \mathit{Adjacent}(s_1, s_2) = \mathit{True} \wedge \mathit{Intersection}(s_1, s_2) \\ = \mathit{NotShareBoundaryEdge}.$$

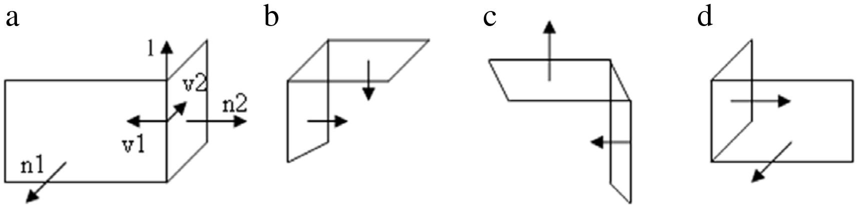


Fig. 3. Coherent and non coherent adjacent surfaces.

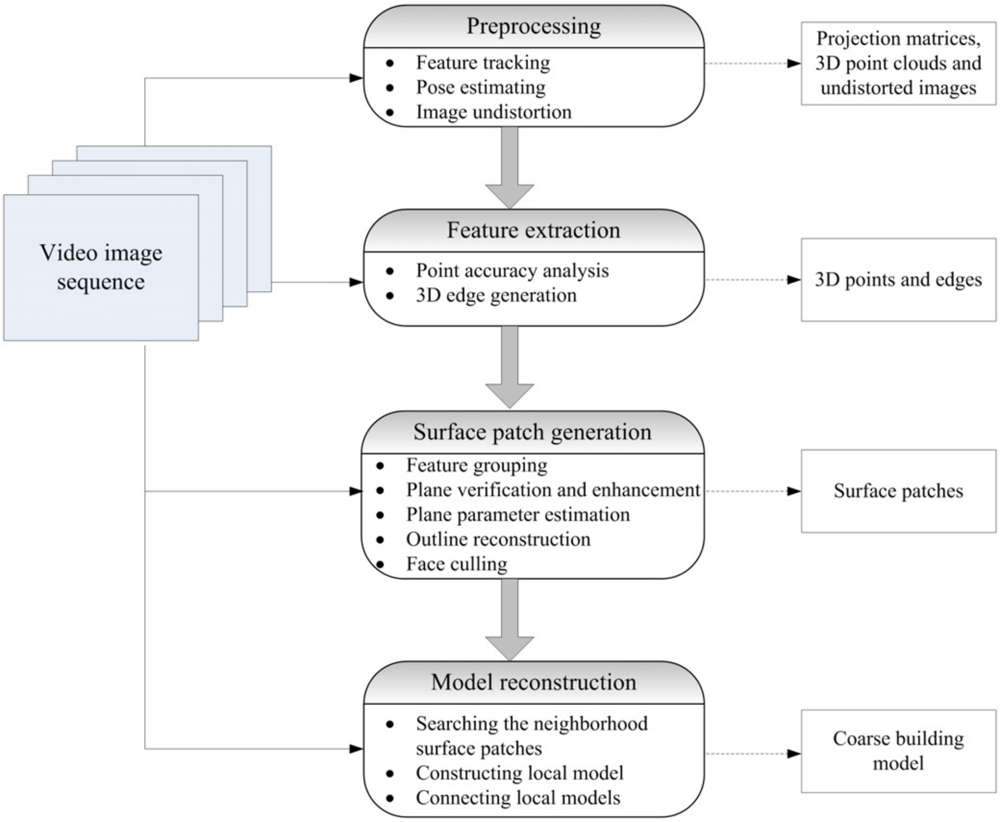


Fig. 4. The modeling pipeline for building reconstruction from image sequence.

We can also use them to present preferential knowledge as follows.

- Building ground planes mostly have rectangular angles.

  $\forall s_1, s_2, \mathit{Semantic}(s_1) = \mathit{Semantic}(s_2) = \mathit{Wall}$

  $\Rightarrow \mathit{if}\ \mathit{Relation}(s_1, s_2) = \mathit{Meet},$

  $\mathit{then}\ \mathit{Angle}(s_1, s_2) = \mathit{RightAngle}.$

- Repeating and regular structures are most common (they are an essential component in architecture design, often as result of economical, manufacturing, functional, or aesthetic considerations). For example:

  $\forall s_1, s_2, \mathit{Semantic}(s_1) = \mathit{Semantic}(s_2) = \mathit{Roof}$

  $\Rightarrow \mathit{if}\ \mathit{Orientation}(s_1) = \mathit{Oblique},$

  $\mathit{then}\ \mathit{Orientation}(s_2) = \mathit{Oblique}.$

The preferential knowledge provides essential guidance when there are not enough features extracted to compute a reasonable reconstruction. It is used during fitting of the surface patch's outline, finding the adjacent surface patches and further for formulating the local model hypotheses.

2.3. Workflow

The workflow consists of the following steps (cf. Fig. 4):

1. Preprocessing: After feature tracking across the sequence, the projection matrices and 3D coordinates of feature points are computed through bundle adjustment, and the lens and image distortions are corrected for, i.e. undistorted images were used thereafter.
2. Feature extraction: Starting from 3D points tracked from a video image sequence, the point accuracy is analyzed first to obtain reliable matched points. In order to introduce more constraints for the reconstruction and to fill the gaps in 3D point clouds, 3D edges are also used as primitives for the reconstruction.
3. Surface patch generation: First, the extracted 3D points and edges are grouped and verified according to our rules, see Sections 3.1 and 3.2. After estimating the plane parameters from all the edges and points in the plane (Section 3.3), the knowledge about generic shapes of building surfaces guides the outline generation (Section 3.4). Finally, the normal direction is defined in such a way that it points to the outside of the building (Section 3.5).
4. Model reconstruction: For connecting sparse surface patches, building structure knowledge is integrated into the model's reconstruction. Adjacent surface patches are searched first (Section 4.2). Then local models are recovered by coherent adjacent surface patches (Section 4.3). Finally, local models are connected to form a complete model (Section 4.4). The topologic relation between surface patches is set up during the local model's construction and helps in connecting different local models.

2.4. Preprocessing and feature extraction

In most cases when applying the method as described in this paper, uncalibrated, non-metric cameras are used and precise navigation information through GPS/IMU is normally not accessible. Thus, there is no further information about the image's exterior and interior orientation available. The initial step consists

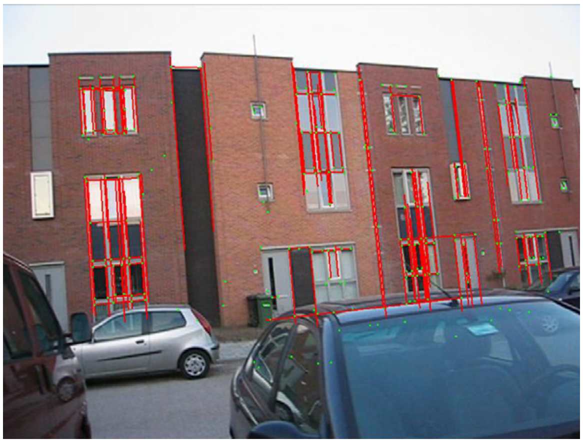


Fig. 5. Feature extraction result, reliable points (dot), matched edges (finite line).

in relating the images to each other. Usually when dealing with video image sequences, this step is done through feature tacking. The most widely used tracker is the KLT tracker (Lucas and Kanade, 1981). By determining 2D–2D point correspondences in consecutive video frames, the camera's relative geometry is established. Further information on the structure from motion can be found in many references, such as Hartley and Zisserman (2004) and Nistér (2004). For the implementation of the workflow at hand, the commercial software Boujou (Vicon, 2010) is currently used. Beside the fully automatic reconstruction up to scale, it is possible to define a coordinate frame and constraints on the actual scene geometry, like known distances in the object space between feature points. Radial distortion coefficients are estimated as well and images can be undistorted accordingly. Refer to Dobbert (2005) for detailed information on the approach as implemented in Boujou.

The method for extracting edge features is presented in Tian et al. (2008). A brief introduction is given here because it is the basis of our work and the feature points and edges are used in the following steps. Starting from 3D points tracked from a video image sequence, the point accuracy is analyzed first to obtain reliable matched points. Only edges near the reliable matched points are considered as edge candidates and all the edge candidates are used to estimate 3D edges. Based on the estimated variance factor, only good 3D edge estimations are accepted, which ensures the accurate position of matched 3D edges. In order to introduce more constraints for the reconstruction and to fill the gaps in 3D point clouds, 3D edges are also used as primitives for further reconstruction.

Fig. 5 shows extracted 3D points and edges projected on the first frame of a video image sequence. The video was taken by a hand-held camera. The camera was oriented sideways and captured the façades of buildings. A 3D view on the edge extraction result is shown in Fig. 6. From the figures it is obvious that only sparse points and edges are extracted and some edges which coincide with the building's outline are missing or not complete. From this observation we motivate the necessity to incorporate object knowledge into the reconstruction of the surface patches. Beside the 3D edges, the vertical direction which is important for the following steps can be easily obtained. In the preprocessing steps, the scale of the local coordinate system is computed approximately through some reference distance measures at the object. Therefore, in the following sections thresholds are defined in the world coordinate system.

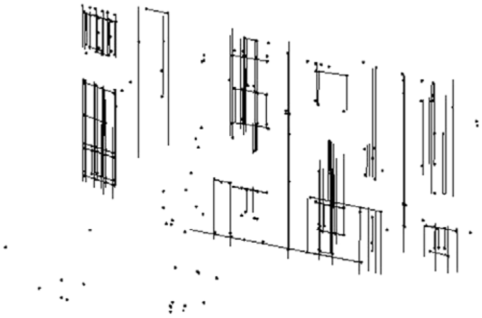


Fig. 6. 3D view of feature extraction result in Fig. 5.

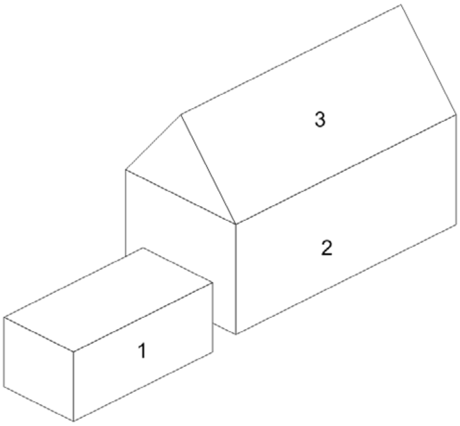


Fig. 7. Example for different surface patches on the same plane.

3. Surface patch generation

In this section we introduce the method for grouping the extracted 3D points and 3D edges to surface patches. The main problem in this step is how to recognize feature points and edges that belong to the same surface patch. For example surface patch 1 and 2 in Fig. 7 are at the same infinite plane, but are actually separated. This example shows that coplanarity is the necessary but not sufficient condition for point and edge grouping. Therefore some constraints need to be defined for feature grouping and outline generation in order to find a reasonable surface patch. Only geometric features are considered during surface patch generation, which means both the grouping and verification are based purely on extracted features.

3.1. Feature grouping

We consider cues from point cloud segmentation, intersecting edges and parallel edges and apply these cues in sequence. As some points and edges may lie on the boundary of planes, our grouping method allows an overlapping clustering result.

3.1.1. Point cloud segmentation

Usually the main building façade is the largest plane with some windows, doors and symbols to let people separate it from other buildings on the grounds. So it probably contains more salient feature points than other parts of the building. Therefore point segmentation can help to recognize the main façades. As shown in Fig. 8, two vertical walls can be reconstructed from the result of point cloud segmentation.

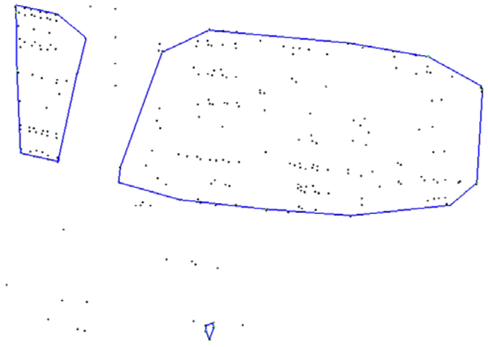


Fig. 8. Point cloud segmentation result of data in Fig. 6, reliable points, point segments with convex outline.

We adopt the planar surface-growing algorithm by Vosselman et al. (2004), which consists of a seed surface detection followed by the actual growing of the seed surface. Because this method is mainly used for laser scanning data, that is much denser than a point cloud extracted from an image sequence without using a dense matching technique, we choose a large surface growing radius according to the density of points. This leads to some segments that do not exist in the real façade. Only the vertical point segments are considered as plane hypotheses and wrong hypotheses will be removed in a further step described in Section 3.2.

3.1.2. Two intersecting edges

If two 3D edges (e_1, e_2) intersect, they must be in the same plane. However, we do not want to group edges that are not corresponding to the actual building façade. For example, edge 1 and edge 2 in Fig. 9 are intersecting in object space. Such a plane hypothesis can be avoided by considering the relation between the intersection point and two edges: If the distance between the intersection point and edge is larger than a threshold, the hypothesis is rejected. The distance between the intersection point and the finite 3D edge is defined by the relationship between this intersection point and two endpoints of the edge. If the point is located between two endpoints, the distance value is zero. Otherwise, the distance is the minimal value of the distance between the point and endpoints, which is applicable to edge 1 and edge 2 in Fig. 9. Using the same way of knowledge representation as before, this cue can be written as:

$$\forall e_1, e_2, \mathit{PlaneFromIntersectingEdges}(e_1, e_2) = \mathit{True} \vee \mathit{False}.$$

3.1.3. Two parallel edges

Many parallel or almost parallel edges exist on the building façade. However, nearby parallel edges are not reliable to make a surface patch hypothesis. For example in Fig. 9, edge 4 and edge 5 could result in an un-realistic plane hypothesis, because edge 5 is close to the wall where edge 4 and edge 6 are located. A minimal threshold between two 3D edges is specified to avoid problems caused by features from detailed parts. The sequence of edge processing within this kind of reasoning should not be arbitrary. Therefore, additional rules need to be applied. One is for each edge a hypothesis is first made from the parallel edge with the smallest distance above the minimal threshold. The other is if any one of the two edges was already grouped to a surface patch, we search for other parallel edges belonging to that patch and choose the two with smallest distance to make a hypothesis.

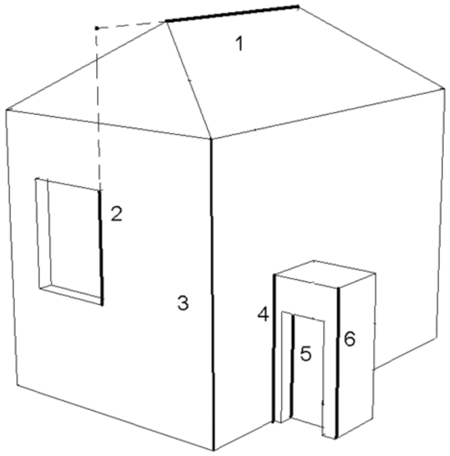


Fig. 9. A simple building with extracted edge 1–6.

For example, edge 3, edge 4 and edge 6 are parallel and extracted in that sequence. According to our rules, edge 3 and edge 4 form a surface patch before a hypothesis from edge 3 and edge 6. So the plane hypothesis made from parallel edge 3 and edge 6 can be avoided. Similar to the cue of the intersecting edges, this can be expressed as:

$$\forall e_1, e_2, \mathit{PlaneFromParallelEdges}(e_1, e_2) = \mathit{True} \vee \mathit{False}.$$

3.2. Plane verification and enhancement

After defining the plane hypotheses, they are verified and enhanced by 3D edges that were not used for grouping so far. Two parameters must be given first to decide whether an edge belongs to a plane: A threshold ε determining the maximum distance of endpoints from the plane and the maximum angle θ between the edge and the plane. The threshold for the angle is currently a fixed value (10°). Therefore, the relation between a surface patch (s) and an edge (e) is:

$$\forall e, s, \mathit{EdgeOnPlane}(e, s) = \mathit{True} \vee \mathit{False}.$$

As our method focuses on main building structures, features within 0.2 m can be ignored and above 1 m have to be identified. Therefore, the default values of ε and the minimal threshold of parallel edges are 0.2 m. The default value of the maximum distance between intersecting points and edges, as well as for two parallel edges is 1 m.

The way to verify and implement cues is related to the particular cues, which are used to define the hypothesis. For the first type, i.e. the point cloud based cue, we segment all the points at one time. For each segment, a convex hull is calculated. Then the verification is done by testing whether there are some 3D edges belonging to that plane and located in that region. If there is no edge belonging to it and it is not parallel with the vertical direction, the corresponding plane hypothesis is rejected. Otherwise, a final surface patch is computed by these points and edges.

For the other two edge-based cue types, edges are checked pairwise. Thus this step is to grow a small surface, which is generated from two edges, to a bigger one by adding more edges to it. As our method allows an overlapping clustering result, edges near a surface patch boundary can be grouped to more than one surface patch. The implementation of intersecting edge-based grouping method is given below as an example.

```
**Inputs**: An array of 3D edges = $edges$
**Initialize**: $i = 0, j = 0$
**For** $i$ **from** 0 **to** $edges.size - 1$
  **For** $j$ **from** $i + 1$ **to** $edges.size$
   **If** $PlaneFromInter\sec tingEdges(edges[i], edges[j]) == true$ **then**
      Initialize: a new surface patch $s$, $count = 0$, $subedgeslist = empty$
      Calculate plane parameters for $s$
      **While** $count < edges.size$ and $count \neq i$ and $count \neq j$ **do**
       **If** $EdgeOnPlane(edges[count], s) == true$ **then**
         Push $count$ to $subedgelist$
       **End if**
       **End while**
       **If** $subedgelist.size > 0$ **then**
         Push $i, j$ to $subedgelist$
         Reconstruct surface patch outline from edges corresponded to $subedgelist$
         Identify edges near outline and remove their numbers from $subedgelist$
         Remove edges that correspond to $subedgelist$ from $edges$
         Push $s$ to surface patch set
       **End if**
      **End if**
  **End for**
**End for**
```

3.3. Plane parameter estimation

Plane parameters are obtained using all the edges and points in the plane. All possible combinations of two edges are used to define the plane and remaining points and edges are projected to the plane. The one with the least residual RMS is chosen as the best fit. As some points and edges may lie on the boundary of two planes, we label the edges inside the respective region after defining the outline of a plane. Therefore, the edges at the boundary can be grouped into more than one surface patch. This overlapping cluster method results in more surface patches, but requires an effective outline reconstruction method that can present the patches' shape and correctly judge edges at the boundary.

3.4. Outline reconstruction

Through the orientation and position of the surface patch, its semantic information can be concluded and together with the position and orientation of the edges a reasonable shape can be derived (defined in Section 2.2):

$$\left.\begin{array}{r} Orientation(s_i) \wedge Position(s_i) \Rightarrow Semantic(s_i) \\ Position(E_i) \wedge Orientation(E_i) \end{array}\right\} \Rightarrow Shape(s_i).$$

Using the estimated normal direction, the 3D outline reconstruction problem can be simplified to a 2D problem by rotating the plane into the XY-plane. This is possible because we only assume planar faces. Based on the position and orientation relation between edge and surface, which are decided by corresponding edge and surface coverage in XY-plane in rotated space, the best fitted generalized shape is chosen for this surface patch.

3.5. Face culling

In the above steps, the normal vector of a surface patch can point either to the outside or to the inside of the building. As the further reconstruction method depends on the relation between the adjacent surfaces' direction, the normal vector must be homogenously oriented for each surface patch. Because of our estimation method only the surface patches that are visible from the camera positions are generated. This means that the dot product of the plane's normal (N) and the viewing direction (C) should satisfy the following equation:

$$dot(N, C) \geq 0. \tag{2}$$

As the sequence of boundary points is consistent with the normal vector in a right-hand coordinate system, the sequence must be modified if the normal vector of a surface patch is changed to the opposite direction.

4. Building model reconstruction

In our approach, the building model's reconstruction can be seen as a process of connecting generated surface patches. The perfect situation is achieved when all surface patches of the building are correctly recovered. However due to occlusions, failed or wrong extraction, there are always some parts that are missing or wrong. Meanwhile, some building edges are not represented by salient image edges. As our surface patch generation method is based on extracted point and edge features, the pure geometric reconstruction fails to determine some surface patches and the outlines of some surface patches may need modification. Topological constraints help to make hypotheses on occluded areas and to define relations between different surface patches. During the reconstruction step, some surface patches will be extended and new surface patch hypotheses need to be verified. The verification method will be introduced (Section 4.1), which is followed by our method for the building model's reconstruction. There, surface patch neighborhood relationships are set up first (Section 4.2). The local model hypotheses then are made based on adjacent surface patches (Section 4.3). Afterwards, all local models are connected to form a complete building model (Section 4.4).

4.1. Surface patch verification

If a surface patch is visible in the images, we can get a high cross-correlation result for the particular areas as visible in any two images. So we compute the similarity over the image sequence to evaluate the surface patch, which is similar to Baillard and Zisserman (1999). Instead of computing the correlation with respect to the whole surface we only consider it at endpoints of edges and compute the respective similarity within a 5 × 5 window. This processing is motivated by the observation that whole surfaces may be partly invisible in one frame, e.g. through occlusion, biasing the correlation and making an overall verification difficult.

In more detail, given the plane π, there is a homography represented by 3×3 matrix H_i between the first and ith frame, so that corresponding points are mapped as

$$x_i = H_i x_0 \tag{3}$$

where x_0 and x_i are image points that corresponding to the same object point X and are represented by homogeneous 3-vectors.

The homography matrix is obtained from 3×4 camera projection matrices for each frame. For example, if the projection matrices for the first and ith frame are $P_0 = [I|0]$ and $P_i = [A|a]$, and a plane defined by $\pi^T X = 0$ with $\pi = (v^T, 1)^T$, then the homography induced by the plane is:

$$H_i = A - av^T. \tag{4}$$

In our process, we choose the first frame in which the surface patch is visible as the reference frame and compute the cross-correlation between the reference frame and frames within the frames' visibility range. The homography for $x_i = H_{ri}x_r$ is:

$$H_{ri} = H_i H_r^T \tag{5}$$

where i presents the ith frame, r presents the reference frame.

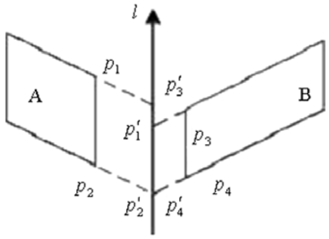


Fig. 10. Two intersecting surface patches.

The points of interest are endpoints of the 2d edge extraction result within the projected surface area in the reference frame. Points that are regularly distributed over that image area are also chosen. The similarity score for the average cross-correlation value for points ($x_{rj}, j = 0, \dots, m$) in the valid images (for example, $n+1$ frames in total) is:

$$sim = \sum_{j=0}^{m} \left(\sum_{i=1}^{n} Cor^2(x_{rj}, H_{ri}x_{rj})/n \right) \Big/ (m+1). \tag{6}$$

If the similarity score is higher than the threshold (0.8 based on experiments from images), the hypothesis is accepted. Otherwise, the hypothesis is rejected.

4.2. Searching adjacent surface patches

When intersecting two surface patches, we can get a finite intersecting line l and four intersecting points p_1', p_2', p_3' and p_4' from the boundary edges and intersecting line as shown in Fig. 10.

Points p_1 and p_2 are the nearest boundary points of p_1' and p_2' in the left surface patch. Similarly, points p_3 and p_4 are the nearest boundary points of p_3' and p_4' in the right surface patch. If the distance between p_i and p_i' is large, a surface extension may not be reliable. So this distance d is one parameter used to evaluate the surface patch found.

$$d = \left(\sum_{i=1}^{4} |p_i - p_i'| \right) \Big/ 4. \tag{7}$$

From Fig. 10, we can also find that edge $l_{(p_1', p_2')}$ and edge $l_{(p_3', p_4')}$ must have some overlap. The percentage of overlap on the intersecting edge can be used as a parameter. We choose points $\min(p_2', p_4')$ and $\max(p_1', p_3')$ to define the intersecting edge. Because our intersecting line (l) is defined by a foot point (p_f) and a direction vector (n), each point can be presented by the corresponding scalar (t).

$$l = p_f + n \cdot t \tag{8}$$

$t_1, \dots, t_4$ are defined as scalars associated to points $p_1', \dots, p_4'$ and let t_1 be larger than t_2 and t_3 larger than t_4. Therefore,

$$\rho = \begin{cases} 0, \quad if\ t_1 \le t_4 \text{ or } t_2 \ge t_3 \\ (\min(t_1, t_3) - \max(t_2, t_4))/(\max(t_1, t_3) - \min(t_2, t_4)), \\ \quad \text{else} \end{cases} \tag{9}$$

where ρ is the overlapping ratio and $\rho \in [0, 1]$.

According to our preferential knowledge as defined in Section 2.2, a building ground plane mostly is composed out of rectangles. Thus, the angle $\theta \in [0°, 180°]$ between surface patches is also considered. Considering the same weight for the overlap ratio and angle between them, $\sin(\theta) \in [0, 1]$ is used as another evaluator. Obviously, if two surface patches are parallel or have no overlap, they cannot be adjacent surface patches. If the value of d (Eq. (7)) is very small, for example smaller than the standard deviation of distance for extracted features to the plane they are grouping to, the adjacency can be ensured. So the small distance may imply a higher probability. We simply combine the three similarity measures by:

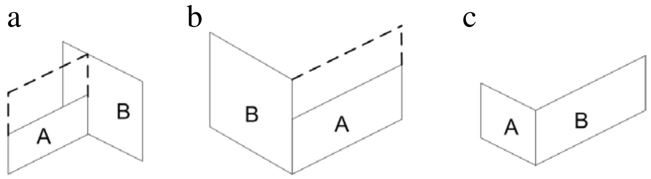


Fig. 11. Example for intersecting two surface patches, (a) the intersecting line is at the boundary of one surface patch and contained in another one, (b) the intersecting line is at the boundary of both surface patches but intersecting points are not consistent, (c) the intersecting line is at the boundary of both surface patches and intersecting points are consistent.

$$\varepsilon = f(d, \rho, \theta) = \rho \cdot \sin(\theta) \cdot d^{-1}. \tag{10}$$

So the surface patch with biggest ε is considered adjacent to the current surface patch.

Obviously, if two surface patches contain the same edge, they must be adjacent. So, we deal with those surface patches first, then we search for adjacent surface patches along boundary edges based on the above method. As introduced in the related concepts section, there are two kinds of neighborhood relations. These two types can be separated by the location of intersecting lines. Some examples for intersecting two surface patches are shown in Fig. 11.

In Fig. 11(a), the intersecting line is contained in surface patch B and the intersecting points are not consistent. The verification method is used to check whether an extension part (shown as dashed boundary) of surface patch A is valid or not. In Fig. 11(b), the intersecting line is at the boundary of both surface patches but intersecting points are not consistent. If such an extension is accepted, surface patch B will be divided into two parts in Fig. 11(a) and the relation between two surface patches in Fig. 11(b) is the same as Fig. 11(c). Otherwise surface patch A is attached to surface patch B. If surface patches are parts of the same volume, intersecting points must be consistent. Due to a possibly imperfect generation result, a small threshold (0.2 m by default) is allowed. We modify the intersecting points as $\min(p_2', p_4')$ and $\max(p_1', p_3')$ to define the intersecting edge of surface patches, which results in Fig. 11(c) and can be presented as:

$$Intersection(s_A, s_B) = ShareBoundaryEdge.$$

4.3. Constructing local model

This step only considers surface patches with the first kind of neighborhood relation, $\forall s_1, s_2, Relation(s_1, s_2) = Meet$ (defined in Section 2.1). The preferential knowledge provides essential guidance when there are not enough features extracted to make a decision.

Each local model is formed by three adjacent surface patches. If they are all extracted, a local model can be formed by intersecting them. However, mostly, only two of them can be observed. So our construction method starts from two adjacent surface patches. As shown in the workflow in Fig. 12, new surface patches are made and verified, if neighborhood surface patches are not coherent.

Such a non coherent situation usually is caused by over grouping features from surface patches with similar plane parameters. So if two nearby surface patches are not coherent, the boundary of one surface patch needs to be modified and a new surface patch hypothesis can be made based on these two surface patches to form a suitable local model.

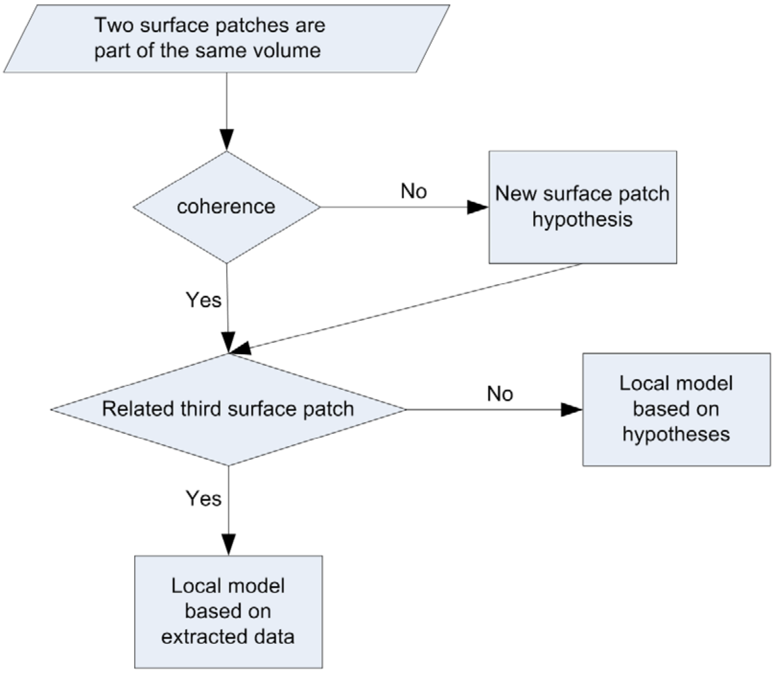

Two surface patches are part of the same volume
coherence
No
New surface patch hypothesis
Yes
Related third surface patch
No
Local model based on hypotheses
Yes
Local model based on extracted data

Fig. 12. Workflow for constructing a local model.

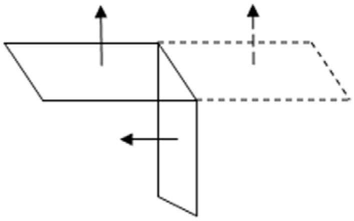


Fig. 13. Example for new surface patch hypothesis.

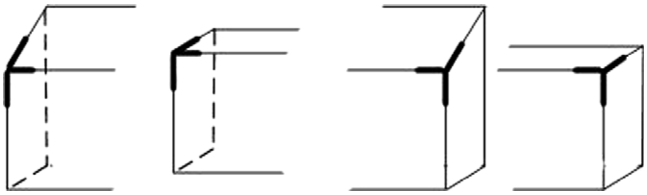


Fig. 14. Example for local models hypotheses.

Fig. 13 shows one example how to make a new surface patch hypothesis. We extend one surface patch versa the common edge to make a new patch hypothesis. Here we assume that the normal direction as computed is already facing outside the volume. The new surface patch and the unmodified surface patch must be coherent. Then the verification method mentioned in Section 4.1 is applied to this new surface patch. If the surface hypothesis is accepted, a local model hypothesis can be made.

After two coherent adjacent surface patches are found, not only can their topological relation be built, but also a local model is constructed based on them. During this step, a surface patch to support the hypothesis is searched first. The searching step is similar to the above step for searching adjacent surface patches. But here two boundary edges should be considered. If three surface patches that form a local model are found, the local model is defined by them. Otherwise, we choose simple block types, based on the building structure defined before, as for example shown in Fig. 14 to fit them. As a building's ground mostly has rectangular angles, we assume a vertical wall is perpendicular to its neighborhood wall when it is self-occluded. The above two sentences can be presented as an example below:

$$\begin{aligned}
&\exists s_1, \mathit{Semantic}(s_1) = \mathit{Roof}, \qquad \exists s_2, \mathit{Semantic}(s_2) = \mathit{Wall}\\
&\Rightarrow \mathit{if}\ \mathit{Relation}(s_1, s_2) = \mathit{Meet} \wedge \neg(\exists s, \mathit{Relation}(s, s_1)\\
&\quad = \mathit{Meet} \wedge \mathit{Relation}(s, s_2) = \mathit{Meet}), \mathit{then}\\
&\{s\} = \{s\} + s_n \wedge \mathit{Angle}(s_n, s_2)\\
&\quad = \mathit{RightAngle} \wedge \mathit{Relation}(s_n, s_1)\\
&\quad = \mathit{Meet} \wedge \mathit{Relation}(s_n, s_2) = \mathit{Meet}.
\end{aligned}$$

If the third plane is visible in the image sequence, the verification test is used. For example some roofs can be visible from the ground; the possibility of an oblique roof is tested. Otherwise, the third surface patch is generated based on common building structures. It is reasonable but could be inaccurate.

Another general hypothesis is that a roof is horizontal if there is no extracted feature on the roof. When two coherent vertical surface patches and some features above them are found, the plane sweeping method is used to find a more reliable roof plane. From the node that will have degree three after local model generation, the two non-vertical edges can form the initial roof plane hypothesis. If the initial roof hypothesis is rejected, a new roof hypothesis is rotated around these two edges. The optimal angle is computed by searching for the maximum of function *sim* (cf. Eq. (6)) over a range $[-\frac{\pi}{6}, \frac{\pi}{6}]$ with 1° each time. The roof position is the hypothesis with the maximum similarity score.

4.4. Connecting separate local models

During a local model's construction, the topological relationship between surface patches is established as well. However, a surface patch is only related to two surface patches at once during a local model's construction. Relations between neighborhood surface patches are recorded by each edge's connecting parameter and the whole building model is reconstructed by connecting local models one by one. If the building belongs to simple block types, the building model is already recovered during the local model construction step. For complex buildings, there are always some surface patches that are attached to surface patches being part

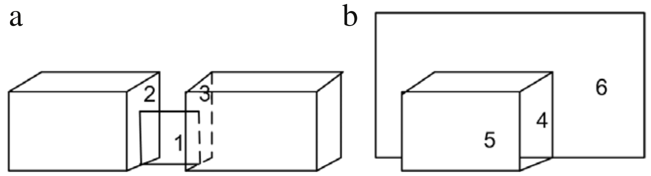


Fig. 15. Example for connecting separate local models.

of a different local model. They usually belong to extrusions and intrusions of buildings. Fig. 15 shows two examples. Two edges of surface patch 1 in (a) are contained in different surface patches (2 and 3). Surface patch 4 belongs to the same local model as surface patch 5 and one edge of it is contained in surface patch 6. These surface patches connect separate local models and are generalized by simple block types.

5. Experiments

Fig. 16 shows the surface patch generation result for the building façade shown in Figs. 5 and 6. The video was captured by a hand-held SONY camera. The images have a resolution of 640 × 480 pixels and a frame rate of 30 frames per second. There are 134 frames in total in this case.

Fig. 17 shows the building reconstruction result. As the video image sequence only shows parts of façades along the street, the building model is not complete. However, the observed four surface patches have been reconstructed with the correct topological relation. One self-occluded wall is recovered from the hypothesis.

More examples are shown in Fig. 18. The result in Fig. 19 shows that the structures of these building façades are successfully recovered. The first two examples are captured by the same camera as mentioned above. Due to the low resolution of images and the homogeneous color within the building façades, only few feature points and edges were extracted at the actual building edge areas. Therefore, most of the extracted surface patches need to be extended. The surface boundaries that are obtained from intersecting two neighborhood surface patches are more accurate than others based on visual examination. Only six surface patches of building (1) are visible in the images. They are reconstructed with correct topological relation. For building (2), three parts can be observed from the images. One part only has one floor. Another part has two floors and one big extrusion is attached to it. Some self-occluded surfaces are reconstructed based on general building structure hypotheses. The completeness evaluation is given in Table 1.

The image sequence of the third example in Fig. 18 was taken by a camera mounted on a trolley. Some surface patches of the stadium are almost on the same plane and few reliable features are extracted from the roof and left part of the building. To avoid the extension of surface patches to areas belonging to other surface patches, our feature grouping rules play an important role. The recovered structure presents the planar part of the stadium and it is supposed to satisfy the requirements of LOD2. As the roof does not intersect exactly with walls as our assumption, the intersecting line is on the roof. Meanwhile, there are some surface patches lost due to a few features being extracted in those areas, such as intrusion parts (two missing planes are in the red circle in Fig. 18). From the results we can also observe that some surface patches' outlines are not very accurate, see incorrect projection to the images. So the accuracy for surface patches needs to be improved in the further work.

The relation between surface patches can also be used to evaluate the results. According to our preferential knowledge as defined in Section 2.1, a building ground plane mostly has rectangular angles. For extracted surface patches that are visible in image sequence and which could stratify this rule, the angles are computed. The angle between potentially parallel surface patches is also computed, as shown in Table 2. For buildings (1) and (2), the deviations are less than 5 degrees. For building (3), three similar surfaces 2, 4 and 6 show their small difference in normal vectors. Because the slope between the oblique roof and their intersecting wall can be similar within the same building, the angles between surface 1 and 2, 3 and 4, 5 and 6 in building (1) and (3) are considered. The results confirm such a suggestion.

We have airborne laser scanning data with a point density of 20 pts/m^2 over the city of Enschede, in which buildings (1) and (2) are located. The heights of different floors are measured from laser data and compared with our results. As our results are based on a coordinate system computed from the first few frames, the

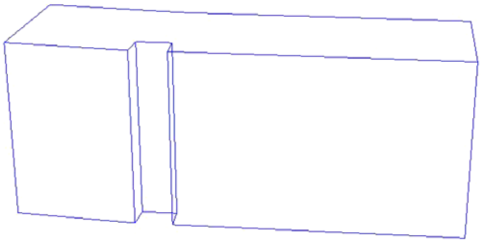
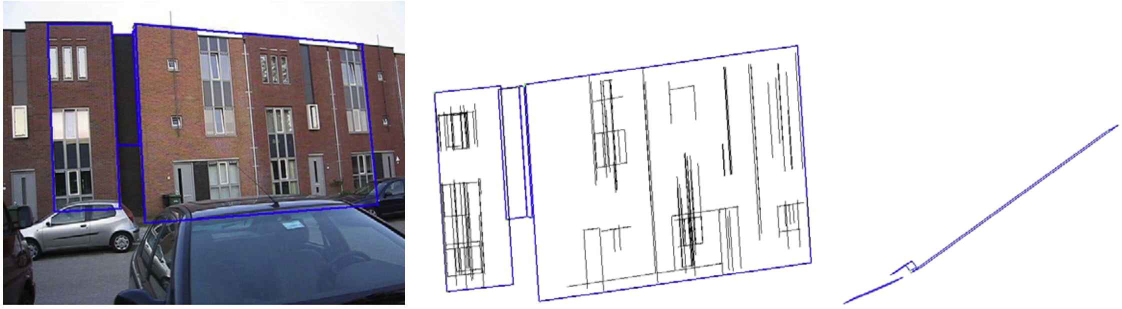


Fig. 17. Reconstruction result of building façade in Fig. 16.

Table 1
Completeness evaluation data for reconstructed buildings in Fig. 19.

Building no.	Reconstructed surface patches	Self-occluded surface patches	Observed surface patches	Missed surface patches
(1)	6	0	6	0
(2)	12	5	7	0
(3)	16	4	14	2



(a) Projection on one frame. (b) 3D side view. (c) 3D top view.

Fig. 16. Surface patch generation result of building façade in Figs. 5 and 6, constructed surface patches (blue). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

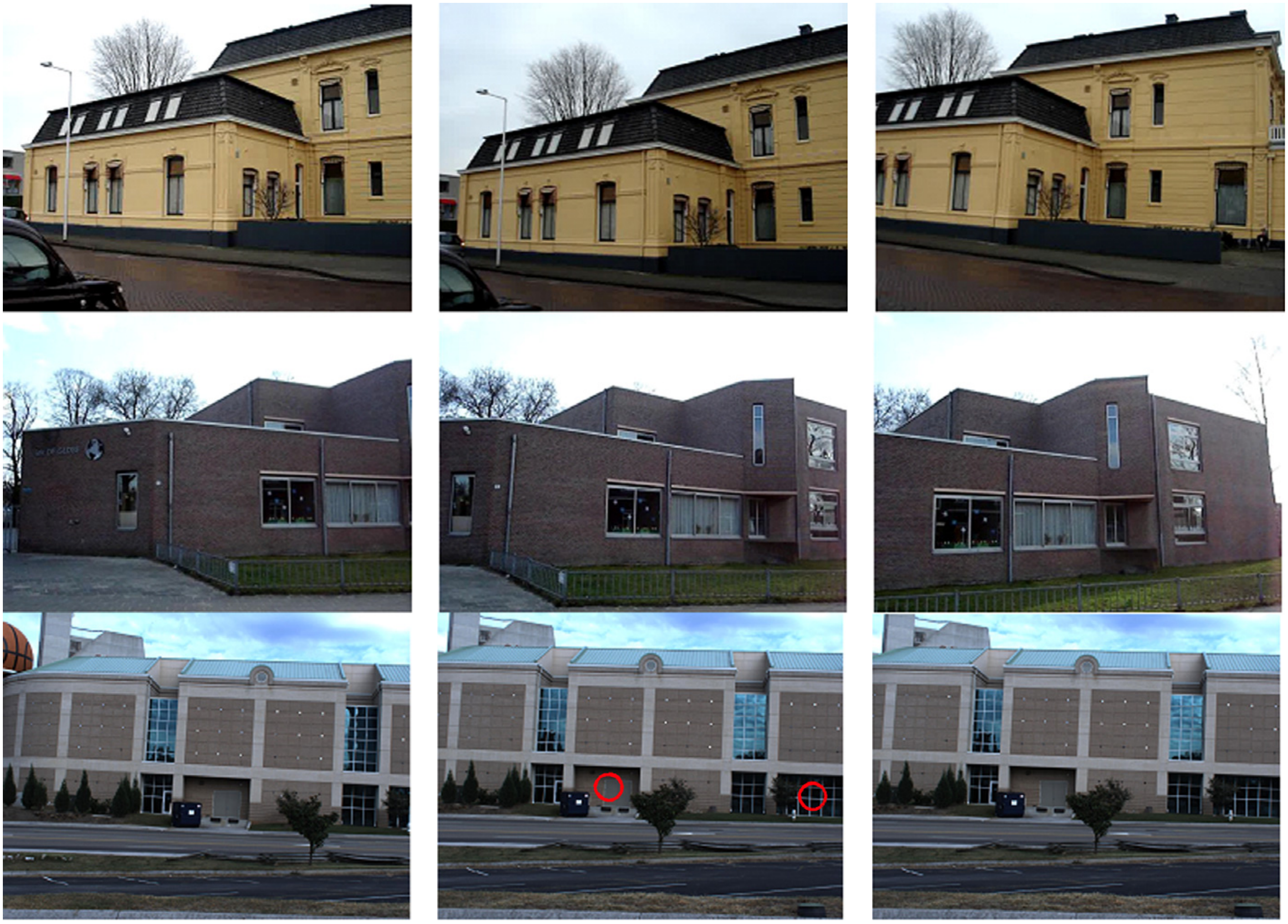

(a) First frame. (b) Middle frame. (c) Last frame.

Fig. 18. Frames from image sequences (1) Yellow House (126 frames) (2) Brown House (189 frames) (3) Stadium (61 frames). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Table 2
Angle (degree) between surface patches in Fig. 19.

Building no.	1 and 2	3 and 4	5 and 6	2 and 4	4 and 6	2 and 6
(1)	51.34	48.67	52.75	89.35	91.13	0.58
(2)	94.22	93.17	–	92.29	90.59	1.71
(3)	60.44	62.25	62.25	0.50	0.19	0.67

Table 3
Height values and ratios of surface patches in Fig. 19.

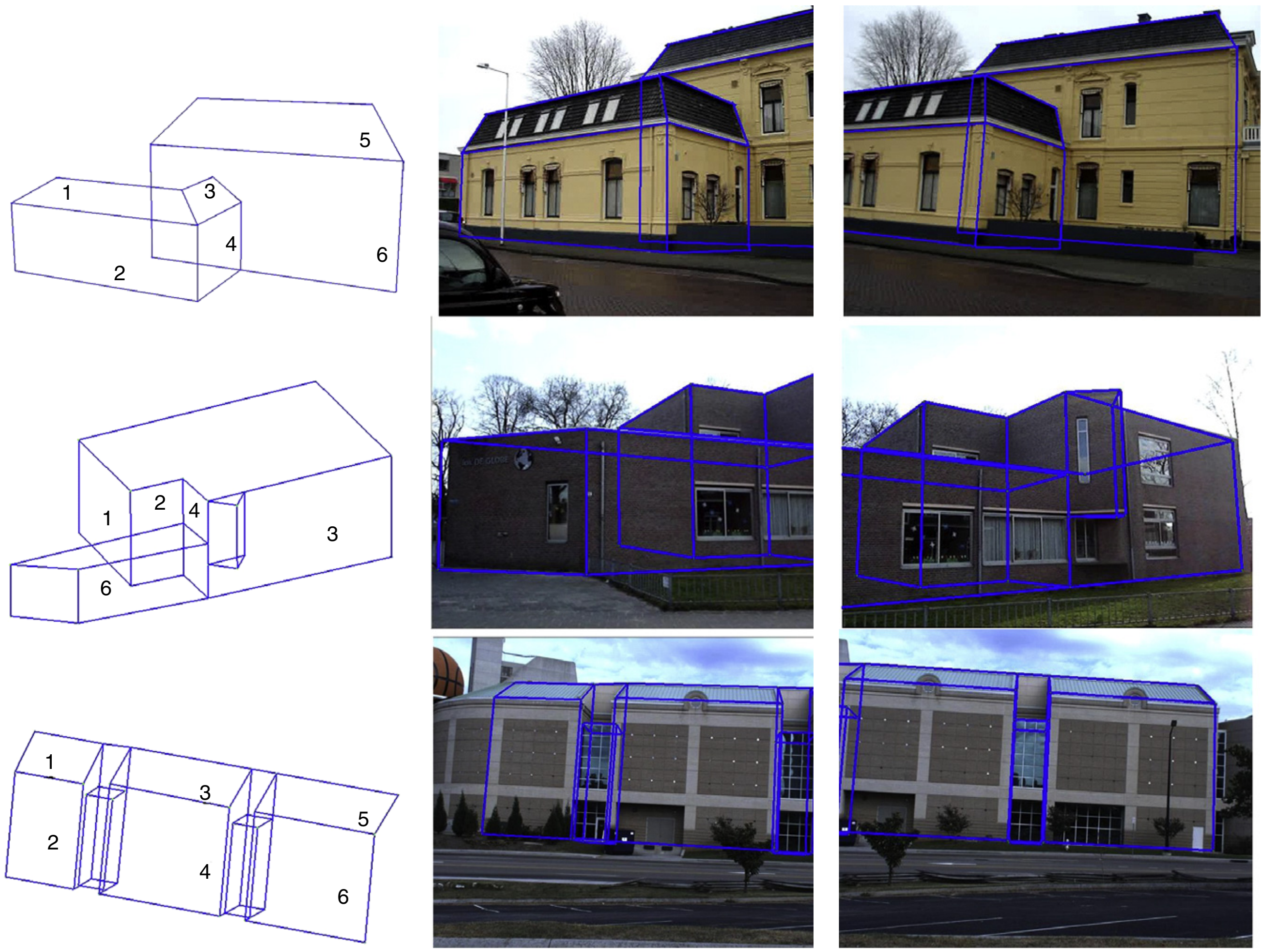
Building no.	2 (laser)	6 (laser)	2/6 (laser)	2 (result)	6 (result)	2/6 (result)
(1)	5.16	10.03	0.51	4.91	9.47	0.52
(2)	7.77	4.35	1.79	3.29	1.84	1.79

absolute value cannot be directly compared with measurements. However, the values and ratios in Table 3 show they have a similar height relation. For parameter settings, the approximate scale difference between the local coordinate system and the world coordinate system is considered. As mentioned in Section 2.4, such a difference can be removed by some reference distance measures. For building in Fig. 16 and building (1) in Fig. 18, some estimations from general knowledge, e.g. usual window height, are used. For building (2), the scale seems much smaller (the absolute values in Table 3 confirm it), so thresholds related to distance are reduced to half of the default setting.

For the stadium, we do not have reference data. However, from images in Fig. 18 the intersecting edges between surface patch 3 and 4, 5 and 6 seem to have a similar length. In our coordinate system, the length ratio of these two edges is 1.01, which is a quite reasonable difference. For this case, the image quality is much better than in the above cases. The point density is also higher, so we selected a smaller value (0.1 m) for the maximum distance between point and surface patch. Other parameters are still default values.

Figs. 20 and 21 show reconstruction results on a street. For these tests, one person sat in the back seat of a car holding the above mentioned camera that was mounted on a tripod. The speed of the car was around 15 km/h and the viewing angle with respect to driving direction was between 45°–90°. Car trajectories are shown in Pictometry oblique images. The buildings to be reconstructed are on the right side of streets. The image sequence used for the test in Fig. 20 has 892 frames. This street is in a residential area, therefore only a few cars and people are visible during work time. With default thresholds, the surface patch generation result was quite good. Only the roof of the first building was not successfully reconstructed because it was not visible in the image sequence. The sizes of these four buildings seem the same as shown in the image. The length and height of front façades of these buildings are measured based on a local coordinate system and listed in Table 4.

The image sequence used for the test in Fig. 21 has 1268 frames. Some occlusions, such as brushwood, trees and cars, are in front of the buildings. And with a slow speed, the car had to be on the right lane, which resulted in a short distance between the camera and building façades. The poor feature extraction result



(a) Reconstructed 3d model.

(b) Projection on first frame.

(c) Projection on last frame.

Fig. 19. Reconstruction result of the building façade (1) Yellow House (2) Brown House (3) Stadium. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

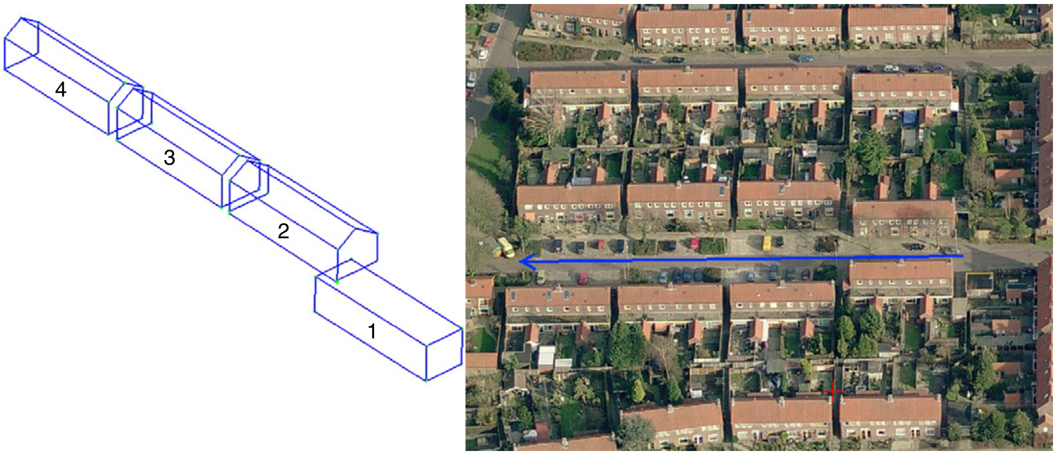


Fig. 20. Reconstruction results of building façades on the street (1) (images: (c) Blom).

Table 4
Length and height (m) of building front façades in Fig. 20.

	Façade no.			
	1	2	3	4
Length	26.08	26.01	24.98	24.53
Height	6.68	6.22	6.18	6.17

cannot be avoided because we only captured images from one direction. Larger distance thresholds, which reduce the possibility of separating different objects, were chosen. Because of these weaknesses above, there are some mistakes in the result. The façades of the two nearby buildings on right side of the image are merged together. The other four buildings can be identified

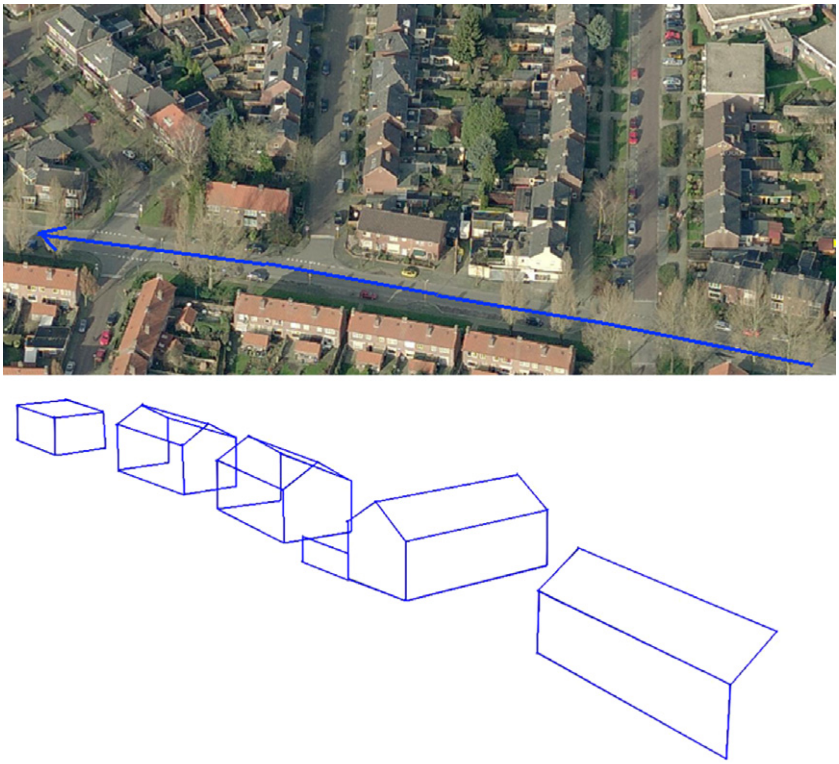


Fig. 21. Reconstruction results of building façades on the street (2) (images: (c) Blom).

separately based on large gaps between them. Parts of the third building façades and roof of the last building are lost.

6. Conclusions

In this paper, an approach for a building model's reconstruction from a video image sequence was presented. The main attention is to extract main surface patches, set up topological relationships between them, and then to recover the building's shape structure. Under the guidance of our building structure knowledge, the rules and processing steps are set up to reasonably group extracted sparse 3D points and 3D edges. Then the surface patch outline helps to restrict planes in the region corresponding to the actual case and makes it easier to connect adjacent surface patches later. Only the geometric information conveyed by the 3D points and 3D edges has been used for surface patch generation. Image information, especially on the intensity and texture, is used in the verification step when there are new surface patch hypotheses in areas where few features were extracted. A hybrid data-/model- driven strategy is used to connect neighborhood surface patches and to reconstruct the building model.

The results show that our method correctly sets up topological relationships between generated surface patches and also gets reasonable structures in areas with occlusions. The integration of knowledge does not complicate the processing but rather it simplifies the reconstruction process. Our method reconstructs building models from features first, so it does not restrict us to basic structures. We can also identify complex buildings when there are observations that contradict the preferential knowledge. On the other hand, our method also depends on the quality of feature extraction results, which affects the threshold settings. For cameras with a normal viewing angle, the distance between camera and buildings has to be large enough to let façades and at least part of their boundary be visible.

Our building models are required to be consistent with the modeled knowledge, but some buildings may not fully satisfy such constraints. For building models that are reconstructed, they are presented in a generalized way. Therefore the location of some intersecting edges does not correspond to actual boundary edges. Due to the way of capturing video, some areas are only visible in the first and last few frames and fewer reliable features are extracted from these areas based on the short baseline. This also affects surface patch boundary extraction. Accordingly the accuracy for surface patches and outline needs to be improved in the further work. The geometrical and topological model information as well as the theoretically available accuracy will be integrated into further steps as internal constraints and to trigger the thresholds during the model's reconstruction.

Acknowledgements

Parts of the image data (Stadium, cf. Fig. 18:3) was made available by Matthias Heinrichs, Technical University of Berlin, Germany. We gratefully acknowledge his support. The work described in this paper is supported by the National Basic Research Program of China (973 Program, NO. 2010CB731800), the National Natural Science Foundation of China (40871212 and 40721001), and the National High Technology Research and Development Program of China (2008AA121600). We also thank the anonymous reviewers for their comments.

References

Autodesk, 2010. Homepage ImageModeler software. http://usa.autodesk.com/adsk/servlet/index?siteID=123112\&id=11390028 (accessed 10.04.10).

Baillard, C., Zisserman, A., (1999). Automatic reconstruction of piecewise planar models from multiple views. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR'99, 2, pp. 559–565.

Baltsavias, E.P., 2004. Object extraction and revision by image analysis using existing geodata and knowledge: current status and steps towards operational systems. ISPRS Journal of Photogrammetry and Remote Sensing 58 (3–4), 129–151.

Brenner, C., 2005. Building reconstruction from images and laser scanning. International Journal of Applied Earth Observation and Geoinformation 6 (3–4), 187–198.

Cornelis, N., Leibe, B., Cornelis, K., Van Gool, L., 2008. 3D urban scene modeling integrating recognition and reconstruction. International Journal of Computer Vision 78 (2), 121–141.

Debevec, P., Taylor, C., Malik, J., 1996. Modeling and rendering architecture from photographs: a hybrid geometry- and image-based approach. In: SIGGRAPH'96, pp. 11–20.

Dick, A., Torr, P., Cipolla, R., 2004. Modelling and interpretation of architecture from several images. International Journal of Computer Vision 60 (2), 111–134.

Dobbert, T., 2005. Matchmoving: The Invisible Art of Camera Tracking. Sybex Inc.

Egenhofer, M.J., Franzosa, R.D., 1991. Point-set topological spatial relations. International Journal of Geographical Information Systems 5 (2), 161–174.

Egenhofer, M.J., Herring, J.R., 1990. A mathematical framework for the definition of topological relationships. In: The Fourth International Symposium on Spatial Data Handling, Zürich, Switzerland, pp. 803–813.

Ellul, C., Haklay, M., 2006. Requirements for topology in 3D GIS. Transactions in GIS 10 (2), 157–175.

Eos Systems, 2010. Homepage PhotoModeler software. http://www.photomodeler.com (accessed 10.04.10).

Hartley, R.I., Zisserman, A., 2004. Multiple View Geometry in Computer Vision, second ed. Cambridge University Press.

Henricsson, O., Baltsavias, E., 1997. 3-D building reconstruction with ARUBA: a qualitative and quantitative evaluation. In: Gruen, A., Baltsavias, E., Henricsson, O. (Eds.), Automatic Extraction of Man-Made Objects from Aerial and Space Images (II). Birkhauser Verlag, pp. 65–76.

Lucas, B.D., Kanade, T., 1981. An iterative registration technique with an application to stereo vision. International Joint Conference on Artificial Intelligence 674–679.

Mayer, H., Reznik, S., 2007. Building facade interpretation from uncalibrated wide-baseline image sequences. ISPRS Journal of Photogrammetry and Remote Sensing 61 (6), 371–380.

Nistér, D., 2004. An efficient solution to the five-point relative pose problem. IEEE Transactions on Pattern Analysis and Machine Intelligence 26 (6), 756–770.

Open Geospatial Consortium, 2008. OpenGIS city geography markup language (CityGML) encoding standard. OGC 08-007r1, Version 1.0.0.

Pigot, S., 1991. Topological models for 3D spatial information systems. In: 10th International Symposium on Computer Assisted Cartography, Baltimore, Maryland, USA, pp. 369–391.

Pilouk, M., 1996. Integrated Modelling for 3D GIS. Ph.D. Thesis, International Institute for Aerospace Survey and Earth Sciences (ITC), Enschede.

Pollefeys, M., Nistér, D., Frahm, J.-M., Akbarzadeh, A., Mordohai, P., Clipp, B., Engels, C., Gallup, D., Kim, S.-J., Merrell, P., Salmi, C., Sinha, S., Talton, B., Wang, L., Yang, Q., Stewénius, H., Yang, R., Welch, G., Towles, H., 2008. Detailed real-time urban 3D reconstruction from video. International Journal of Computer Vision 78 (2), 143–167.

Pu, S., Vosselman, G., 2009. Knowledge based reconstruction of building models from terrestrial laser scanning data. ISPRS Journal of Photogrammetry and Remote Sensing 64 (6), 575–584.

Remondino, F., EL-Hakim, S., 2006. Image-based 3D modeling: a review. The Photogrammetric Record 21 (115), 269–291.

Roman, A., Garg, G., Levoy, M., 2004. Interactive design of multi-perspective images for visualizing urban landscapes. IEEE visualization 537–544.

Schindler, G., Dellaert, F., Sing Bing, K., 2007. Inferring temporal order of images from 3D structure. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR'07, pp. 1–7.

Snavely, N., Seitz, S., Szeliski, R., 2008. Modeling the world from internet photo collections. International Journal of Computer Vision 80 (2), 189–210.

Suveg, I., Vosselman, G., 2004. Reconstruction of 3D building models from aerial images and maps. ISPRS Journal of Photogrammetry and Remote Sensing 58 (3–4), 202–224.

Tian, Y., Gerke, M., Vosselman, G., Zhu, Q., 2008. Automatic edge matching across an image sequence based on reliable points. International Archives of the Photogrammetry, Remote Sensing and Spatial Information Science 37 (Part 3B), 657–662.

Tian, Y., Gerke, M., Vosselman, G., Zhu, Q., 2009a. Automatic Surface Patch Generation from a Video Image Sequence, 3D Geo-Information Sciences: Requirements, Acquisition, Modeling, Analysis, Visualization. Springer, Berlin, pp. 235–246.

Tian, Y., Zhu, Q., Gerke, M., Vosselman, G., 2009b. Knowledge-based topological reconstruction for building facade surface patches. International Archives of the Photogrammetry Remote Sensing and Spatial Information Science 38 (Part 5/W1), 6 p (on CD-ROM).

Vicon, 2010. Homepage Boujou software. http://www.vicon.com/boujou/ (accessed 10.04.10).

Vosselman, G., Gorte, B.G.H., Sithole, G., Rabbani, T., 2004. Recognising structure in laser scanner point clouds. International Archives of Photogrammetry Remote Sensing and Spatial Information Sciences 36 (Part 8/W2), 33–38.

Werner, T., Zisserman, A., 2002. New techniques for automated architectural reconstruction from photographs. In: Seventh European Conference on Computer Vision, pp. 541–555.