Building large urban environments from unstructured point data

Florent Lafarge, Clément Mallet

▶ **To cite this version:**

Florent Lafarge, Clément Mallet. Building large urban environments from unstructured point data. ICCV, Nov 2011, Barcelona, Spain. hal-00759269

HAL Id: hal-00759269

https://inria.hal.science/hal-00759269v1

Submitted on 30 Nov 2012

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

HAL Authorization

Building large urban environments from unstructured point data

Florent Lafarge
INRIA Sophia Antipolis
florent.lafarge@inria.fr

Clément Mallet
Université Paris Est, IGN
clement.mallet@ign.fr

Abstract

We present a robust method for modeling cities from unstructured point data. Our algorithm provides a more complete description than existing approaches by reconstructing simultaneously buildings, trees and topologically complex grounds. Buildings are modeled by an original approach 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. We experimentally validate the approach on complex urban structures and large urban scenes of millions of points.

1. Introduction

The 3D reconstruction of urban environments is a topic of major interest in computer vision. Driven by new virtual applications, this research domain has considerably progressed during the last decade.

Related works on urban scene modeling - Many works are devoted to street level modeling from ground or oblique-view data, *e.g.* [4, 8, 9, 21]. Other works propose large city descriptions from airborne data, and offer complementary advantages to street level representations such as fine roof descriptions. Most of the 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 large city modeling. Lafarge *et al.* [13] and Zebedin *et al.* [24] generate DSMs from MultiView Stereo (MVS) imagery in order to model buildings using polyhedral structures. Other approaches consider unstructured point clouds obtained from Laser/lidar systems [17, 19, 22, 25] or MVS imagery [6]. Such data have spatially heterogeneous point distributions without induced neighborhood relationship between the points, and contain outliers, especially when generated from MVS imagery. Matei *et al.* [17] and Poullis *et al.* [19] propose flat roof models adapted to "Manhattan-world" environments. Both approaches put efforts in segmenting the buildings and simplifying their boundaries, either by estimating building orientations [17] or by using statistical considerations [19]. A more general building representation is proposed in [25] where a mesh simplification procedure based on dual contouring is used. Although this approach wins in level of generalization, semantic information is lost: a simple planar roof section can be described by many mesh facets with different normal orientations. These approaches provide convincing 3D models but do not address two important issues. 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 very efficient for "Manhattan-world" environments but become penalizing for less "well-organized" urban landscapes 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.

Contributions - We propose an algorithm which brings solutions to address the problems mentioned above.

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

• *Hybrid reconstruction:* our 3D-model 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. Those two different types of 3D-representation tools interact through a non-

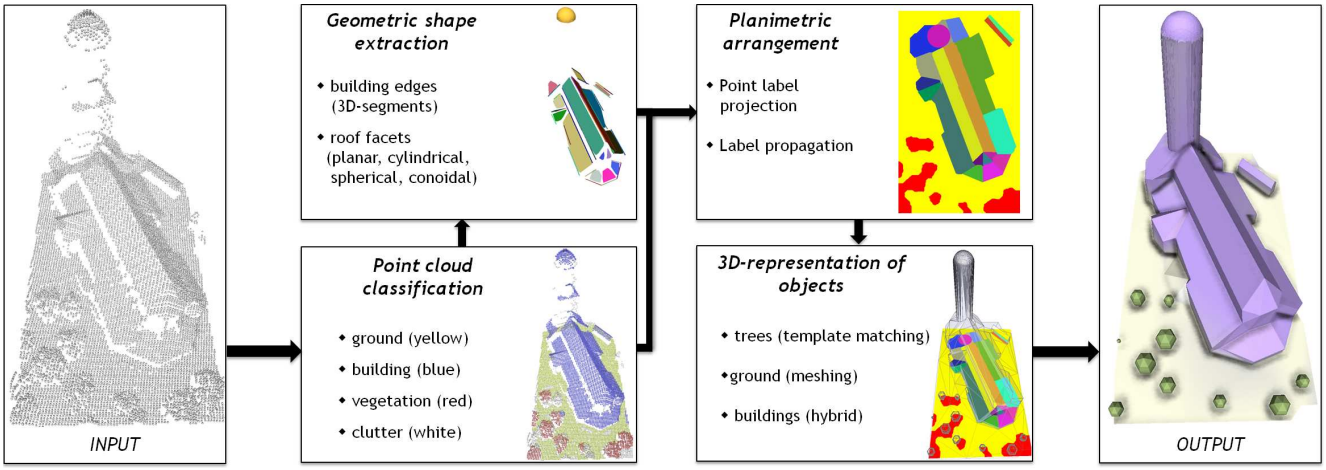


Figure 1. Overview of the proposed approach - Our algorithm digests large amounts of unstructured points in order to provide a compact and semantized representation of urban environments including atypical buildings, trees, and topologically complex grounds.

convex energy minimization problem. Thus, the 3D-models provide urban details while being semantized and compact.

• *2.5D-arrangement scheme of urban structures:* we present a general formulation for roof section arrangement problems, the first to date to our knowledge which works in non-restricted contexts (*i.e.* unspecified primitives, various types of urban objects interacting in the scene, unknown building contours).

A four-step strategy illustrated in Fig. 1 is adopted. First, the lidar point cloud is classified using an unsupervised method presented in Section 2. Four classes 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. Note that technical details and additionnal experiments could be found in [14].

2. Point cloud classification

Four classes of interest are defined: *building*, *vegetation*, *ground*, and *clutter*. 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.

Discriminative features - Several geometric attributes computed for each point are proposed 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. The response to this feature is weak in the case of 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 estimated elevation map of the ground.

• *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 considering point and its neighbors [22].

• *Clutter* f_c is dedicated to outliers and undesirable components having a linear structure. It is defined as the Euclidian distance between the point and the optimal 3D-line among its neighbors, weighted by the number of neighbors.

In order to tune the sensitivity of each feature, four parameters σ_e, σ_p, σ_s and σ_c are introduced. The features are then normalized by a linear projection on the interval $[0, 1]$.

Non-supervised problem formulation - An energy minimization is proposed to classify the point cloud. The energy is a sum of two terms: a pairwise interaction defined by the standard Potts model [15] whose parameter is denoted γ, and a data component E_d based on a combination of the normalized features defined above given by

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

where x_i is the class of the point i and E_{di} is the partial data term of the point i with $E_d = \sum_i E_{di}$. A Graph-Cut based algorithm [5] is used to quickly reach an approximate solution close to the global optimum of the 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 data term E_d. The parameter γ which balances the Potts interaction with respect to the data term, is set to $(2.\widehat{p})^{-1}$ where $\widehat{p}$ is the average point density of the cloud. We set σ_e to 6 m (*i.e.* the height of two floors), σ_s to 0.5, σ_p to 0.5 m, and σ_c to 0.25 m (*i.e.* critical distances to be considered as planar and linear, respectively). One can imagine tuning these parameters using a learning procedure, as for example in [11]. However, we notice that these values are stable on a wide range of data in spite of the simplicity of the model, as explained in [14]. Thus, this would unnecessarily make the system heavier.

3. Geometric shape extraction

3D-primitives are then extracted 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 [20, 22], is not required. Two types of elements are detected.

3D-segments- These 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 missing parts between them (see Fig. 2). First, the points located on the building borders are selected from the *building* point set. A point is selected if its Euclidean distance to the optimal 3D-line among its neighbors is inferior to $(2\sqrt{\widehat{p}})^{-1}$. Then, 3D-lines are detected by region growing from the selected points. The 3D-segments are obtained by projecting the accumulated points on the detected 3D-lines.

Planar, spherical, cylindrical, and conoidal shapes- They are extracted on the regular sections of the roofs. First, the planar structures, which constitute the most common shape of roofs, are detected. A region growing allows the fast extraction of 3D-planes. The propagation criterion tests whether the normal of the considered point is similar to the average normal of the points in the region. Non-planar shapes are then extracted from the points unfitted to a plane. Fitting spheres, cylinders or cones have 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 [12]. 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 [16] are used to achieve numerically stable fittings. Extracting the 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.

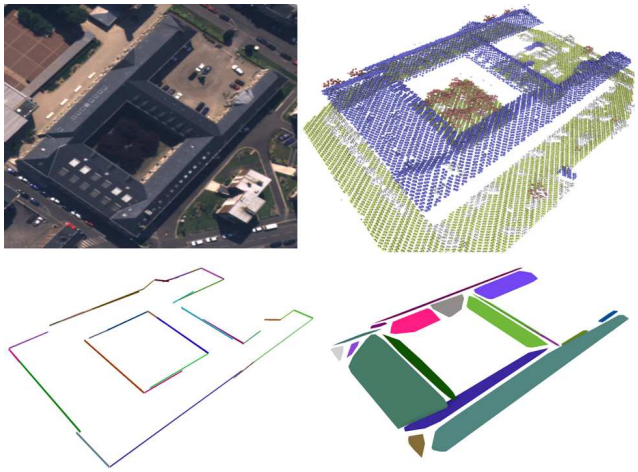


Figure 2. Shape extraction - (*top-left*) aerial image of a building, (*top-right*) classified point cloud [color code: blue=*building*, red=*vegetation*, yellow= *ground* and white= *clutter*], (*bottom-left*) extracted 3D-segments and (*bottom-right*) extracted areal primitives. The main regular roof sections of the buildings are detected as well as the global building contours. Note that the planes are visually represented by their convex envelopes.

4. Planimetric arrangement

The third step 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 in [3] for simple houses. Revolution sections are also taken into account in [24], 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 (see Fig. 3). Performing the arrangement on such a grid, called a planimetric map in the following, allows us to highly reduce the problem complexity by assuming a 2.5D representation of urban scenes.

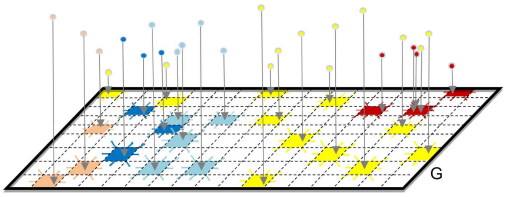


Figure 3. The labels of the points are first projected on a 2D-grid G, and then propagated under arrangement constraints.

Point labels - Each point of the cloud is associated with the label $ground$, $vegetation$, $clutter$, $plane^{(k)}$, $cylinder^{(l)}$, $sphere^{(m)}$, $cone^{(n)}$ or $roof$. The points labeled as $clutter$ are not taken into account in the following. The label $roof$ corresponds to the points classified as *'building'* in Section 2, which have not been fitted to planar, spherical, cylindrical or conoidal shapes. The point labels are projected on a 2D-grid G as illustrated on Fig. 3. We denote by $G^{(proj)}$, the subset of G composed of the cells on which at least one point label has been projected, and $G^{(empty)}$ its complementary subset on G. The labels are then propagated on the entire grid G under structure arrangement constraints.

Label propagation under arrangement 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. The quality of a configuration l is measured by the energy U of the standard form:

$$U(l) = \sum_{i \in G} D_i(l_i) + \beta \sum_{\{i,j\} \in E} V_{ij}(l_i, l_j) \qquad (2)$$

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:

$$\{i,j\} \in E \Leftrightarrow \begin{cases} \|i-j\|_2 \leq r \\ \mathcal{O}(i, \mathcal{L}_k) = \mathcal{O}(j, \mathcal{L}_k) \end{cases} \qquad (3)$$

where $\mathcal{L}_k$ is the 2D-line obtained by projecting on the planimetric map the k^{th} 3D-segment interacting with the pair $\{i,j\}$ (see Fig. 4). $\mathcal{O}(i, \mathcal{L})$ 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.

Data term - D_i checks the coherence of the label l_i at the cell i with respect to the input point cloud. We have

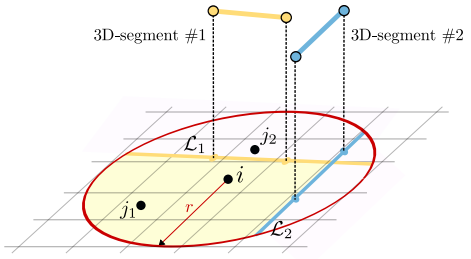


Figure 4. Breakline-dependent neighborhood - The neighbors of the cell i are contained in the yellow area. $\{i, j_1\} \in E$ but $\{i, j_2\} \notin E$.

$$D_i(l_i) = \begin{cases} c & \text{if } l_i = roof \\ \mathbb{1}_{\{i \in G^{(proj)}\}} \min(1, |z_{l_i} - z_{p_i}|) & \text{otherwise} \end{cases} \qquad (4)$$

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

Propagation constraints - V_{ij} allows both the label consistency and a coherent structure arrangement. To do so, an arrangement law, denoted by $\bowtie$, is introduced to test whether two labels of neighboring cells are spatially coherent. Two labels l_i and l_j are said coherent (denoted by $l_i \bowtie l_j$) if $\mathcal{O}(i, \mathcal{I}_{l_i,l_j}) \neq \mathcal{O}(j, \mathcal{I}_{l_i,l_j})$ 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 checks that the projection in XY of the 3D-line intersecting the two planes is located in between the two cells (see Fig. 5). Thus, the exact separation of two connected planes is constrained as shown in Fig. 6. The pairwise interaction is then given by:

$$V_{ij}(l_i, l_j) = \begin{cases} \epsilon_1 & \text{if } l_i \bowtie l_j \\ \epsilon_2 & \text{if } l_i = l_j \\ 1 & \text{otherwise} \end{cases} \qquad (5)$$

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

Optimization with parallelization scheme- Finding the label configuration which minimizes the energy U is a non-convex optimization problem. Simulated annealing techniques [15], graph-cut based algorithms [5] or belief propagation methods [23] could provide a good approximation of the solution but at the expense of high computing time. The scenes are generally of large scale and the number of labels is very high. In order to reach reasonable computing times,

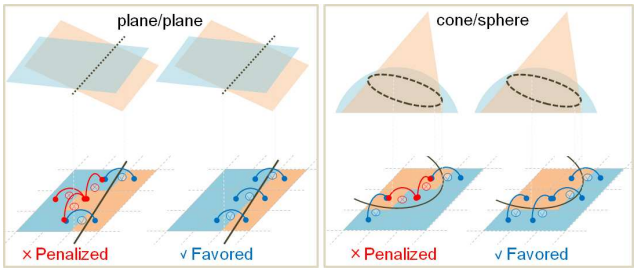


Figure 5. Principle of the $\bowtie$-law on two examples - The blue (respectively red) junctions between neighboring cells correspond to spatially coherent (resp. non-coherent) labels.

an original parallelization scheme is proposed, relying on the two following assumptions:

- 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 cannot be used for an other building),
- the point labels originally projected in the 2D-grid G are of quality, *i.e.* they are probably correct (see Fig. 3).

Thus, the configuration space can be significantly reduced by considering the minimization of U as a set of N local independent (and thus parallelizable) energy minimization problems over a partition of the grid $G = \bigcup_{k\in[1,N+1]} G_k$:

$$\min_{l\in L} U(l) \Leftrightarrow \left\{ \min_{l_{/G_k}\in L_k} U(l_{/G_k}) \right\}_{k\in[1,N]} \quad (6)$$

where $l_{/G_k}$ is a configuration of labels on the cluster G_k, and L_k is the subset of labels restricted to G_k in the initial configuration $l^{(0)}$ which is obtained by a basic isotropic propagation on $G^{(empty)}$ of the projected labels in $G^{(proj)}$ (see Fig. 6). The partition of G is obtained by separating the low-level urban components (*e.g.* blocks of buildings, groups of trees, etc). Each cluster G_k corresponds to the connected cells labeled as non-ground in $l^{(0)}$. G_{N+1} represents the remaining set of cells labeled as *ground*.

The α-expansion algorithm [5] 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 cells originally labeled as *empty*, and then on the remaining cells for readjusting 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.

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

	α-exp. [5]	LBP [23]	α-exp. with our parallelization scheme
Energy	2832.9	3016.6	2853.3
Time	6.7 hrs	10.3 hrs	209.3 sec

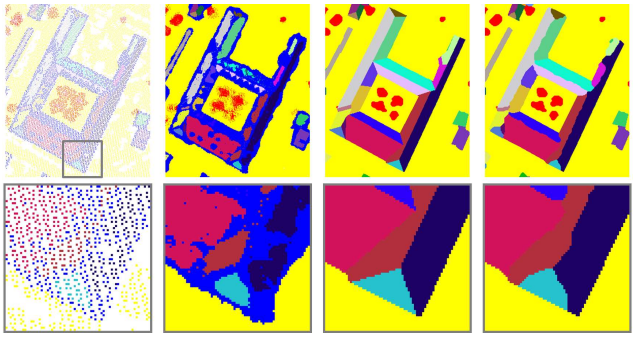


Figure 6. Planimetric arrangement - (*from left to right*) grid G of the projected point labels, initial label map $l^{(0)}$, label map after minimizing U, and after minimizing a variant of U discarding the $\bowtie$-law. One can notice the label propagation is correctly stopped beyond building contours and neighboring primitives. The $\bowtie$-law allows the optimal arrangement of the roof sections [color code: white=empty, yellow=ground, red=vegetation, blue=roof, other colors=primitives].

5. Representation of the urban elements

The three types of elements contained in the scenes are differently represented in 3D.

Buildings - A hybrid representation is used to model the buildings with a high level of generalization and 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 as polyhedral structures easily computed from the label map (see Section 4) by using the Computational Geometry Algorithms Library [1]. The mesh-patches are created by meshing a regular grid of 3D-points obtained from the cells labeled as *roof* in the label map (blue cells on the figures). As illustrated on Fig. 9, one of the main advantages of this strategy is to simplify the mesh patches while controlling the approximation error. Thus a standard mesh simplification algorithm [10] can be used to obtain more compact and coarser building representations. The fac ades are obtained by projecting the building contours on the ground.

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. 7). As directly matching 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 canopy radius are then simultaneously found by minimizing the Euclidean distance from points to an ellipse.

Ground - A meshing procedure is used to model the ground. A grid of 3D-points is created from a spatial sub-

sampling of the cells labeled as *ground*. It allows an accurate surface description without imposing geometric constraints. Note that, as with the mesh patches of the buildings and the non-planar primitives, the mesh can be simplified using a decimation algorithm [10] to gain in compaction.

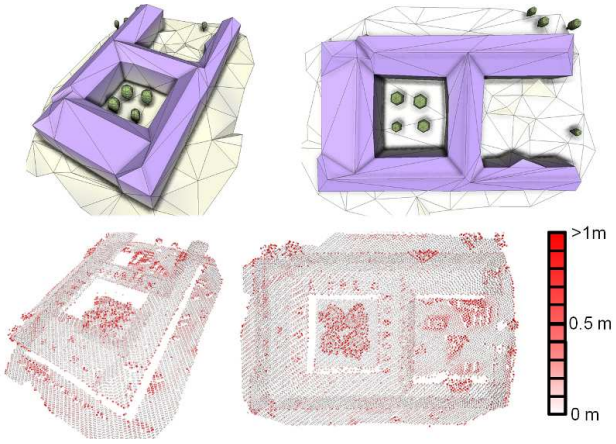


Figure 7. Object representation - (*top*) obtained 3D-model and (*bottom*) input cloud (2 pts/m^2) with the points colored according to their distance to the 3D-model. The high errors correspond to points from trees (the points of a tree do not obviously describe an perfect ellipsoidal shape), and from small urban components as cars or roof superstructures.

6. Experiments

Our method has been tested on various datasets having different point densities. One of the major strengths of the algorithm is that the point density of the data does not interfere with the planimetric arrangement in terms of result quality and computation complexity. 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 to $1.5s_c$, ϵ_1 to $0.5\epsilon_2 = 1/3$ and β to 0.5.

Visual considerations - Fig. 11 shows large scenes reconstructed with different types of landscapes including business districts with large and tall buildings, historic downtowns 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. One of the main advantages of this hybrid representation is that we are not penalized by primitive under-detection problems because the regular roof sections missed during the geometric shape extraction stage are completed by mesh-patches. The level of detail of the results mainly depends on the input point density. For example, the roof details such as the dormer-windows or chimneys in Fig. 7 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. 8, the input data has a 17 pts/m^2 density which is high enough to recover roof details such as the chimney.

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 and 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. 7 and 9). However, the mean distance is computed from all the points of the input data, including the outliers and the undesirable points corresponding to cars, fences or wires which highly corrupt the error value. Without taking these points into account, the mean error is usually inferior to 0.1 m.

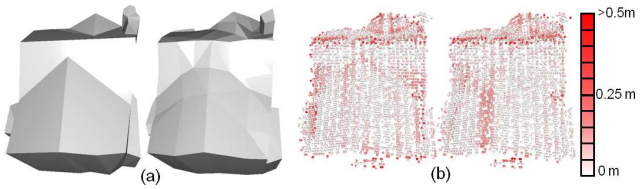


Figure 8. Comparison with a mesh simplification method- *(a)* 3D-models obtained *(left)* by our method and *(right)* by [25], *(b)* input clouds (17 pts/m^2) with the points colored according to their distance to the associated 3D-models. Our model presents a better roof component recovery. Although 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, the altimetric accuracy of the algorithm is evaluated with respect to ground truth obtained by topographical measurement on two buildings , and compared to a constructive solid geometry approach and a Digital Elevation Model from point cloud (see Fig. 10). From a 2 pts/m^2 density input data, we obtain the best mean error (0.21 m) on the evaluated buildings in spite of some high local errors on the contours (see the thin black lines partially surrounded the buildings on the altimetric error maps). From such a low point density, it is indeed difficult to accurately extract the building contours. Note also that the algorithm is globally more efficient from point clouds generated by laser acquisition than by MVS imagery, as explained in [14]. In terms of tree detection, 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 (see Fig. 11, top right crop). Less than 10 minutes is required to model a 1 km^2 dense urban area using a single computer. The computing times are competitive compared

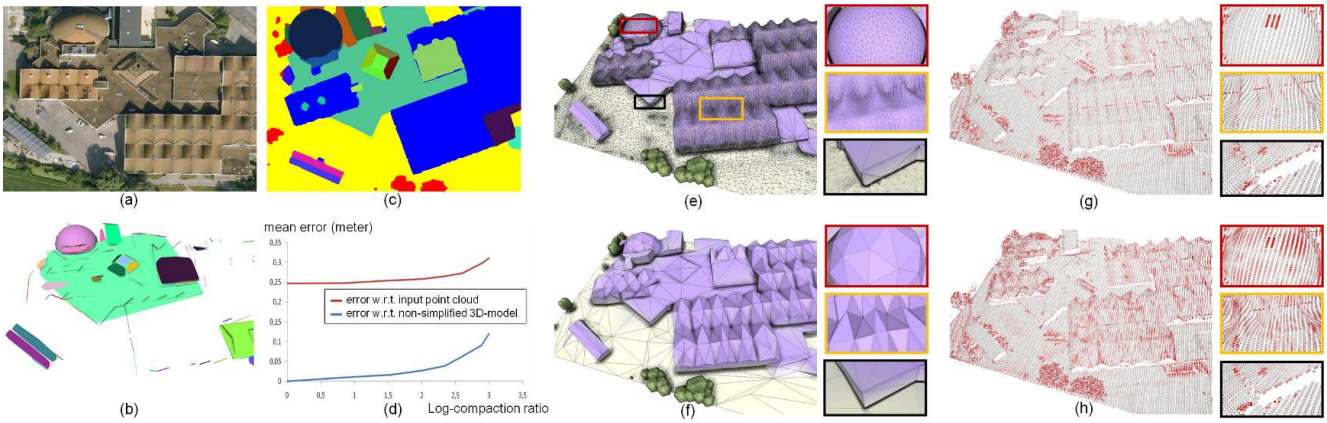


Figure 9. Hybrid reconstruction of a complex building - *(a)* aerial image, *(b)* extracted 3D-primitives, *(c)* label map [color code: see Fig. 6], 3D-models obtained with *(e)* fine and *(f)* coarse mesh patch descriptions, input cloud (2 pts/m^2) with the points colored according to their distance to the *(g)* "fine" and *(h)* "coarse" 3D-models [color code: see Fig. 7], and *(d)* 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. The interest of our hybrid representation is underlined here: 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 (resp. coarse) 3D-model has $46K$ facets (resp. 864 facets) and a 0.24 m (resp. 0.33 m) mean error to the input data .

to most of the large scale modeling algorithms (*e.g.* [19] with around half an hour per km^2, or [18] requiring several interactive operations per building).

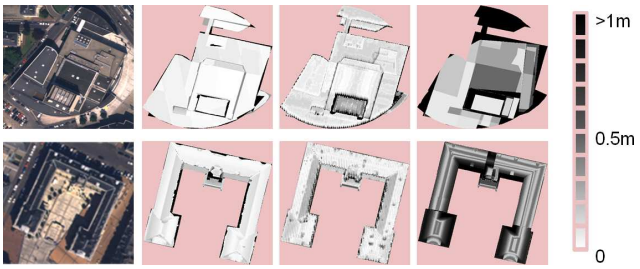


Figure 10. Altimetric accuracy w.r.t. ground truth - (*from left to right*) aerial images of two buildings, altimetric error maps for our method, for a Digital Elevation Model from point cloud and for a Constructive Solid Geometry approach [13]. See how the roof sections from our method are accurately estimated when compared to the other models.

Limitations - Firstly, some urban components are not taken into account in our representation. In particular, the bridges and the elevated roads which are locally planar structures elevated above the ground are frequently detected as buildings (see Fig. 11, 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 could 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.

7. Conclusion

We propose an original approach for modeling large urban environments from unstructured point data. One of the main strengths 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 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 could 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-based multi-view stereo [2, 7] which contain more outliers and have spatial distributions highly heterogeneous.

Acknowledgments

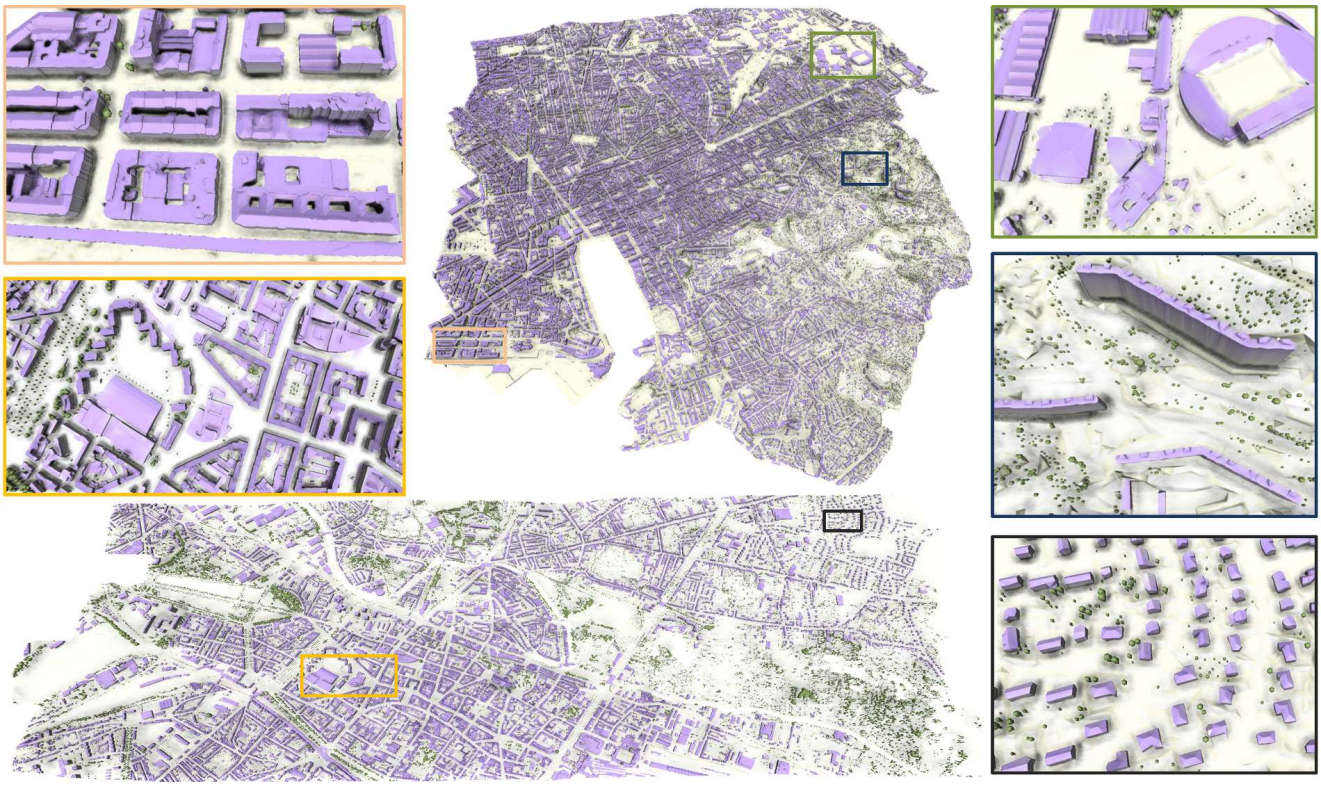
We thank the GDR Isis for the partial financial support, the reviewers for their valuable comments, Qian-Yi Zhou for the high density Lidar scan sample and the French Mapping Agency for the other datasets and ground truth.

References

[1] www.cgal.org. 5

[2] S. Agarwal, N. Snavely, I. Simon, S. Seitz, and R. Szeliski. Building Rome in a day. In *ICCV*, Kyoto, Japan, 2009. 7

[3] C. Baillard and A. Zisserman. Automatic reconstruction of piecewise planar models from multiple views. In *CVPR*, Los Alamitos, US, 1999. 3



	#input points ($\times 10^6$)	area (km^2)	altimetric variation (m)	#primitives ($\times 10^3$)	#trees ($\times 10^3$)	computing time (hour)	compaction (Mo)
Marseille, France (*top*)	38.67	19.8	192	108.6	35.7	2.52	131
Amiens, France (*bottom*)	24.52	11.57	76	56.7	22.8	1.34	93

Figure 11. Reconstruction of two large urban environments with some crops and performance statistics.

[4] 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. 1

[5] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *PAMI*, 23(11), 2001. 3, 4, 5

[6] 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. 1

[7] J.-M. Frahm et al. Building Rome on a cloudless day. In *ECCV*, Hersonissos, Greece, 2010. 7

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

[9] Y. Furukawa, B. Curless, S. Seitz, and R. Szeliski. Manhattan-world stereo. In *CVPR*, Miami, US, 2009. 1

[10] M. Garland and P. Heckbert. Surface simplification using quadric error metrics. In *SIGGRAPH*, Los Angeles, US, 1997. 5, 6

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

[12] F. Han, Z. W. Tu, and S. C. Zhu. Range image segmentation by an effective jump-diffusion method. *PAMI*, 26(9), 2004. 3

[13] F. Lafarge, X. Descombes, J. Zerubia, and M. Pierrot-Deseilligny. Structural approach for building reconstruction from a single DSM. *PAMI*, 32(1), 2010. 1, 7

[14] F. Lafarge and C. Mallet. Modeling urban landscapes from point clouds: a generic approach. Tech. Rep. 7612, INRIA, 2011. 2, 3, 6

[15] S. Li. *Markov Random Field Modeling in Image Analysis*. Springer, 2001. 2, 4

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

[17] 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. 1

[18] P. Muller, P. Wonka, S. Haegler, A. Ulmer, and L. Van Gool. Procedural modeling of buildings. In *SIGGRAPH*, Boston, 2006. 7

[19] C. Poullis and S. You. Automatic reconstruction of cities from remote sensor data. In *CVPR*, Miami, US, 2009. 1, 7

[20] R. Schnabel, R. Wahl, and R. Klein. Efficient RANSAC for point-cloud shape detection. *Computer Graphics Forum*, 26(2), 2007. 3

[21] 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. 1

[22] A. Toshev, P. Mordohai, and B. Taskar. Detecting and parsing architecture at city scale from range data. In *CVPR*, San Francisco, US, 2010. 1, 2, 3

[23] Y. Weiss and W. Freeman. On the optimality of solutions of the max-product belief propagation algorithm in arbitrary graphs. *Trans. on Information Theory*, 47(2), 2001. 4, 5

[24] L. Zebedin, J. Bauer, K. Karner, and H. Bischof. Fusion of feature- and area-based information for urban buildings modeling from aerial imagery. In *ECCV*, Marseille, France, 2008. 1, 3

[25] Q. Zhou and U. Neumann. $2.5d$ dual contouring: A robust approach to creating building models from aerial lidar point clouds. In *ECCV*, Hersonissos, Greece, 2010. 1, 6