Modeling urban landscapes from point clouds: a generic approach

Florent Lafarge, Clément Mallet

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

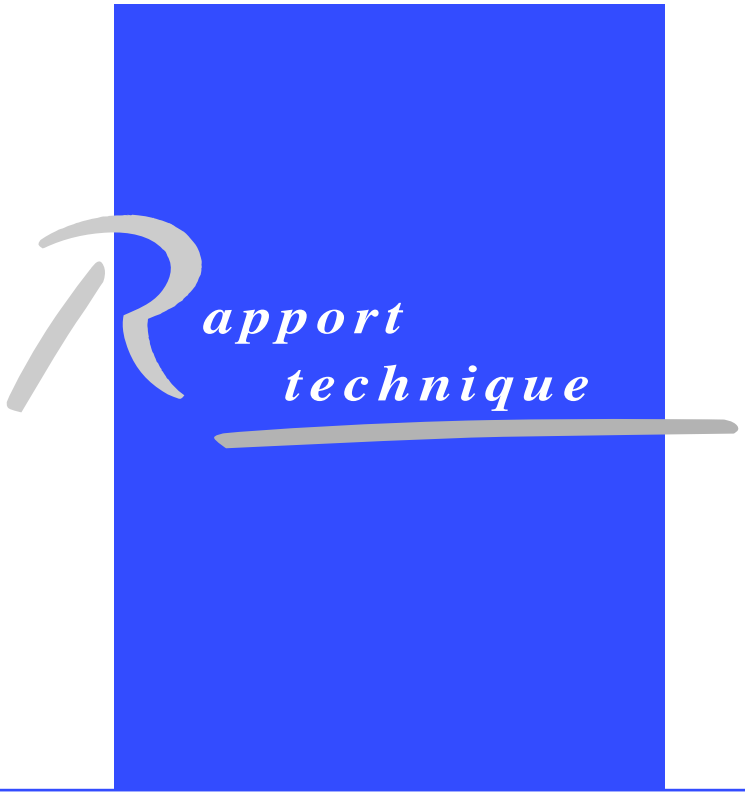
Modeling urban landscapes from point clouds: a generic approach

Florent Lafarge — Clément Mallet

N° 7612

May 2011

Vision, Perception and Multimedia Understanding



ISSN 0249-0803 ISRN INRIA/RT--7612--FR+ENG

INSTITUT NATIONAL
DE RECHERCHE
EN INFORMATIQUE
ET EN AUTOMATIQUE

INRIA

centre de recherche
SOPHIA ANTIPOLIS - MÉDITERRANÉE

Modeling urban landscapes from point clouds: a generic approach

Florent Lafarge *, Clément Mallet†

Theme : Vision, Perception and Multimedia Understanding
Perception, Cognition, Interaction
Équipes-Projets Ariana

Rapport technique n° 7612 — May 2011 — 34 pages

Abstract: We present a robust method for modeling cities from 3D-point data. Our algorithm provides a more complete description than existing approaches by reconstructing simultaneously buildings, trees and topographically complex grounds. A major contribution of our work is the original way of modeling buildings which guarantees a high generalization level while having semantized and compact representations. Geometric 3D-primitives such as planes, cylinders, spheres or cones describe regular roof sections, and are combined with mesh-patches that represent irregular roof components. The various urban components interact through a non-convex energy minimization problem in which they are propagated under arrangement constraints over a planimetric map. Our approach is experimentally validated on complex buildings and large urban scenes of millions of points and compare it to state-of-the-art methods.

Key-words: Computer vision, 3D-geometry, shape representation, urban scenes, point data, energy minimization, Markov Random Fields

* INRIA Sophia Antipolis Méditerranée
† IGN

Centre de recherche INRIA Sophia Antipolis – Méditerranée
2004, route des Lucioles, BP 93, 06902 Sophia Antipolis Cedex
Téléphone : +33 4 92 38 77 77 — Télécopie : +33 4 92 38 77 65

Modéliser des paysages urbains à partir de nuages de points : une approche générique

Résumé : Nous présentons une méthode robuste pour modéliser les villes à partir de nuages de points 3D. Notre algorithme fournit une description plus complète que les approches existantes en reconstruisant simultanément bâtiments, arbres et sols topographiquement complexes. Une des contributions importantes réside dans la manière originale de modéliser en 3D les bâtiments, garantissant un niveau de généralisation élevé tout en ayant une représentation compacte et sémantisée. Des primitive géométriques 3D telles que des plans, des cylindres, des sphères ou des cones décrivent les facettes de toit régulières. Elles sont combinées avec des parties de maillages qui représentent les composants de toits irréguliers. Les différents éléments urbains intéragissent au sein d'un problème de minimisation d'énergie non convexe dans lequel ils sont propagés sous des contraintes d'arrangement sur une carte planimétrique. L'approche est validée expérimentalement sur des bâtiments complexes et sur des scènes à grandes échelles contenant des millions de points, et comparée à des méthodes références.

Mots-clés : Vision par ordinateur, géométrie 3D, représentation de formes, scènes urbaines, nuage de points, minimisation d'énergie, champs de markov

Contents

1 Introduction

1.1 Problem statement

The 3D-modeling of urban scenes is a topic of major interest in computer vision. Driven by new virtual applications, this research domain has considerably progressed during the last decade as underlined in recent studies [43, 25, 16]. A part of the existing methods is devoted to street level modeling from ground or oblique-view data. These works propose accurate facade 3D-models which are particularly useful for the ground based navigation systems. They can be obtained from various types of data such as multi-view images [10, 32, 13, 31, 37, 35], laser scans [3, 8, 12] or video [28]. Other works propose large city descriptions from airborne data, and offer complementary advantages to the street level representations, in particular fine roof descriptions. These works are crucial for a large range of applications, from virtual globe visits to urban planning through to video games. We focus here on large-scale city modeling problems from aerial data, in particular from point set data generated by airborne acquisition systems.

1.2 Related works from point set data

Most of the existing city modeling approaches directly or indirectly tackle the problem through point cloud analysis.

Digital Surface Models (DSM), which are 2.5D view-dependent representations, constitute structured point clouds having a regular point distribution in the XY-plane well adapted to aerial-based city modeling. Zebedin *et al.* [41] and Lafarge *et al.* [18] generate DSMs from MultiView Stereo (MVS) imagery in order to model buildings by polyhedral structures. The latter use a Constructive Solid Geometry (CSG) based approach by reconstructing a building as an assembling of parametric 3D-blocks, the former propose to partition a building in small 2D-polygons which are then labeled by graph-cut optimization.

Other approaches consider unstructured point clouds directly generated from Laser/Lidar systems [36, 34, 24, 29, 33, 42] or MVS imagery [7]. Such data have spatially heterogeneous point distributions without induced neighborhood relationships between the points, and contain outliers, especially when generated from MVS imagery. Vosselman *et al.* [36] present a semi-automatic approach using an interactive segmentation of the parcel boundaries on which are fitted flat, gable, or hip roofs. Matei *et al.* [24] and Poullis *et al.*[29] propose flat roof models adapted to *Manhattan World* environments [9]. Both approaches focus on segmenting the buildings and simplifying their boundaries, either by estimating building orientations [24] or by using statistical considerations [29]. [34] identify some building components from a Delaunay triangulation of the point data which are then combined to model simple roof structures. A more general building representation is proposed by Zhou *et al.* [42] who use a mesh simplification procedure based on dual contouring. Although this approach wins in terms of generalization, semantic information is lost: a simple planar roof section can be described by many mesh facets with different normal orientations.

1.3 Motivations

These approaches provide convincing 3D-models but have some important limitations. Firstly, strong urban prior on orthogonality, symmetry and roof typology are frequently introduced to reduce the solution space, and thus the problem complexity. These assumptions are usually efficient for *Manhattan World* environments but become penalizing for less well-organized urban landscapes having high variations of roof structures such as the areas presented in Section 6. Secondly, these methods provide a sparse description of urban scenes. They are focused on the building modeling task and disregard all the other objects which can be found in an urban scene such as trees, or even sometimes ground surfaces by assuming a constant altitude over the global scene. Thirdly, these models are each designed for a specific type of input data, and the resulting quality generally falls down when modifying data specifications. For instance, the mesh simplification algorithm proposed by Zhou *et al.* [42] is of limited interest with point clouds of low densities, as well as the CSG-based approach of Lafarge *et al.* [18] with unstructured point sets generated from laser or MVS.

1.4 Contributions

We propose an algorithm which brings solutions to address the problems mentioned above. Our method presents several significant contributions to the field.

• *More complete models of unspecified urban environments:* we do not simply reconstruct the buildings: a more complete representation is provided by modeling vegetation and topologically complex grounds. Moreover, our method is adapted to various types of urban landscapes, from financial districts of big cities to small mountainous villages, including historical towns with old buildings of architectural interest. Besides, it is robust on a large range of point data having different point densities and various sensor characteristics.

• *Hybrid reconstruction of buildings:* the modeling of the buildings combines geometric 3D-primitives such as planes, cylinders, spheres or cones to represent standard roof sections and mesh-patches to describe more irregular roof components. Thus, 3D-models provide urban details while being semantized and compact. These two different types of 3D-representation tools interact through a non-convex energy minimization problem. This idea has been originally proposed in former works [19] in order to reconstruct facades from MVS images and has revealed a high potential.

• *2.5D-arrangement scheme for the urban structures:* a general formulation for the roof section arrangement problems is presented, the first to date to our knowledge which works in non-restricted contexts, *i.e.* with *(i)* unspecified primitives, *(ii)* various types of urban objects interacting in the scene, and *(iii)* unknown building contours. This 2.5D-arrangement scheme allows the combination of parametric 3D-shapes as well as unspecified urban components in a planimetric label map while imposing structural constraints.

A four-step strategy, illustrated in Fig. 1, is adopted. First, the point cloud is classified using an unsupervised method presented in Section 2. Four classes

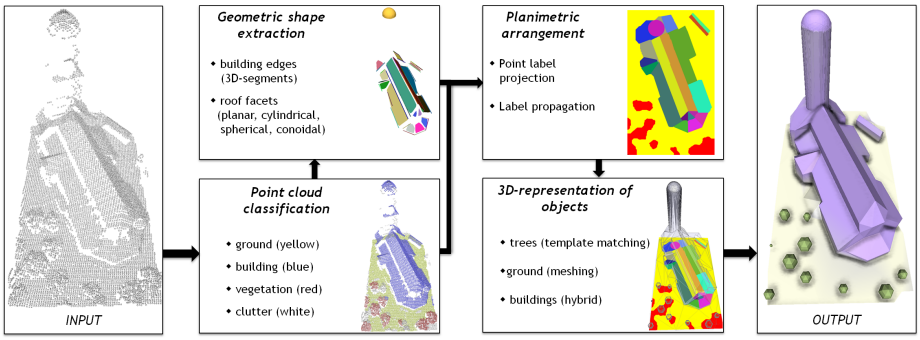


Figure 1: Overview of the proposed approach - Our algorithm digests large amounts of 3D-points in order to provide a compact and semantized representation of urban environments including atypical buildings, trees, and topologically complex grounds.

are distinguished: *ground*, *building*, *vegetation* and *clutter*. The second step, detailed in Section 3, consists in extracting geometric primitives such as 3D-segments, planes or cylinders from the point set classified as *building* by a fast process. Section 4 constitutes the key element of the system in which the geometric primitives and the other urban components are arranged in a common planimetric map through a multi-label energy minimization formulation. In the last stage, the various urban objects are represented in 3D using template fitting and meshing procedures explained in Section 5. Experimental results on complex urban structures and various types of large urban scenes are presented and commented in Section 6, as well as a comparison from Laser-based and MVS-based input data.

2 Point cloud classification

Four classes of interest are defined: *building*, *vegetation*, *ground* and *clutter*. The class *vegetation* represents the trees which have a non negligible size at the city scale, *i.e.* with a height of several meters, excluding the shrubs. The class *clutter* corresponds to the outliers contained in the data and to small urban components which temporarily perturb the scene such as cars, fences, wires, roof antennas or cranes. This class also includes the vertical structures such as facades because these have a sparse and irregular point repartition penalizing the scene understanding. A neighboring relationship is defined to create spatial dependencies between the 3D-points. Two points are neighbors if their Euclidean distance is inferior to a certain value, in practice 2 m (spherical neighborhood).

2.1 Discriminative features

For each point, several geometric attributes are computed in order to distinguish the four classes of interest.

- *Local non-planarity* f_p represents the quadratic distance between the point

and the optimal 3D-plane computed among its neighbors. Low values correspond to buildings and ground.

• *Elevation* f_e allows the distinction between the ground and the other classes. This feature corresponds to the height difference between the point and its planimetric projection on an elevation map of the ground estimated by a standard algorithm [5].

• *Scatter* f_s measures the local height dispersion of the points. It provides a high value in the case of trees and also some undesirable urban components. This feature is usually defined as the minimal principal curvature mean of the considered point and its neighbors [33]. In the case of point sets generated from full waveform topographic Lidar systems, an alternative way to compute the scatter attribute f_s is considered using the echo number information [22]. The feature f_s is then defined as the ratio between the number of neighbors whose echo number is strictly superior to 1 and the total number of neighbors. This alternative allows the improvement of the feature accuracy (see Section 6).

• *Regular grouping* f_g is dedicated to outliers and undesirable urban components having a linear structure such as wires, facade parts, cranes or fences. This feature corresponds to the quadratic distance between the considered point and the optimal 3D-line computed among its neighbors, weighted by the number of neighbors. The response is low in the case of small isolated sets of points and linear layouts of points.

In order to tune the sensitivity of each feature, four parameters σ_e, σ_p, σ_s and σ_g are introduced. The features are then normalized by a linear projection on the interval $[0, 1]$. Fig. 2 shows the behavior of these features on a small area, and underlines their complementarity in order to discriminate our four classes of interest. For example, the building roofs can be distinguished from the other urban elements as the areas having a high response to the elevation feature f_e while having low responses to the scatter and local non-planarity features, f_s and f_p.

2.2 Non-supervised classification formulation

An energy minimization is proposed to classify the point set. Let $x = (x_i)_{i=1..N_c}$ be a potential classification result with N_c the number of points of the cloud, and $x_i \in \{$*building*, *vegetation*, *ground*, *clutter*$\}$ the class of the i^{th} point. The energy $E(x)$ is defined as a sum of partial data terms $E_{di}(x_i)$ and pairwise interactions defined by the standard Potts model [21] which introduces spatial coherence between neighboring elements:

$$E(x) = \sum_{i=1..N_c} E_{di}(x_i) + \gamma \sum_{i \sim j} \mathbb{1}_{\{x_i \neq x_j\}} \tag{1}$$

where $\gamma > 0$ is the parameter of the Potts model, $i \sim j$ represents the pairs of neighboring points, and $\mathbb{1}_{\{.\}}$, the characteristic function. The partial data term $E_{di}(x_i)$ measures the coherence of the class x_i at the i^{th} point. It is defined as

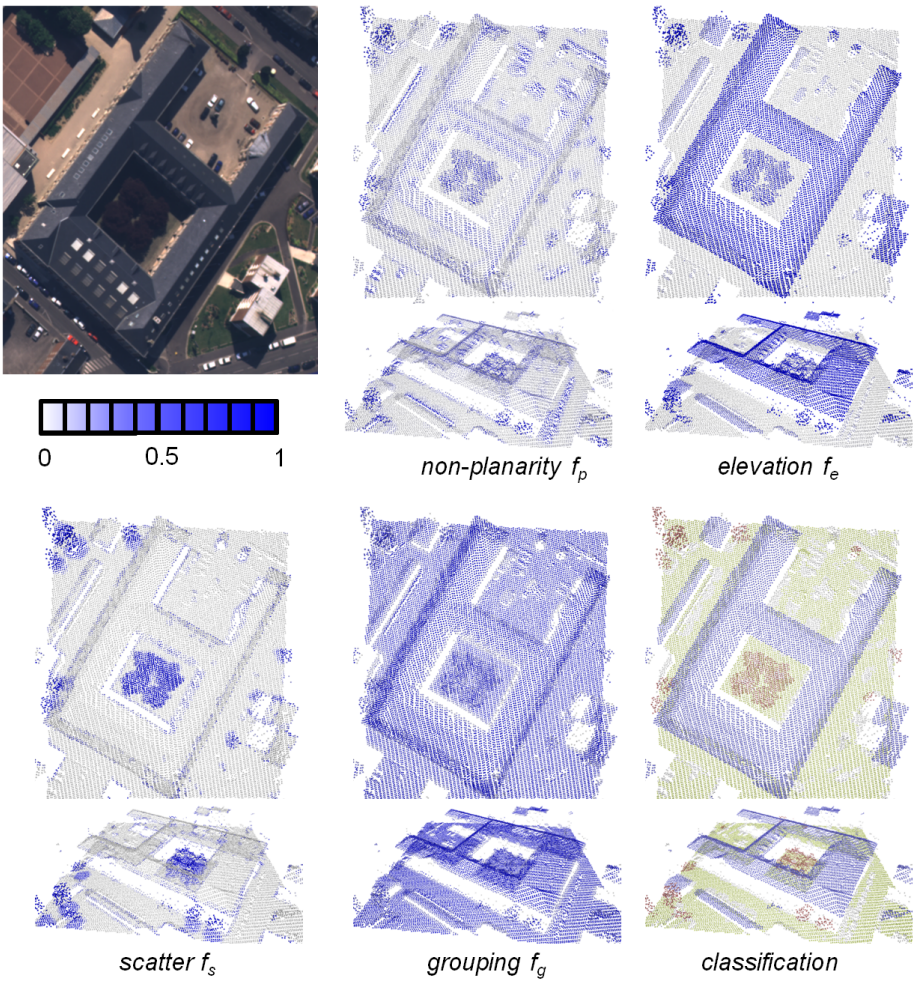


Figure 2: Behavior of the discriminative features- *(from top to bottom, left to right)* an aerial picture of the scene (not used) containing a building and its surrounding area, input clouds with the points colored according to the response of the features [color code: white=low response, blue=high response], and the classification result [color code: blue=*building*, red=*vegetation*, yellow= *ground* and white= *clutter*]. Each feature brings a specific type of information such that the combinations of the features allow the distinction of the different classes in the input point cloud. In particular, note how the points corresponding to trees and facades are correctly labeled as *vegetation* and *clutter* respectively.

a combination of the normalized features defined above given by

$$E_{di}(x_i) = \begin{cases} (1-f_e).f_p.f_s & \text{if } x_i = \textit{building} \\ (1-f_e).(1-f_p).(1-f_s) & \text{if } x_i = \textit{vegetation} \\ f_e.f_p.f_s & \text{if } x_i = \textit{ground} \\ (1-f_p).f_s.f_g & \text{if } x_i = \textit{clutter} \end{cases} \quad (2)$$

A Graph-Cut based algorithm [4] is used to quickly reach an approximate solution close to the global optimum of our energy. One can easily check that our model fits the requirements for this algorithm. In our experiments, the initial configuration is chosen as the configuration minimizing the partial data terms. The energy has five parameters: γ, σ_e, σ_p, σ_s and σ_g. The parameter γ which balances the Potts interaction with respect to the partial data terms, is set to $(2.\widehat{p})^{-1}$ where $\widehat{p}$ is the average point density of the dataset. σ_e is set to 6 m (*i.e.* the height of two floors), σ_s to 0.5, σ_p to 0.5 m, and σ_g to 0.25 m. One can imagine tuning these parameters using a learning procedure, as for example in the works of Golovinskiy *et al.* [15] or Munoz *et al.* [27]. However, we notice that these values are stable on a wide range of input data. Thus, this would unnecessarily make the system heavier.

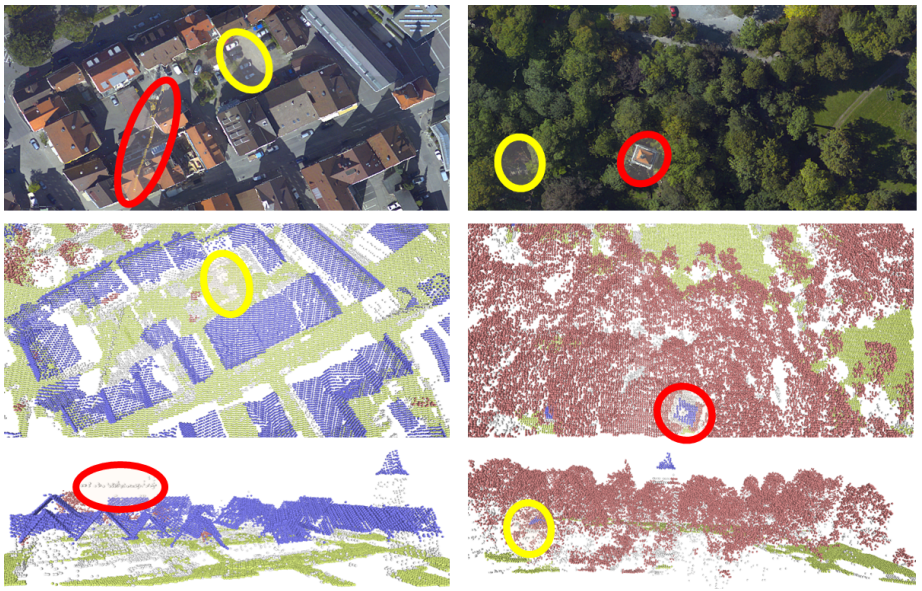


Figure 3: Point cloud classification on two small areas - (*top*) aerial pictures, and (*middle*) top and (*bottom*) profile views and of the classified cloud [color code: blue=*building*, red=*vegetation*, yellow= *ground* and white= *clutter*]. Note that the two towers in the middle of a dense vegetation are correctly detected as *building* despite their small size (*right*) and how the crane, the cars, the facades and the outliers are well classified as *clutter* (*left*).

2.3 Comments

The energy model has a relatively simple formulation and provides convincing results in practice. Fig. 3 shows the potential of the model on two difficult examples, in particular, with the retrieval of two thin towers in the middle of a dense wood. Note also that the eventual local errors do not necessarily have consequences on the final result. In fact, they can be corrected during the planimetric arrangement procedure detailed later in Section 4 by using urban structure layout considerations.

3 Geometric shape extraction

The second step consists in extracting 3D-primitives from the point set classified as *building*. As the classification proposed in Section 2 rejects outliers from the *building* point set, the use of Ransac-based algorithms, which are more computationally expensive for similar problems [30, 33], is not required. Two types of elements are detected: 3D-segments to locate the building contours, and surface primitives to identify the roof sections. In order to be fitted by a 3D-segment or a surface primitive, a subset of points has to verify the two following requirements:

• *Minimum quality of fitting:* the quadratic error ϵ between the set of the considered points and a 3D-segment (respectively a surface primitive) is required to be inferior to a reference error ϵ_s (resp. ϵ_a). The quadratic error ϵ between a subset of points $(p_k)_{k=1..K}$ and a manifold $\mathcal{M}$ is defined by

$$\epsilon = \sqrt{\frac{1}{K}\sum_{k=1}^{K} d(p_k, \mathcal{M})^2} \tag{3}$$

where $d(p_k, S)$ is the Euclidean distance from the point p_k to the manifold $\mathcal{M}$.

• *Minimum number of points:* for each primitive, a minimun number of matched points is imposed in order to guaranty robust fittings and to exclude non-significant small structures. The number of points fitted by a 3D-segment (respectively by a surface primitive) has to be superior to a certain parameter N_s (resp. N_a) whose value is fixed according to the input data characteristics (see Section 6).

3.1 3D-segments

Segments are used to locate the building contours. Our concern is not to describe the contour of a building as a set of perfectly connected segments (which is a difficult talk requiring urban assumptions and geometric approximations), but rather to have an accurate positioning of the main edges with potentially small parts missing between them (see Fig. 4). Indeed, our strategy consists in filling in the eventual missing parts further in Section 4 during the planimetric arrangement procedure.

First, the points located on the building borders are selected from the point set classified as *building*. The selection is performed by testing whether the

Euclidean distance of the considered point to the optimal 3D-line among its neighbors is inferior to a certain threshold which depends on the point density of the input data. In practice, the threshold is equal to $(2\sqrt{\widehat{p}})^{-1}$.

Then, 3D-lines are detected from the selected points by a clustering procedure. The process finds successive clusters of points whose quadratic error to the optimal 3D-line is inferior to ϵ_s and whose the number is superior to N_s. Note that the point aggregation is performed among the neighbors of the points already contained in the cluster. It allows us to detect a 3D-line formed by a compact set of points without holes. The 3D-segments are finally obtained by projecting the two extreme points of each cluster on the corresponding optimal 3D-line.

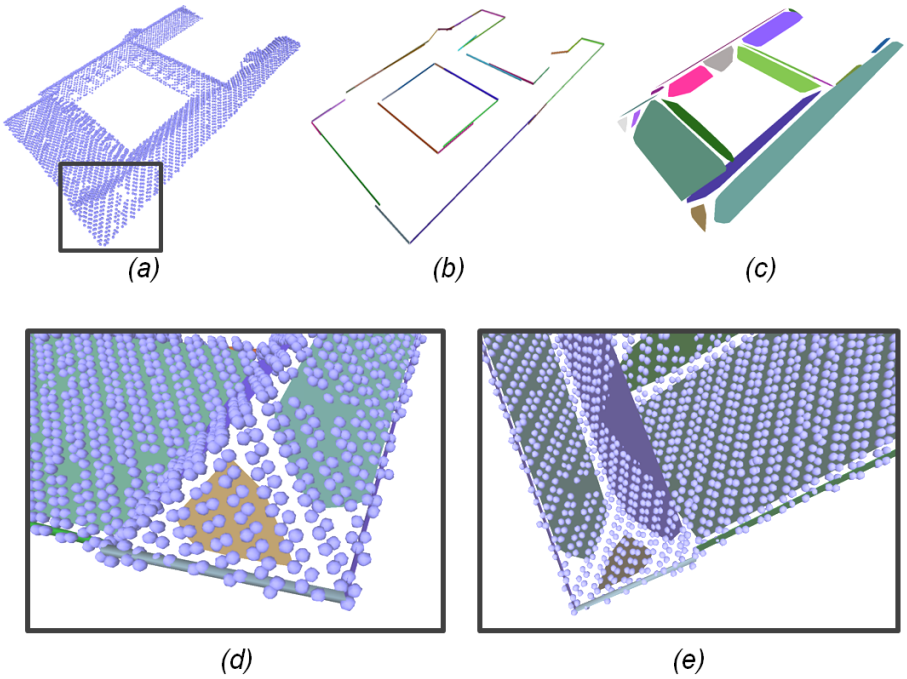


Figure 4: Shape extraction from the building of Fig. 2 - Both *(b)* 3D-segments and *(c)* surface primitives are extracted from *(a)* the set of points classified as *building*. The main regular roof sections of the buildings are detected as well as the global building contours. The cropped part with *(d)* top and *(e)* bottom views show the primitives in the middle of the point set. Note that the planes are visually represented by their convex envelopes.

3.2 Planar, spherical, cylindrical, and conoidal shapes

The surface primitives allow the detection and the recognition of the regular roof sections.

The planar structures, which constitute the most common shape of roofs, are extracted first. A region growing allows the fast detection of 3D-planes. The propagation criterion tests whether the direction of the normal of the considered point is similar to those of the points in the region. When the propagation stops, the optimal 3D-plane is computed from the points of the region. The plane is then selected as a primitive if both the number of points in the region is superior to N_p and the quadratic distance to the points of the region is inferior to ϵ_p. This procedure is iteratively performed on the unfitted points.

Non-planar shapes are then detected from the points which have not been fitted by a plane. Extracting spheres, cylinders or cones has no obvious solution when the points only represent an unknown portion of the whole shape. One can use Monte Carlo sampling but it requires high computing time [17]. We prefer extracting these non-planar primitives using an iterative non-linear minimization, typically by a Levenberg-Marquardt optimization. The parametrization and the first order Euclidean distance approximation to spheres, cylinders and cones proposed by Marshall *et al.* [23] are used to achieve numerically stable fittings. The extracted primitives are kept if the conditions on the minimal number of points per primitive and the maximum quadratic error are validated.

Extracting non-planar shapes subsequently to the 3D-planes avoids both high computing times and typical confusions between large non-planar primitives and planes which could have the same fitting error.

4 Planimetric arrangement

The third step represents the key part of the system. It consists in arranging both the geometric shapes extracted in Section 3 and the other urban components identified in Section 2 in a common dense representation.

Several efficient methods of roof section arrangement have been proposed in restricted contexts. A model for planar sections is presented by Baillard *et al.* [2] for simple houses. Revolution sections are also taken into account by Zebedin *et al.* [41], but this graph-cut based approach does not address the building contouring problem and requires building masks as input. It remains an open issue when *(i)* the primitives are unspecified, *(ii)* different types of urban objects interact in the scene, and *(iii)* the building contours are not given. We propose an original solution by propagating the point labels in a grid of X and Y axis under structure layout constraints (see Fig. 5). Performing the arrangement on such a grid, called a planimetric map in the following, allows us to substantially reduce the problem complexity by assuming a 2.5D representation of urban scenes, and also to combine two different types of 3D-geometry tools, *i.e.* primitives and mesh patches, in a common framework.

4.1 Point labels and 2D-grid

Each point of the cloud is associated with the label *ground*, *vegetation*, *clutter*, $plane^{(l)}$, $cylinder^{(m)}$, $sphere^{(n)}$, $cone^{(o)}$ or *roof*. The points labeled as *clutter* are not taken into account in the following. The label *roof* corresponds to the

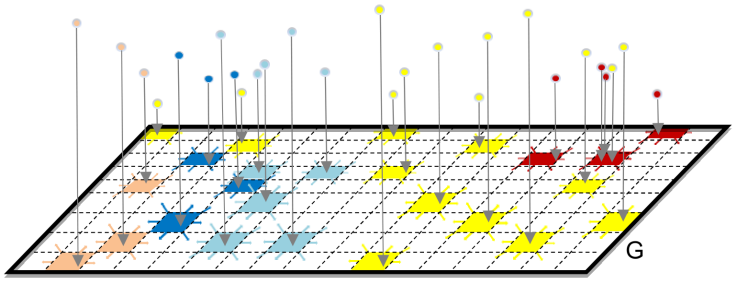


Figure 5: The labels of the 3D-points are first projected onto a 2D-grid G, and then propagated under arrangement constraints.

points classified as *building* in Section 2, which have not been fitted to planar, spherical, cylindrical or conoidal shapes.

The labels of the 3D-points are projected on a 2D-grid G as illustrated on Fig. 5. We denote by $G^{(proj)}$, the subset of G composed of the cells on which at least one point label has been projected, and $G^{(empty)}$, the complementary subset of $G^{(proj)}$ on G, *i.e.* the subset composed of the empty cells:

$$G = G^{(proj)} \cup G^{(empty)} \tag{4}$$

under the condition $G^{(proj)} \cap G^{(empty)} = \emptyset$.

Then, the projected labels are extended to the empty cells of $G^{(empty)}$ by a basic isotropic diffusion in order to have a dense labeling on the entire grid G, as illustrated in Fig. 8, second column. This first label map, denoted by $l^{(ini)}$, constitues the initial configuration of the propagation process under smoothness and structure arrangement constraints described in the next part.

4.2 Label propagation under geometric constraints

The label propagation procedure is performed using a Markov Random Field (MRF) with pairwise interactions, whose sites are specified by the cells of the 2D-grid G, and whose adjacency set E is given by a breakline-dependent neighborhood. $l = (l_i)_{i \in G} \in L$ represents a configuration of labels of the MRF, where L is the configuration space:

$$L = \{\text{ground}, \text{vegetation}, \text{plane}^{(l)}, \text{cylinder}^{(m)}, \text{sphere}^{(n)}, \text{cone}^{(o)}, \text{roof}\}^{card(G)} \tag{5}$$

The quality of a configuration l is measured by the energy U of the standard form:

$$U(l) = \sum_{i \in G} D_i(l_i) + \beta \sum_{\{i,j\} \in E} V_{ij}(l_i, l_j) \tag{6}$$

where D_i and V_{ij} constitute the data term and propagation constraints respectively, balanced by the parameter $\beta > 0$.

Breakline-dependent neighborhood - The neighborhood relationship is not defined by an isotropic area, but takes into account the 3D-segments extracted in Section 3 in order to stop the propagation beyond building contours. It is given by:

$$\{i,j\} \in E \Leftrightarrow \begin{cases} \|i-j\|_2 \leq r \\ \mathcal{O}(i,\mathcal{L}_k) = \mathcal{O}(j,\mathcal{L}_k) \end{cases} \tag{7}$$

where $\mathcal{L}_k$ is the 2D-line obtained by projecting the k^{th} 3D-segment interacting with the pair $\{i,j\}$ (see Fig. 6). $\mathcal{O}(i,\mathcal{L})$ is the oriented side in which the cell i is located with respect to the line $\mathcal{L}$, and r is the maximal distance between two neighboring cells. This breakline-dependent neighborhood allows us to efficiently address the building contouring problem, which is usually a critical point in existing methods.

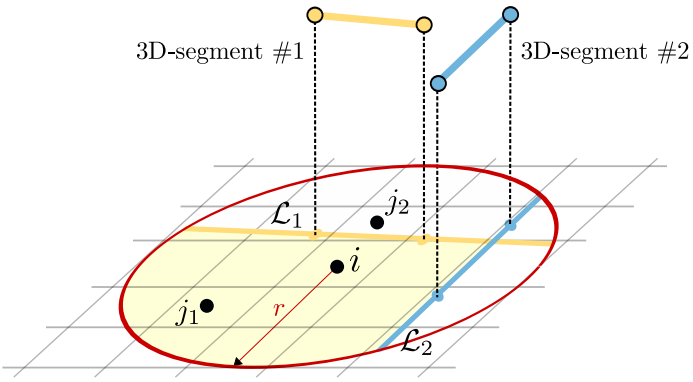


Figure 6: Breakline-dependent neighborhood- The neighbors of the cell i are contained in the yellow area. $\{i,j_1\} \in E$ but $\{i,j_2\} \notin E$. Note that the 3D-segments do not have to be connected as the yellow area is computed by intersecting the 2D-lines supporting the segments.

Data term - D_i checks the coherence of the label l_i at the cell i with respect to the input point cloud. The term is given by

$$D_i(l_i) = \begin{cases} c & \text{if } l_i = roof \\ \min(1, |z_{l_i} - z_{p_i}|) & \text{else if } i \in G^{(proj)} \\ 0 & \text{otherwise} \end{cases} \tag{8}$$

where $c \in [0,1]$ is a coefficient penalizing the labels $roof$ in order to favor the primitive-based description of buildings. z_{l_i} is the height associated with l_i, and z_{p_i} the maximal height of the input 3D-points contained in the cell i.

Propagation constraints - V_{ij} allows both the label smoothness and a coherent arrangement of the structures. To do so, an arrangement law, denoted by $\bowtie$, is introduced to test whether two labels, l_i and l_j, of neighboring cells, i and j, are spatially coherent:

$$l_i \bowtie l_j \Leftrightarrow \mathcal{O}(i,\mathcal{I}_{l_i,l_j}) \neq \mathcal{O}(j,\mathcal{I}_{l_i,l_j}) \tag{9}$$

where $\mathcal{I}_{l_i,l_j}$ is the XY-intersection between the two objects l_i and l_j, and $\mathcal{O}(i,\mathcal{I})$ is the oriented side in which the cell i is located with respect to the curve $\mathcal{I}$. In other words, the intersection of the two objects must be spatially located in between the two cells i and j.

For example, if two neighboring cells are associated with two different planar labels, the $\bowtie$-law will check that the projection in the 2D-grid of the 3D-line intersecting the two 3D-planes is located in between the two cells. Thus, the exact separation of two connected planes is constrained as illustrated in Fig. 7. Finally the pairwise interaction is formulated by:

$$V_{ij}(l_i,l_j) = \begin{cases} \epsilon_1 & \text{if } l_i \bowtie l_j \\ \epsilon_2 & \text{if } l_i = l_j \\ 1 & \text{otherwise} \end{cases} \tag{10}$$

where ϵ_1 and ϵ_2 are real values in $[0,1]$ with $\epsilon_1 < \epsilon_2$. They tune the label smoothness with respect to the coherent object arrangement considerations.

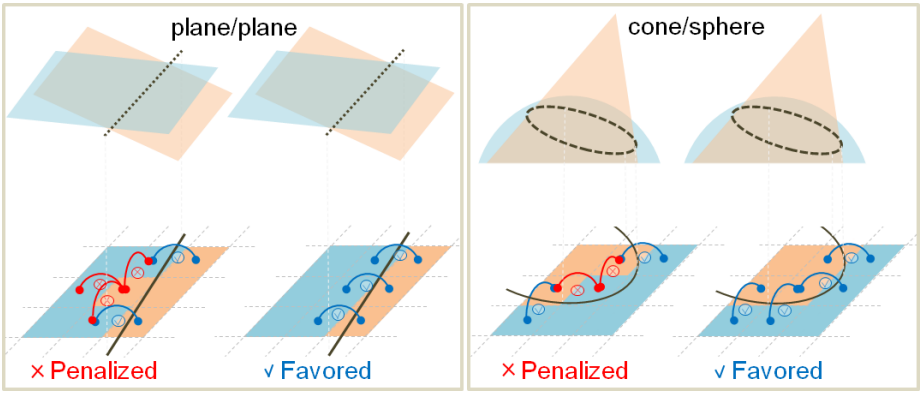


Figure 7: Principle of the $\bowtie$-law on two examples - The blue (respectively red) junctions between neighboring cells correspond to spatially coherent (resp. non-coherent) labels.

Optimization with parallelization scheme- Finding the label configuration which minimizes the energy U is a non-convex optimization problem. Simulated annealing techniques [21], graph-cut based algorithms *e.g.* [4] or belief propagation methods *e.g.* [38] could provide a good approximation of the solution but at the expense of high computing time. The scenes are generally of a large scale and the number of labels is very high.

In order to reach reasonable computing times, an original parallelization scheme is proposed, relying on the two following assumptions:

• H1: the labels cannot be propagated between two non-overlapped urban objects in the scene (*e.g* the label corresponding to the roof section of a building

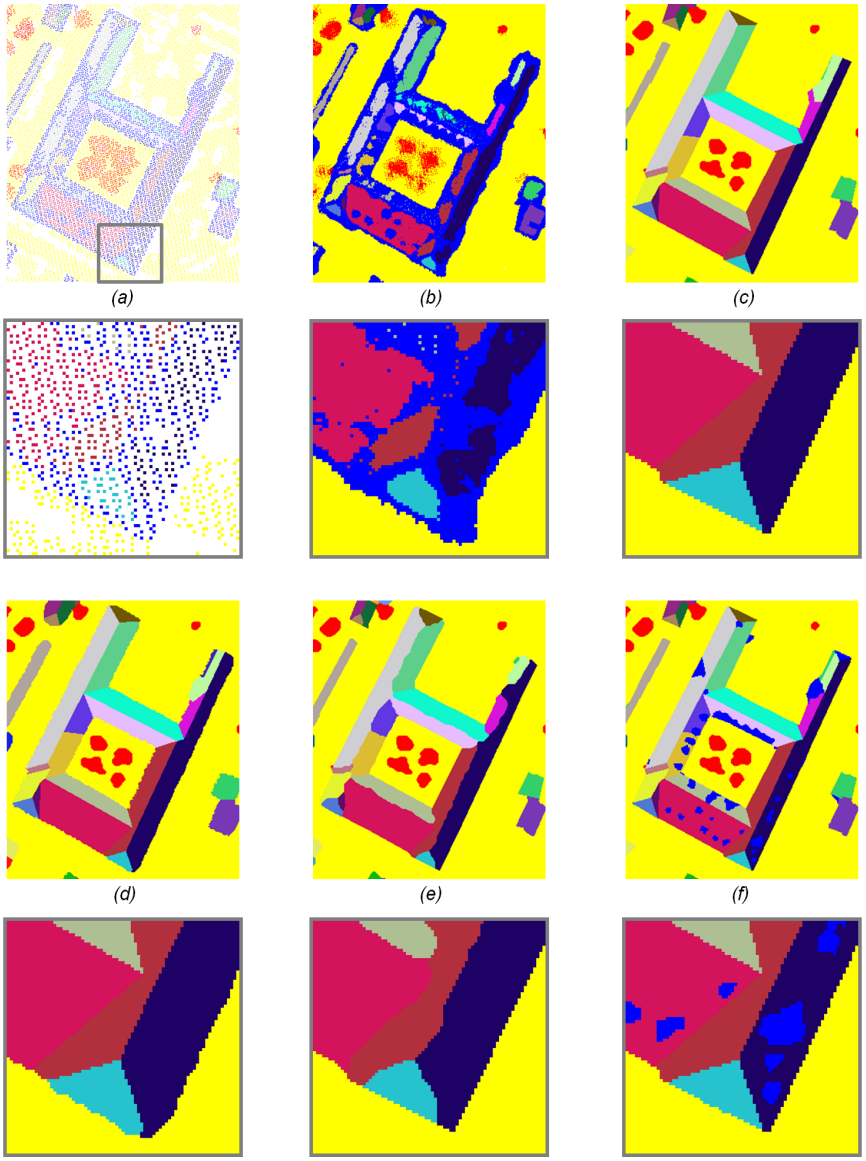


Figure 8: Planimetric arrangement from the building of Fig. 2 - *(a)* the grid $G^{(proj)}$ of the projected point labels, *(b)* the initial label map $l^{(ini)}$, *(c)* the label map after minimizing U, *(d)* the label map after minimizing a variant of U where the breakline-dependent neighborhood is substituted by a standard isotropic neighborhood, *(e)* the label map after minimizing a variant of U where the $\bowtie$-law is not taken into account, and *(f)* the label map after minimizing U whose parameter c has been significantly decreased. One can notice that the label propagation is correctly stopped beyond building contours and neighboring primitives. The $\bowtie$-law allows the optimal arrangement of the roof sections, and the breakline-dependent neighborhood avoids the wavy building contours. Note also how the decrease of the parameter c impacts on the apparition of *roof* labels in order to describe the small irregular roof components [color code: white=empty cell, yellow=ground, red=vegetation, blue=roof, other colors=primitives].

cannot be used for an other building),

• H2: the point labels originally projected in the grid $G^{(proj)}$ are of quality, *i.e.* they are probably correct (See Fig. 5).

The grid G is partitioned in an unknown number N of clusters such that

$$G = \underset{k\in[1,N]}{\cup} G_k \tag{11}$$

with $G_k \cap G_{k'} = \emptyset,\ \forall k,\ k' \in [1, N],\ k \neq k'$. The partition is obtained from the initial label map $l^{(ini)}$ by separating the low-level urban components (*e.g.* blocks of buildings, groups of trees, *etc*) which are supposed to be independent of each others (H1). The quality of the partition relies on the initial label map $l^{(ini)}$, and thus on the point labels originally projected in the 2D-grid (H2).

Each cluster G_k corresponds to a set of connected cells labeled as non-*ground* in the initial label map $l^{(ini)}$, and whose area is maximal. In particular, it implies that the outside contour of G_k, denoted by ∂G_k, is labeled as *ground*:

$$\forall i \in \partial G_k, k \in [1, N-1],\ l_i = \text{ground} \tag{12}$$

Note that a morphological erosion is preliminarily performed in the initial label map on the cells labeled as *ground* to give robustness to the component separation and avoid the omission of building pieces. As illustrated on Fig. 9, the last cluster G_N corresponds to the remaining cells labeled as *ground*. Fig. 15 also shows an example of a grid partitioning on a 1 km^2 dense urban area.

The original configuration space L (see Eq. 5) can be then significantly reduced by decomposing the minimization of U as a set of $N-1$ local independent (and thus parallelizable) energy minimization problems over the partition of the grid G:

$$\min_{l\in L} U(l) \Leftrightarrow \begin{cases} \min\limits_{l_{/G_k}\in L_k} U(l_{/G_k}) \ , \ \forall k \in [1, N-1] \\ l_{/G_N} = \{\text{ground}\}^{card(G_N)} \end{cases} \tag{13}$$

where $l_{/G_k}$ is a configuration of labels on the cluster G_k, and L_k the local configuration space on the cluster G_k. In order to limit the number of possible labels per local problem, L_k only contains the labels present in $l^{(ini)}_{/G_k}$:

$$L_k = \{l_i / l_i \in l^{(ini)}_{/G_k}\}^{card(G_k)} \tag{14}$$

Thus the label of a primitive belonging to a certain cluster is not uselessly tested in an other cluster (H1). This decomposition scheme has also an other advantage: the last cluster G_N of the remaining cells labeled as *ground*, which is usually of big size, is not concerned by the optimization. We rely here on the hypothesis H2 which allows a significant gain of time.

The α-expansion algorithm [4] is used to solve each local independent optimization problem. This algorithm is particularly efficient in our context, *i.e.* with a limited number of labels and a good initial configuration. Confidence

is given to the labels originally projected: the expansions are first performed on the subset $G^{(empty)}$, *i.e.* the cells originally considered as empty, and then on the complementary subset $G^{(proj)}$ to readjust the configuration. The parallelization scheme allows us to reach a good approximation of the solution while significantly reducing the computing times on a 8-core computer compared to standard techniques as shown in Tab. 1.

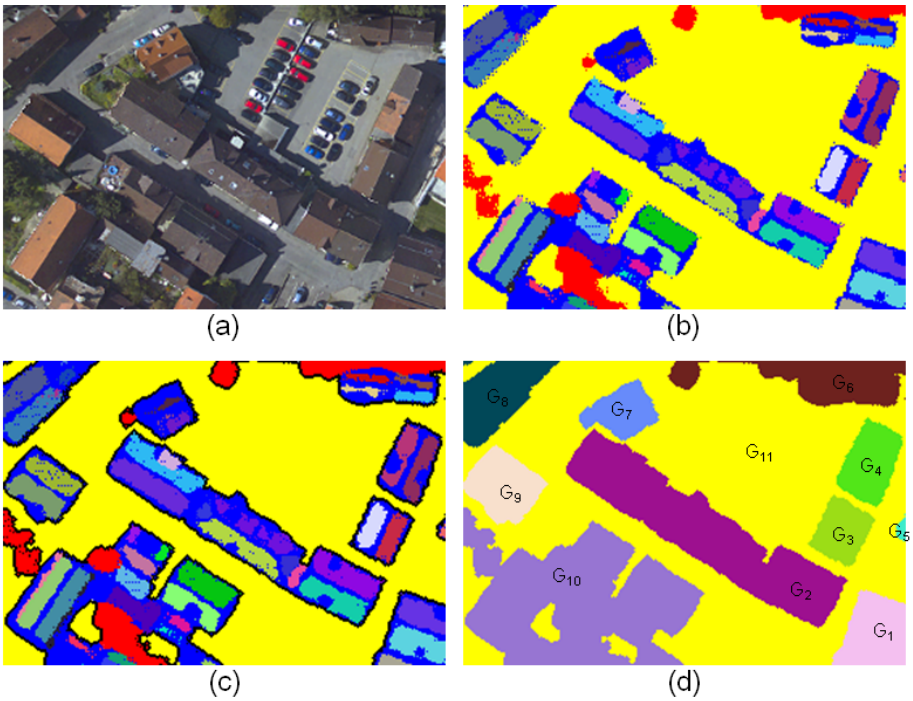


Figure 9: Partitioning of the grid on a downtown sample - *(a)* an aerial picture of the scene (not used), *(b)* the initial label map $l^{(ini)}$, *(c)* the initial label map $l^{(ini)}$ with a morphological erosion performed on the cells labeled as *ground* and illustrated by the black contours, and *(d)* the resulting partition where each of the 11 clusters is illustrated by a random color. The last cluster G_{11} corresponds to the eroded set of *ground* cells.

Table 1: Comparisons of different optimization techniques on a 1 km^2 dense urban area.

	Energy	Time
α-expansion [4]	2832.9	6.7 hrs
Belief propagation [38]	3016.6	10.3 hrs
α-expansion with our parallelization scheme (8 cores)	2853.3	209.3 sec

5 Representation of the urban elements

The three types of elements contained in the scenes are differently represented in 3D from the obtained label map. Buildings are modeled by combining arrangements of geometric 3D-primitives and mesh patches, trees by template matching, and the ground by a meshing procedure guarantying a continuous surface.

5.1 Buildings

A hybrid representation is used to model the buildings with a high level of generalization and a good compaction. Arrangements of geometric 3D-primitives for the standard roof sections, and mesh-patches describing the irregular roof components are combined.

The primitive arrangements are represented by polyhedral structures directly extracted from the label map obtained in Section 4. Note that, in case of non-planar primitives such as spheres or cylinders, the geometric accuracy of the polyhedral structure is fixed by a discretization parameter.

The mesh-patches are created by meshing, according to the 2D-grid, the 3D-points obtained from the cells labeled as *roof* in the label map (blue cells on the figures). As illustrated on Fig. 10, one of the main advantages of this strategy is the simplification of the mesh-patches while controlling the approximation error. A standard mesh simplification algorithm [14] can then be used to obtain more compact and coarser building representations.

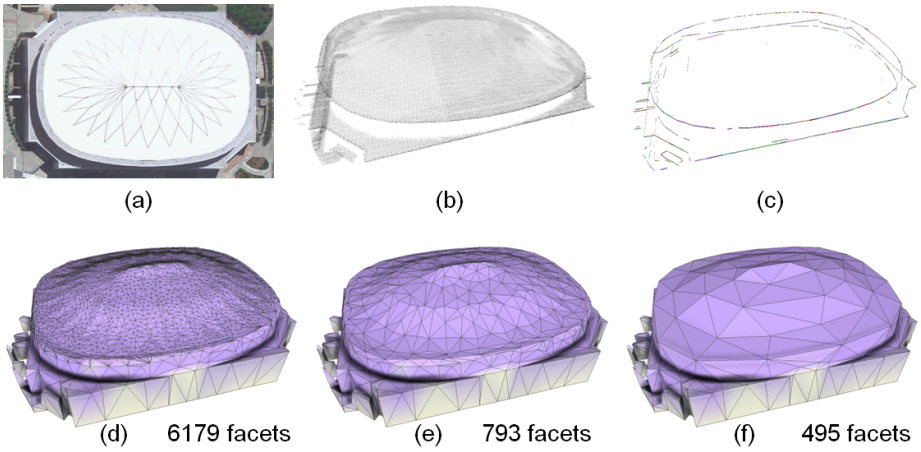


Figure 10: Simplification of the mesh patches on an irregular roof- *(a)* an aerial picture of the scene, *(b)* the input point set with a 17 pts/m^2 density, *(c)* the extracted 3D-segments, and the obtained 3D-models with *(d)* fine, *(e)* medium and *(f)* coarse mesh-patches. Note that the primitive arrangement and the facades are not affected by the simplification process.

The facades are obtained by projecting vertically the building contours on the estimated ground. The final result can be seen as a general triangular mesh in which the regular roof sections associated to a planar primitive are usually represented by one or two triangular facets and some finer mesh-patches describe the irregular components, as illustrated on Fig. 12.

5.2 Trees

They are reconstructed in 3D using template matching. The template is a simple ellipsoidal tree model whose compaction and rendering are well adapted to large urban scenes (see Fig. 11). For a street-view representation, one can imagine proposing a more realistic tree modeling, *e.g.* [40]. As directly matching an ellipsoid to the point set is computationally expensive, the center of mass of trees is first detected using a watershed algorithm performed on the estimated height of the cells labeled *vegetation*. The other parameters of the template such as the height and the radius of the crown are then simultaneously found by minimizing the Euclidean distance from points to an ellipse. The tree trunk is modeled by a cylinder which makes the link between the ellipsoid and the ground surface.

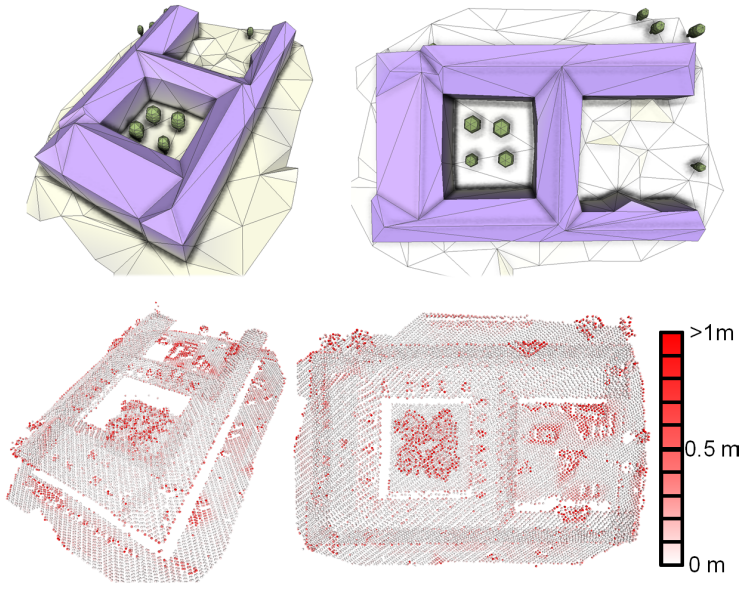


Figure 11: Object representation - (*top*) obtained 3D-model and (*bottom*) input cloud (2 pts/m^2) with the points colored according to their distance to the 3D-model. The high errors correspond to points from trees (the points of a tree do not obviously describe a perfect ellipsoidal shape) and from small urban components such as cars or roof superstructures. The mean error is 0.2 m, and the number of triangular facets is 205 without including the trees. Note that the surface primitives are divided into triangular facets for visual rendering and compaction measurement.

5.3 Ground

A standard meshing procedure is used to model the ground surface by a continuous surface. A grid of 3D-points is created from a spatial sub-sampling of the cells labeled as *ground*. It allows an accurate description without imposing any geometric constraints on the surface. Note that, similar to the mesh-patches of the buildings and the non-planar primitives, the mesh can be simplified using a decimation algorithm [14] to gain in compaction as shown in Fig. 12.

6 Experiments

6.1 Implementation and parameter settings

The algorithm has been implemented in C++ using the Computational Geometry Algorithms Library [6]. This library provides the basic geometric functions for the analysis of point clouds and the mesh processing. For example, this allows the search of neighbors in the input cloud or the computation of distances from point to parametric surfaces.

Several parameters are introduced during the four steps of the algorithm. One of the major strengths of the algorithm is that the point density of the input data does not interfere with the planimetric arrangement in terms of result quality and computation complexity. Thus, most of the parameters are stable on a large range of input data. The size of a cell s_c is usually chosen in the interval [0.2m,0.4m]. The radius r of the breakline-dependent neighborhood is fixed to $1.5s_c$. The parameters of the pairwise interactions in the planimetric arrangement model proposed in Eq. 10 are chosen as $\epsilon_1 = 0.5 \times \epsilon_2 = \frac{1}{3}$ and $\beta = 0.5$.

Other parameters depend on the input data types as shown in Tab. 2. This concerns the primitive extraction parameters, *i.e.* N_s, N_p, ϵ_s and ϵ_p which are sensitive to the point density of the input cloud and also to the acquisition type (Laser or MVS). The number of expansion cycles during the optimization of the label map (see Section 4.2) has also to be set according to the point density of the input data. More precisely, it must be set according to the proportion of empty cells in the map: the lower this ratio, the lower the number of expansion cycles.

Table 2: Parameter settings in function of the input data type

	N_s	N_p	ϵ_s	ϵ_p	Exp. cycles
Lidar, 2 pts/m^2	12	15	0.4	0.1	6
Lidar, 17 pts/m^2	25	100	0.2	0.1	4
MVS, 16 pts/m^2	15	120	0.5	0.5	4
MVS, 100 pts/m^2	35	500	0.4	0.5	2

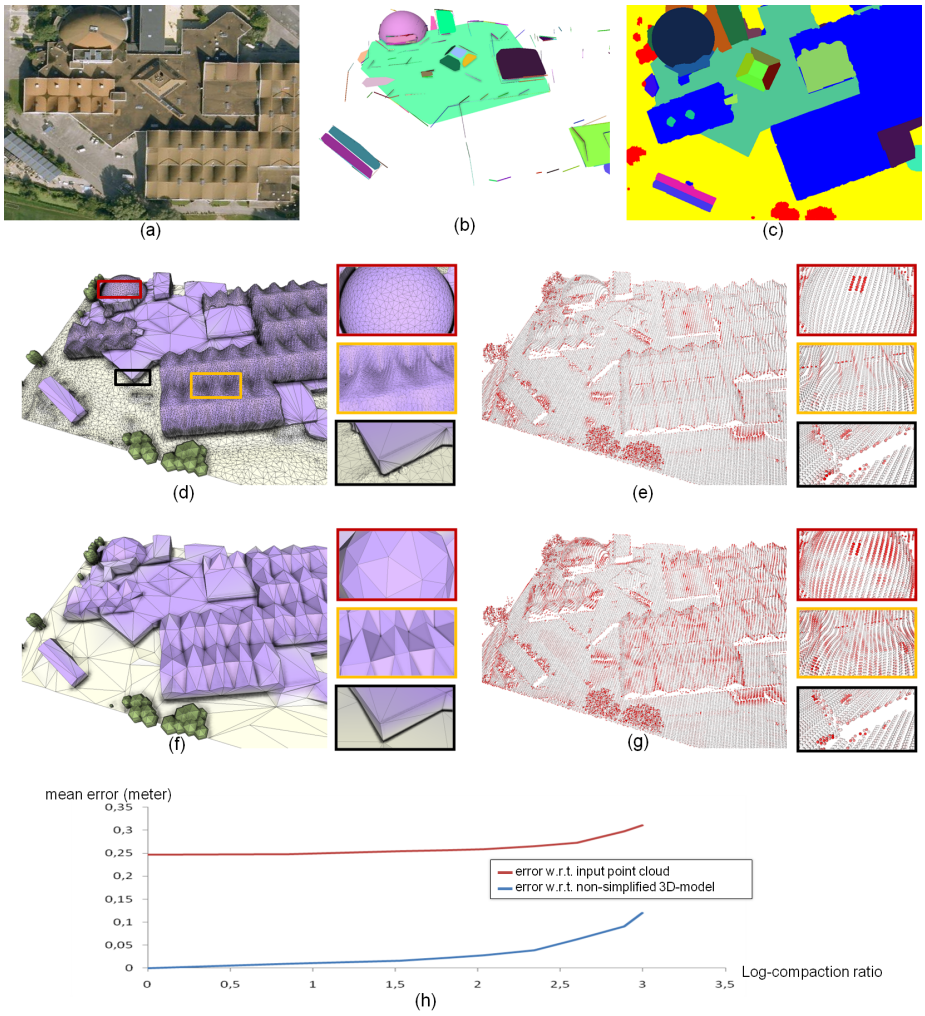


Figure 12: Hybrid reconstruction of a complex building - *(a)* aerial picture, *(b)* extracted 3D-primitives, *(c)* label map [color code: see Fig. 8], 3D-models obtained with *(d)* fine and *(f)* coarse mesh patch descriptions, input cloud (2 pts/m^2) with the points colored according to their distance to the *(e)* "fine" and *(g)* "coarse" 3D-models [color code: see Fig. 11], and *(h)* error graph of the 3D-models with mesh-patch simplification w.r.t. the input point cloud (red) and the unsimplified 3D-model (blue) in function of the log-compaction ratio of the unsimplified 3D-model to the simplified one. Our hybrid representation is particularly interesting in such a case: the building is accurately modeled by planes and a sphere for the regular parts, and by mesh-patches for the atypical surfaces, *i.e.* the undulating roofs. The fine (respectively coarse) 3D-model has $46K$ facets (resp. 864 facets) and a 0.24 m (resp. 0.33 m) mean error to the input data .

6.2 Visual considerations

Our method has been tested on various types of urban landscapes. Most of the datasets has been acquired by Laser, more precisely with lidar systems having different point densities from 2 and 17 pts/m^2.

Fig. 19 shows large scenes reconstructed with different types of landscapes including business districts with large and tall buildings, historic towns with a high concentration of both small buildings and trees, and hilly areas with high altimetric variations and dense forests. The input data generated from aerial laser scanning contain more than ten million of points. The results are obtained without using prior information on the landscape type and the object distributions within the scenes.

The level of detail of the results depends mainly on the input point density. For example, the roof details such as the dormer-windows or chimneys in Fig. 11 are described by less than 4 points in the 2 pts/m^2 density data. Our method ignores these sets of points in the computation of the main roof sections because they are too small to extract robust information. In Fig. 16, the input data has a 17 pts/m^2 density which is high enough to recover roof details such as the chimney. The building contours are correctly located, due to the breakline-dependent neighborhood introduced in the planimetric arrangement, even when they overlap at different locations with trees as shown in Fig. 13 (Building #2).

One of the main advantages of this hybrid representation is that the eventual primitive under-detection does not necessarily penalize the approach in terms of results. Indeed the regular roof sections missed during the geometric shape extraction stage are completed by mesh-patches. The final 3D-model remains coherent and correct even if it loses in terms of compaction. The eventual under-detection of 3D-segments is more penalizing, especially when the input cloud has both a spatially heterogeneous point distribution and a low point density. In such a case, the building 3D-models can have wavy contours which correspond to the shape induced by the bordering points of the building as shown in Fig. 12. One solution can be then to simplify the mesh but this engenders a loss of accuracy. On the other hand, over-detecting primitives would increase the number of labels during the planimetric arrangement, and thus, the computing times as well as the compaction of the 3D-model.

6.3 Performances

The evaluation of building reconstruction methods is a difficult task due to the absence of a benchmark in the field, the problems of data sharing as well as the difficulty in achieving ground truth. In order to measure the quality of the results, two main criteria are considered: the distance of the input points to the 3D-model and the compaction of the 3D-model. The mean distance on a 2 pts/m^2 density point cloud is typically contained in the interval [0.2 m, 0.35 m] (see Fig. 12 and 11). However, the mean distance is computed from all the points of the input data: this includes the outliers and the undesirable points corresponding to cars, fences or wires, which highly corrupt the obtained mean

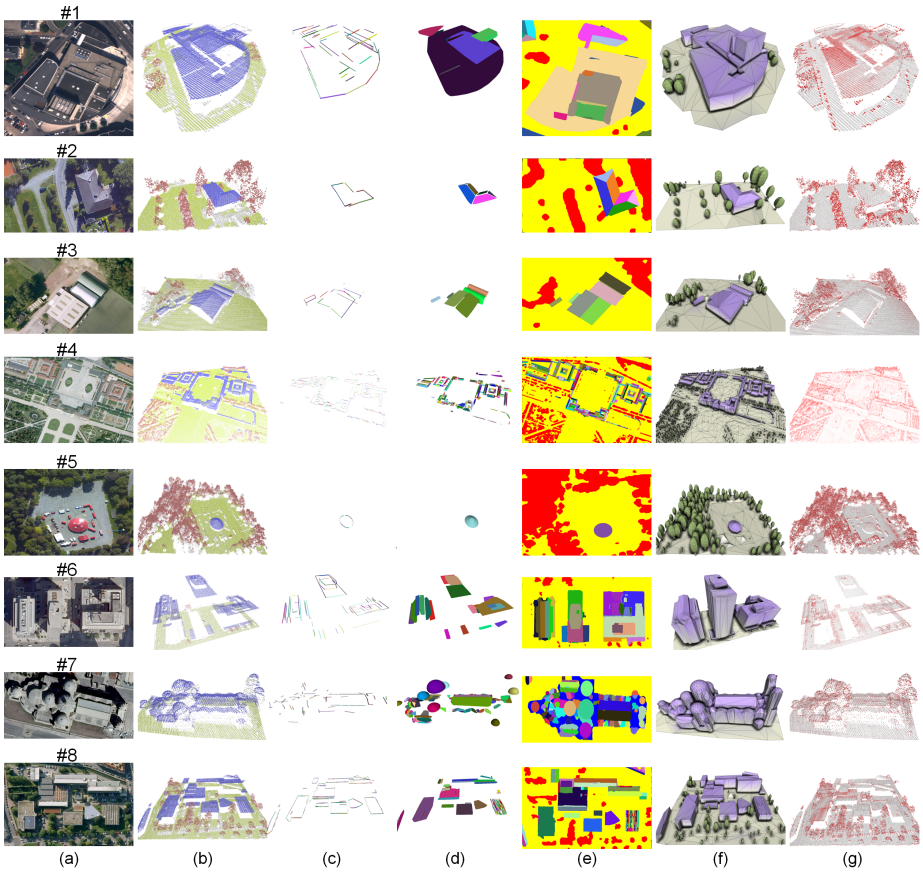


Figure 13: Results on several buildings with varying point densities (from 2 to 5 pts/m^2) - (a) aerial pictures, (b) classified point sets [color code: see Fig. 3], (c) extracted 3D-segments, (d) extracted surface primitives, (e) label maps, (f) obtained 3D-models, and (g) input point data with the points colored according to their distance to the associated 3D-models [color code: see Fig. 11]. Building #1 is an atypical piecewise planar structure with curved footprints. Building #2 is a classic house surrounded by trees. Note how the building and the trees are correctly reconstructed in spite of the fact they overlap at different locations. Building #3 is a simple structure with cylindrical parts. Building #4 is a Rococo-style castle with mainly gable and mansard roofs. Building #5 is a circus with a conoidal shape. Note how the trucks located around the circus are rejected as *clutter* during the point set classification. Building #6 represents a set of three north American skyscrapers which are particularly well adapted to the *Manhattan World* assumption. Building #7 is a Roman cathedral with a complex structure including spherical domes, small planar sections and irregular roof parts. Building #8 is a typical set of industrial structures.

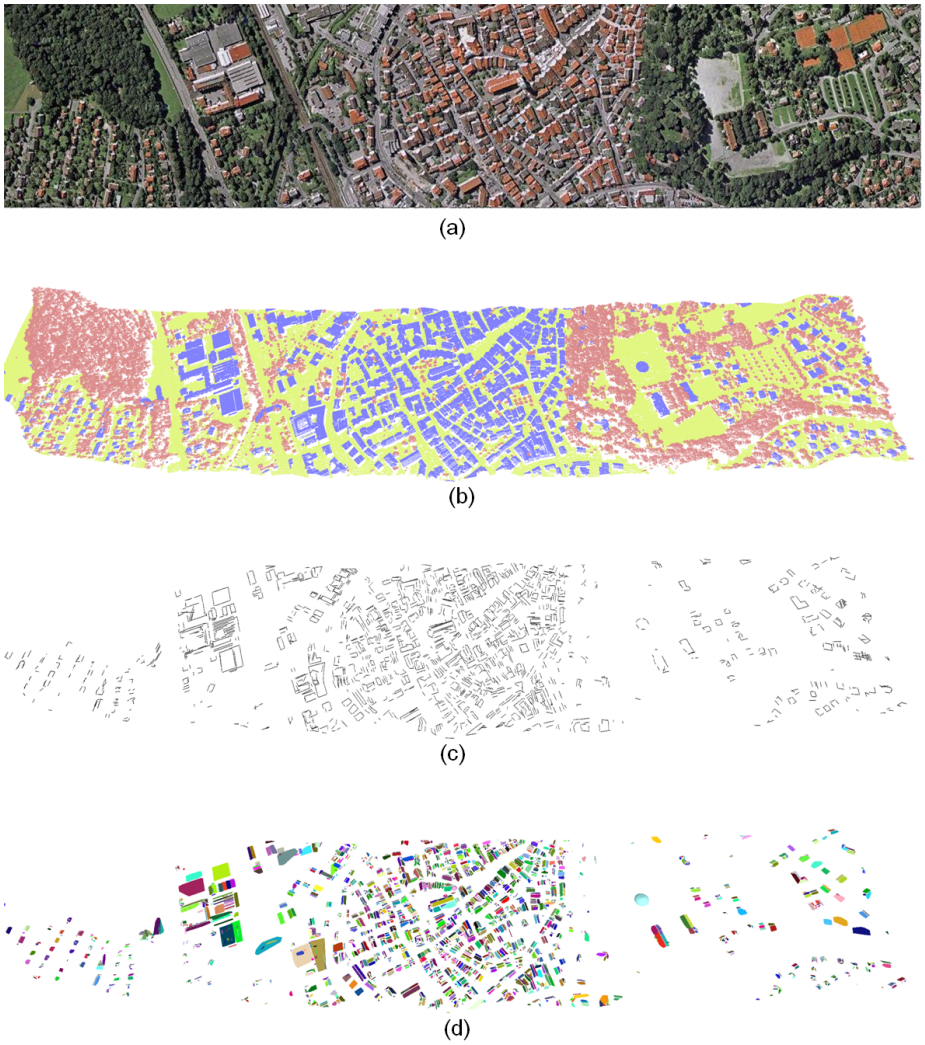


Figure 14: Result from a 2.3M point set representing a 1 km^2 area urban scene (Biberach, Germany) with a 128 m altimetric variation (Part I)- *(a)* an aerial picture (not used) of a city center, *(b)* the classified point set [color code: see Fig. 3], *(c)* the extracted 3D-segments, *(d)* the extracted surface primitives. Note that, as the aerial picture has been captured several years before the point data, some buildings are missing on this picture.

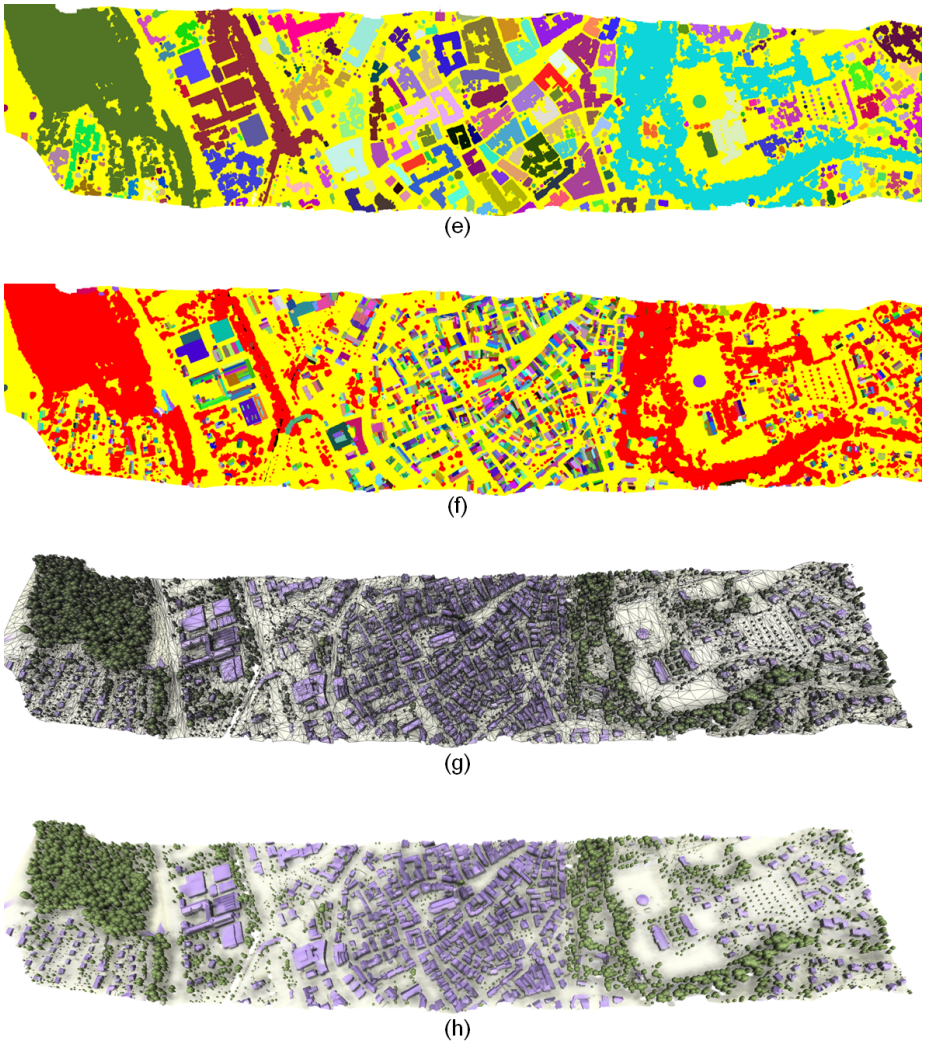


Figure 15: Result from a 2.3M point set representing a 1 km^2 area urban scene (Biberach, Germany) with a 128 m altimetric variation (Part II)- *(e)* the partition of the grid $\bigcup_{k\in[1,N]} G_k$ for the optimization decomposition [each cluster G_k is randomly colored], *(f)* the label map, the obtained 3D-model *(g)* with and *(h)* without mesh visualization. The result is obtained in approximatively 10 minutes.

error. Without taking these points into account, the mean error is usually inferior to 0.1 m.

We compare our method according to these two criteria to the mesh simplification algorithm proposed by Zhou *et al.* [42]. The compaction of our model is almost twice better, for a similar mean error to the input data as shown in Fig. 16.

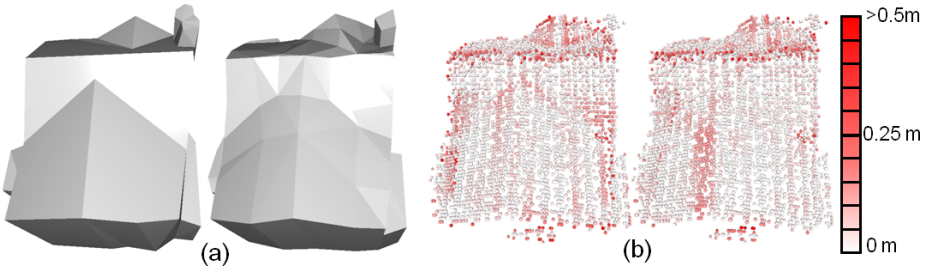


Figure 16: Comparison with a mesh simplification method- *(a)* 3D-models obtained *(left)* by our method and *(right)* by Zhou *et al.* [42], *(b)* input clouds (17pts/m^2) with the points colored according to their distance to the associated 3D-models. Our model presents a better roof component recovery. Although the mean errors to the input data are similar (0.07 m), the compaction of our model is almost twice better (126 *vs* 228 facets).

In addition, we evaluate the altimetric accuracy of the algorithm with respect to the ground truth obtained by the topographical measurement on two buildings, and compare it to a constructive solid geometry approach and a Digital Surface Model from point cloud as shown in Fig. 17. From a 2 pts/m^2 density input data, we obtain the best mean error, *i.e.* 0.21 m, on the evaluated buildings in spite of some high local errors on the contours illustrated by the thin black lines partially surrounding the buildings on the altimetric error maps. From such a low point density, it is indeed difficult to perfectly extract the building contours.

In regard to tree detection, the results are satisfactory. The false alarm rate and the under-detection rate are respectively estimated to 2% and 6% on the Amiens dataset. However, certain building contour points associated with atypical roof sections may be detected as vegetation, especially when the scatter feature F_s is computed without using echo information (see Fig. 19, top right crop).

Around 10 minutes is required to model a 1 km^2 dense urban area using a single computer. The computing times are competitive compared to most of the large scale modeling algorithms, *e.g.* [29] with around half an hour per km^2, or [26] who require several interactive operations per building.

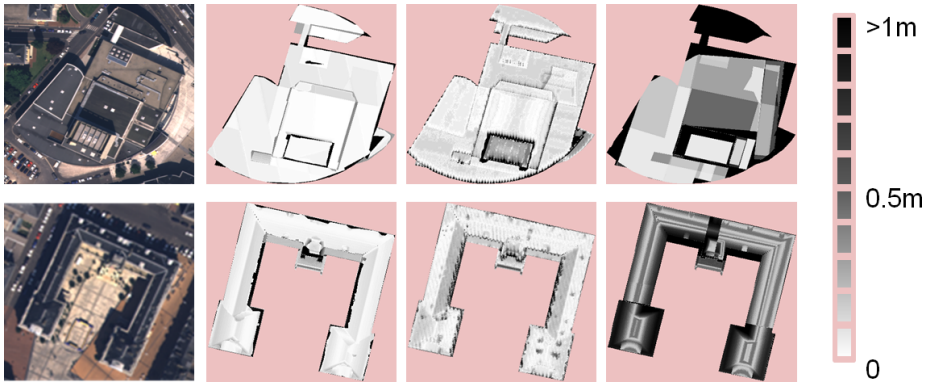


Figure 17: Comparison with pixel-based and primitive-based approaches in terms of altimetric accuracy w.r.t. ground truth - (*from left to right*) aerial pictures of two buildings, altimetric error maps for our method, for a Digital Surface Model from point cloud, and for the Constructive Solid Geometry approach by Lafarge *et al.* [18]. Note how the roof sections from our method are accurately estimated when compared to the other algorithms.

6.4 Point clouds from Laser or MVS?

The acquisition type of the input data impacts on the result quality provided by our method. Several works, such as the study of Leberl *et al.* [20], compare the potential of Laser and MVS for urban scene analysis. Such comparisons are usually difficult to realize as the performances depend strongly on the own characteristics of the acquisition system, and also on the dense stereo algorithm used to generate the DSM in case of MVS acquisition.

Contrary to the point sets from MVS, Laser-based point clouds have a high altimetric accuracy but a heterogeneous planimetric distribution, and usually a lower point density. These differences play an important role during the surface primitive extraction step. As illustrated in Tab. 2, the maximal fitting errors and the minimal numbers of fitted points per primitive must be higher in the case of a MVS-based input data in order to compensate for the approximative altimetric accuracy of the points. In order to improve the surface primitive extraction procedure in the case of low resolution MVS-based input data, one can substitute the quadratic error (see Eq. 3) by a softer distance such as the L_1-norm error which is frequently used from MVS-based DSM computations [39].

At low resolutions, the DSM-based point clouds do not have strongly marked discontinuities on the building contours as shown on the crops in Fig. 18. This is due to the dense stereo algorithms used to generate the DSM which usually introduce smoothness constraints on the surface. This point penalizes the recovery of the building contours compared to the Laser acquisition.

The tree detection is more efficient from Laser-based point clouds than from MVS-based data. Indeed, the point diffusion of a tree is not a simple surface as

for the MVS-based points which makes its recognition easy.

Finally, our algorithm globally provides better results from Laser than from MVS. 3D-models from MVS-based point sets at low resolution usually have shape approximation errors. At high resolution, the results are similar to those obtained from Laser but the computation times of the first and second steps of the algorithm are higher.

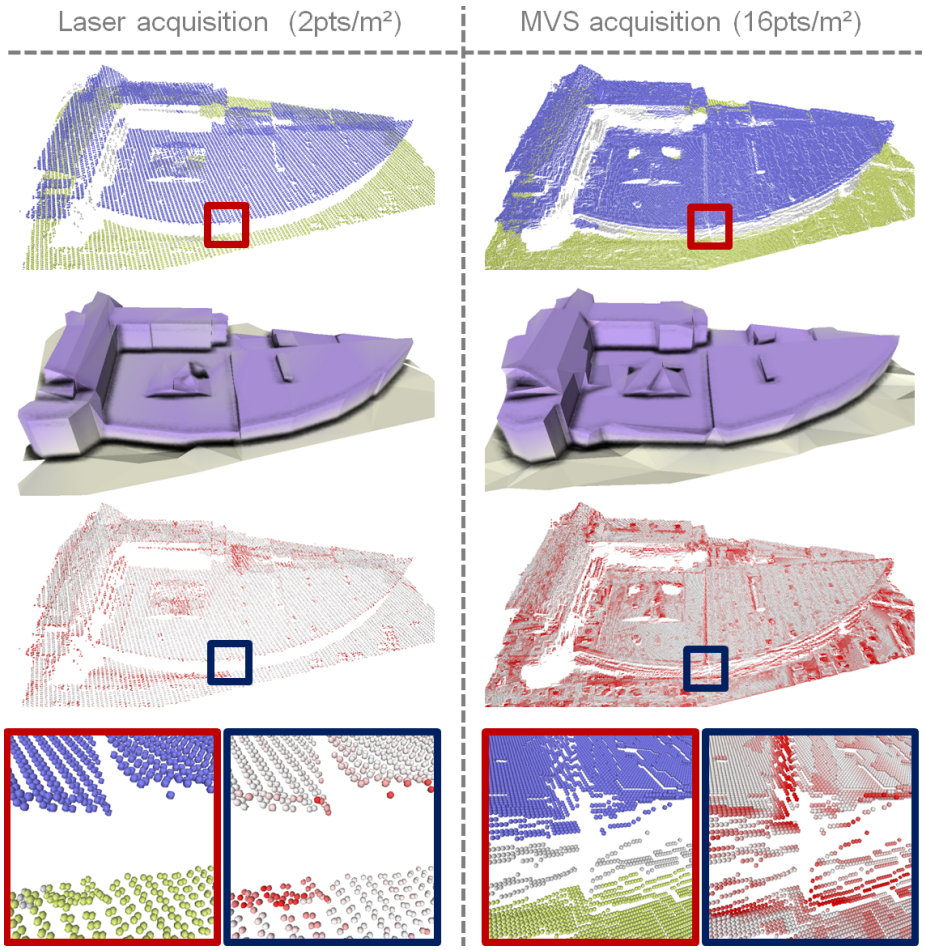


Figure 18: Impact of the acquisition system type on our approach- *(from top to bottom)* the classified point clouds, the obtained 3D-models, the input clouds with the points colored according to their distance to the associated 3D-models [color code: see Fig. 11], and some associated crops. The 3D-model from MVS is less accurate in terms of structure contouring, but is of similar quality concerning the surface recovery. The Laser-based 3D-model *(left)* has a mean error to the input data slightly lower than the MVS-based 3D-model one *(right)*, *i.e.* 0.26 m *vs* 0.33 m.

6.5 Limitations

First, some urban components are not taken into account in our representation. In particular, the bridges and the elevated roads which are local planar structures elevated above the ground are frequently detected as buildings (see Fig. 19, top right crop). This problem can be solved by considering additional urban components in the point cloud classification. Note that in this perspective, the energy formulation of the planimetric arrangement can be easily adapted. Secondly, the modeling of the trees is restricted to the use of an ellipsoidal shape template. It is sufficient for large scene descriptions but too limited for street-view representations. In light of this, it seems relevant to introduce a library of tree forms and create more complex dependencies between neighboring elements. Thirdly, our algorithm is not optimal when both the altimetric accuracy of the input points is poor and the point density is weak, typically with low resolution Digital Surface Models, *i.e.* >0.5 m. In such cases, it is necessary to use less generic methods based on very strong urban assumptions, such as the structural-based approach proposed in [18], in order to compensate for the poor quality of the data.

7 Conclusion

We propose an original approach for modeling large urban environments from 3D-point data. An important strength of the algorithm compared to existing methods is the complete and realistic semantized description of urban scenes by simultaneously reconstructing buildings, trees and topologically complex ground surfaces, but also the original hybrid representation of buildings combining a high level of generalization and compaction. Moreover, a general mathematical formulation for roof section arrangement problems is defined, the first to date to our knowledge which works in non-restricted contexts. In future works, it would be interesting to improve the parallelization scheme of the energy minimization by using GPU. Another interesting challenge is to adapt our approach to point clouds generated from Internet photo collections [1, 11] which contain more outliers and have spatial distributions highly heterogeneous.

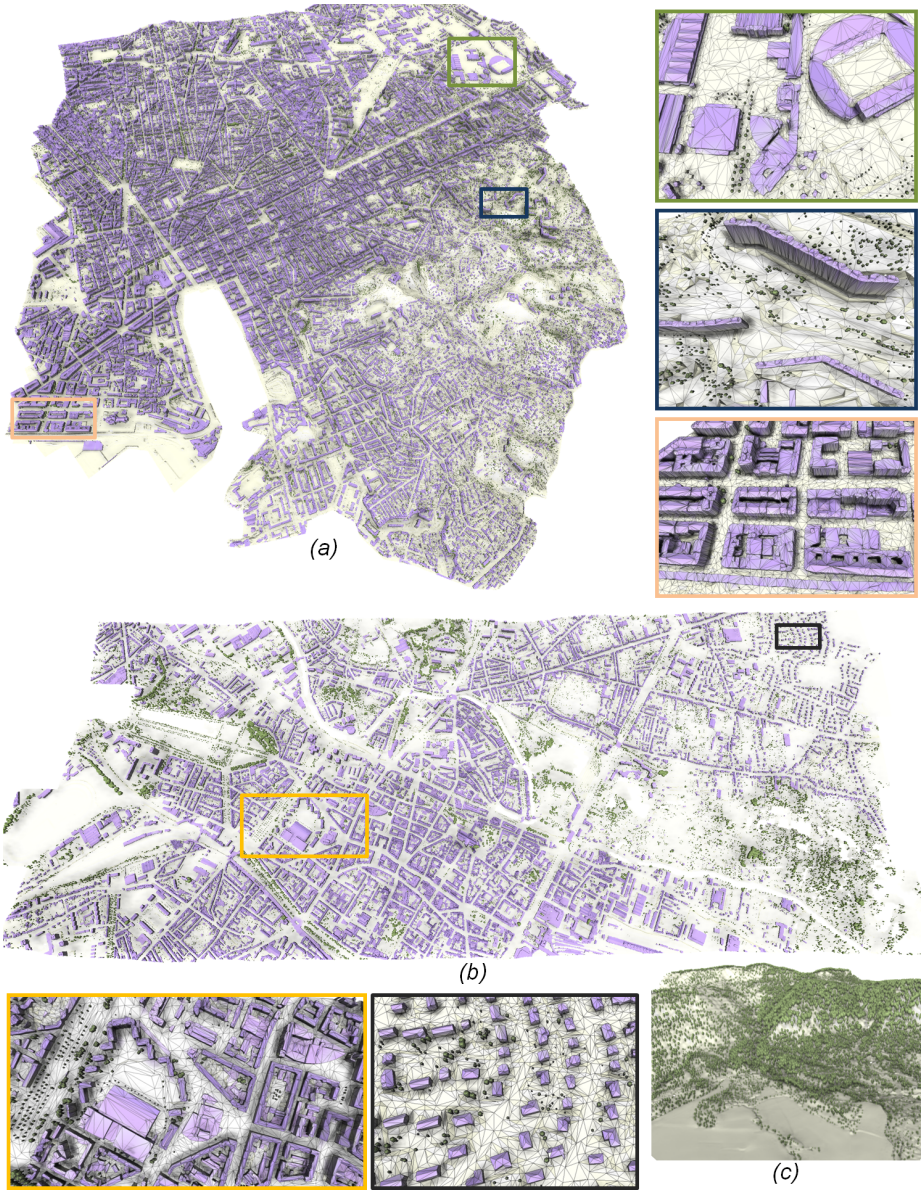
Acknowledgments

We are grateful to the GDR Isis for the partial financial support. We thank the city of Vienna for the Lidar scan of the Schönbrunn castle, Quian-Yi Zhou for the two Atlanta high density Lidar scan samples (Fig. 10 and 16), Sean Belshaw from Optech for the Toronto Lidar scan sample, and the French Mapping Agency for the other datasets, including the DSM-based point clouds.

References

[1] S. Agarwal, N. Snavely, I. Simon, S.M. Seitz, and R. Szeliski. Building Rome in a day. In *ICCV*, Kyoto, Japan, 2009.

[2] C. Baillard and A. Zisserman. Automatic reconstruction of piecewise planar models from multiple views. In *CVPR*, Los Alamitos, US, 1999.



	Marseille *(a)*	Amiens *(b)*	Mountain area *(c)*
#input points ($\times 10^6$)	38.67	24.52	22.67
area (km^2)	19.8	11.57	3.41
altimetric variation (m)	192	76	525
#primitives ($\times 10^3$)	108.6	56.7	0.01
#trees ($\times 10^3$)	35.7	22.8	21.1
computing time (hour)	2.52	1.34	0.31
compaction (Mo)	131	93	34

Figure 19: Reconstruction of three large scenes with some performance statistics and crops on various types of urban landscapes.

[3] A. Banno, T. Masuda, T. Oishi, and K. Ikeuchi. Flying laser range sensor for large-scale site-modeling and its applications in Bayon digital archival project. *IJCV*, 78(2-3), 2008.

[4] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *PAMI*, 23(11), 2001.

[5] C. Briese, N. Pfeifer, and P. Dorninger. Applications of the robust interpolation for DTM determination. In *PCV*, Graz, Austria, 2002.

[6] CGAL. www.cgal.org, 2011.

[7] A.-L. Chauve, P. Labatut, and J.-P. Pons. Robust piecewise-planar 3D reconstruction and completion from large-scale unstructured point data. In *CVPR*, San Francisco, US, 2010.

[8] J. Chen and B. Chen. Architectural modeling from sparsely scanned range data. *IJCV*, 78(2-3), 2008.

[9] J. M. Coughlan and A. L. Yuille. The Manhattan world assumption: Regularities in scene statistics which enable Bayesian inference. In *NIPS*, Denver, US, 2000.

[10] A.R. Dick, P.H.S. Torr, and R. Cipolla. Modelling and interpretation of architecture from several images. *IJCV*, 60(2), 2004.

[11] J.-M. Frahm et al. Building Rome on a cloudless day. In *ECCV*, Hersonissos, Greece, 2010.

[12] C. Frueh and A. Zakhor. An automated method for large-scale, ground-based city model acquisition. *IJCV*, 60(1), 2004.

[13] Y. Furukawa, B. Curless, S.M. Seitz, and R. Szeliski. Manhattan-world stereo. In *CVPR*, Miami, US, 2009.

[14] M. Garland and P. Heckbert. Surface simplification using quadric error metrics. In *SIGGRAPH*, Los Angeles, US, 1997.

[15] A. Golovinskiy, V.G. Kim, and T. Funkhouser. Shape-based recognition of 3D point clouds in urban environments. In *ICCV*, Kyoto, Japan, 2009.

[16] N. Haala and M. Kada. An update on automatic 3D building reconstruction. *Journal of Photogrammetry and Remote Sensing*, 65(6), 2010.

[17] F. Han, Z. W. Tu, and S. C. Zhu. Range image segmentation by an effective jump-diffusion method. *PAMI*, 26(9), 2004.

[18] F. Lafarge, X. Descombes, J. Zerubia, and M. Pierrot-Deseilligny. Structural approach for building reconstruction from a single DSM. *PAMI*, 32(1), 2010.

[19] F. Lafarge, R. Keriven, M. Bredif, and H. Vu. Hybrid multi-view reconstruction by jump-diffusion. In *CVPR*, San Francisco, US, 2010.

[20] F. Leberl, A. Irschara, T. Pock, P. Meixner, M. Gruber, S. Scholz, and A. Wiechert. Point clouds: Lidar versus 3d vision. *Photogrammetric Engineering and Remote Sensing*, 76(10), 2010.

[21] S.Z. Li. *Markov Random Field Modeling in Image Analysis*. Springer, 2001.

[22] C. Mallet and F. Bretar. Full-waveform topographic lidar: State-of-the-art. *Journal of Photogrammetry and Remote Sensing*, 64(1), 2009.

[23] D. Marshall, G. Lukacs, and R. Martin. Robust segmentation of primitives from range data in the presence of geometric degeneracy. *PAMI*, 23(3), 2001.

[24] B. Matei, H. Sawhney, S. Samarasekera, J. Kim, and R. Kumar. Building segmentation for densely built urban regions using aerial lidar data. In *CVPR*, Anchorage, US, 2008.

[25] H. Mayer. Object extraction in photogrammetric computer vision. *Journal of Photogrammetry and Remote Sensing*, 63(2), 2008.

[26] P. Muller, P. Wonka, S. Haegler, A. Ulmer, and L. Van Gool. Procedural modeling of buildings. In *SIGGRAPH*, Boston, 2006.

[27] D. Munoz, J. A. Bagnell, N. Vandapel, and M. Hebert. Contextual classification with functional max-margin Markov networks. In *CVPR*, Miami, US, 2009.

[28] M. Pollefeys et al. Detailed real-time urban 3D reconstruction from video. *IJCV*, 78(2-3), 2008.

[29] C. Poullis and S. You. Automatic reconstruction of cities from remote sensor data. In *CVPR*, Miami, US, 2009.

[30] R. Schnabel, R. Wahl, and R. Klein. Efficient RANSAC for point-cloud shape detection. *Computer Graphics Forum*, 26(2), 2007.

[31] S. N. Sinha, D. Steedly, and R. Szeliski. Piecewise planar stereo for image-based rendering. In *ICCV*, Kyoto, Japan, 2009.

[32] C. Strecha, W. Von Hansen, L. Van Gool, P. Fua, and U. Thoennessen. On benchmarking camera calibration and multi-view stereo for high resolution imagery. In *CVPR*, Anchorage, US, 2008.

[33] A. Toshev, P. Mordohai, and B. Taskar. Detecting and parsing architecture at city scale from range data. In *CVPR*, San Francisco, US, 2010.

[34] R. Tse, C.M Gold, and D.B. Kidner. Using the delaunay triangulation/ voronoi diagram to extract building information from raw lidar data. In *Proc. of International Symposium on Voronoi Diagrams in Science and Engineering*, Urmchi, China, 2007.

[35] C. Vanegas, D. Aliaga, and B. Benes. Building reconstruction using Manhattan-world grammars. In *CVPR*, San Francisco, US, 2010.

[36] G. Vosselman, P. Kessels, and B. Gorte. The utilisation of airborne laser scanning for mapping. *Int. Jour. of Applied Earth Observation and Geoinformation*, 6(3-4), 2005.

[37] H. Vu, R. Keriven, P. Labatut, and J.P. Pons. Towards high-resolution large-scale multiview. In *CVPR*, Miami, US, 2009.

[38] Y. Weiss and W.T. Freeman. On the optimality of solutions of the max-product belief propagation algorithm in arbitrary graphs. *IEEE Trans. on Information Theory*, 47(2), 2001.

[39] G. Xu and Z. Zhang. *Epipolar geometry in stereo, motion and object recognition.* Kluwer, 1996.

[40] H. Xu, N. Gossett, and B. Chen. Knowledge and heuristic-based modeling of laser-scanned trees. *Trans. on Graphics*, 26(4), 2007.

[41] L. Zebedin, J. Bauer, K.F. Karner, and H. Bischof. Fusion of feature- and area-based information for urban buildings modeling from aerial imagery. In *ECCV*, Marseille, France, 2008.

[42] Q.Y. Zhou and U. Neumann. 2.5*d* dual contouring: A robust approach to creating building models from aerial lidar point clouds. In *ECCV*, Hersonissos, Greece, 2010.

[43] Z. Zhu and T. Kanade. Special issue on modeling and representations of large-scale 3D scenes. *IJCV*, 78(2-3), 2008.

Centre de recherche INRIA Sophia Antipolis – Méditerranée
2004, route des Lucioles - BP 93 - 06902 Sophia Antipolis Cedex (France)

Centre de recherche INRIA Bordeaux – Sud Ouest : Domaine Universitaire - 351, cours de la Libération - 33405 Talence Cedex
Centre de recherche INRIA Grenoble – Rhône-Alpes : 655, avenue de l'Europe - 38334 Montbonnot Saint-Ismier
Centre de recherche INRIA Lille – Nord Europe : Parc Scientifique de la Haute Borne - 40, avenue Halley - 59650 Villeneuve d'Ascq
Centre de recherche INRIA Nancy – Grand Est : LORIA, Technopôle de Nancy-Brabois - Campus scientifique
615, rue du Jardin Botanique - BP 101 - 54602 Villers-lès-Nancy Cedex
Centre de recherche INRIA Paris – Rocquencourt : Domaine de Voluceau - Rocquencourt - BP 105 - 78153 Le Chesnay Cedex
Centre de recherche INRIA Rennes – Bretagne Atlantique : IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex
Centre de recherche INRIA Saclay – Île-de-France : Parc Orsay Université - ZAC des Vignes : 4, rue Jacques Monod - 91893 Orsay Cedex

Éditeur
INRIA - Domaine de Voluceau - Rocquencourt, BP 105 - 78153 Le Chesnay Cedex (France)
http://www.inria.fr
ISSN 0249-0803