Single View Reconstruction Using Shape Grammars for Urban Environments

Panagiotis Koutsourakis *

Laboratoire MAS, École Centrale Paris

Computer Science Department, University of Crete

Loïc Simon

Laboratoire MAS, École Centrale Paris

Olivier Teboul

Laboratoire MAS, École Centrale Paris

Microsoft France

Georgios Tziritas

Computer Science Department, University of Crete

Nikos Paragios

Laboratoire MAS, École Centrale Paris

Galen Group, INRIA Saclay, Ile-de-France

Abstract

In this paper we introduce a novel approach to single view reconstruction using shape grammars. Our approach consists in modeling architectural styles using a set of basic shapes and a set of parametric rules, corresponding to increasing levels of detail. This approach is able to model elaborate and varying architectural styles, using a tree representation of variable depth and complexity. Towards reconstruction, the parameters of the rules are optimized using image-based and architectural costs. This is done through an efficient MRF formulation based on the shape grammar itself. The resulting framework can produce precise 3D models from single views, can deal with lack of texture and the presence of occlusions and specular reflections, while maintaining the ability to cope with very complex architectural styles. Promising results demonstrate the potential of our approach.

1. Introduction

There has been an increasing attention in the use of 3D representations of urban environments during the last few years. Public authorities, museums, cinematographic studios, but also industries, providers of car navigation systems or location-based services have a keen interest. Google Earth, Live Earth and similar initiatives already offer 3D city models, but such representations are extremely simplified. This may suffice for a large proportion of buildings, but the very types of historic buildings and complex architectures would not lend themselves to such approach. Moreover the ambition of realism, many of these industries express a need in accurate reconstructions of urban environments. It is precisely the purpose of this article to demonstrate that the two objectives can be achieved together.

Many computer vision methods assume that the scene is being viewed from different view points (multiple images). Then using some basic geometric constraints, for a given 3D point being observable in different, images one can recover the notion of relative depth [7]. The scalability of these methods is questionable since it is not always reasonable to assume the existence of excellent quality multiple view points per building. At the same time traditional image derived 3D models of individual buildings or even entire cities lack structure. They typically consist of surface representations, where the fitting of planes is usually as far as the extraction of structure goes. The resulting polygon representations are voluminous, and at the same time difficult to compress. Even the best decimation algorithms will fail to maintain the relevant structures. When really pressed for storage space, as in car navigation systems, unstructured representations will not withstand the required levels of compression.

Structured representations on the other hand, can account for crucial concepts like periodicity and symmetry, so abundant in architecture. This calls for the introduction of grammars, most probably specific to each of the main architectural styles, in order to exploit such structure.

In architecture, shape grammars were successfully used for the analysis, description and construction of architectural designs [16, 5]). In computer graphics, grammar-based methods are used for the automatic generation of imaginary building designs, rather than existing ones. Procedural modeling aims at scalability while image-based reconstruction targets fine reproduction of the scene geometry. The CityEngine tool [13, 19, 11] implements a procedural generation scheme from the street-network down to single buildings . The main limitation of these methods is the fidelity to the data, since even if the generated model looks as a member of the modeled class, one has to put considerable effort in order for the model to be an accurate reconstruction of the depicted building.

---

*The first three authors have equally contributed to this work. Contact author: panagiotis.koutsourakis@ecp.fr
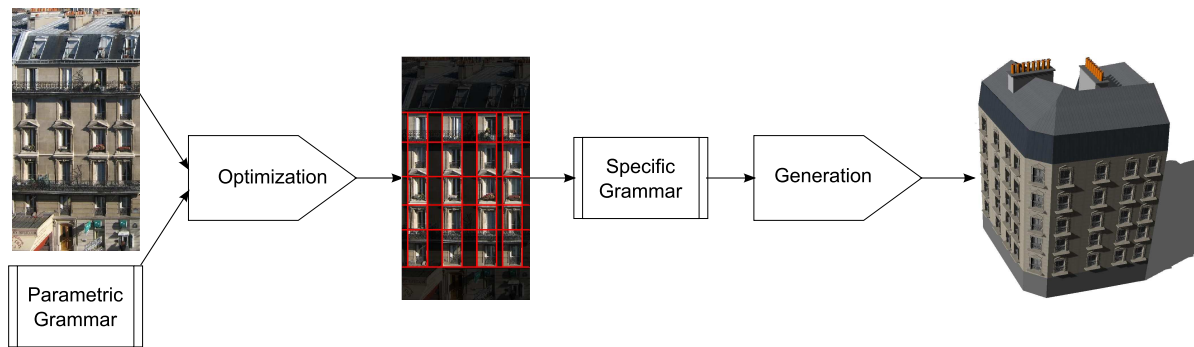
Figure 1. The 3D reconstruction pipeline : A parametric grammar is optimized with respect to the input rectified image. The resulting specific grammar is then turned into a 3D model using a procedural engine.

Ideally, one would like to fuse these two ideas towards photo/geometric-realistic 3D modeling with procedural methods. There have been few such attempts to achieve grammar-based reconstruction. For instance, in [12] the authors look for repetitive patterns in rectified image of single facades based on mutual information. In [1], the authors are using rjMCMC (reversible jump Markov Chain Monte Carlo) sampling and stochastic context-free grammars made of vertical and horizontal splits, in order to find out facade structure from images. Similarly in [14], rjMCMC is considered to explore the derivation tree, and the grammar in order to evaluate superstructures (e.g. symmetry and regularity).

In this paper we present a novel method to use the modeling procedure. The whole pipeline is depicted in Fig. 1. It uses as input a rectifed image of the building and a parametric shape grammar. The rectification can be done manually or automatically using for example [10]. When instancing the parameters one gets a specific grammar which can generate a unique 3D building. The goal is to recover the unknown parameters so that the generated building matches the image. We formulate this optimization problem within a Markov Random Field (MRF). The energy minimized by the MRF mainly relies on similarity measures between sub-regions of the facade and other low level features. In fact, none of the terms require a learning stage and so the framework is completely unsupervised. Inference is driven by the Belief Propagation (BP) algorithm guaranteeing the global optimum solution [9]. Promising results on images of complex buildings from Paris and London demonstrate the potential of our method.

The paper is organized as follows: section 2 discusses briefly the grammar-based building generation. In section 3 we describe how to build a proper MRF, in section 4 we present some results. Section 5 concludes the paper and presents future work perspectives.

2. Shape Grammars and Model Generation

In this section we describe the scheme we use for the generation of 3D models. Shape grammars have proved to be a very powerful tool that fits well with the modeling of architecture [12, 11, 19]. In this work we use a small subset of this powerful framework.

2.1. The Shape Grammar Framework

A shape grammar describes how *basic shapes* interact together through a set *replacement rules* to produce complex structured geometries. A basic shape carries semantic information and an appearance (mesh, material, etc), and can be positioned in the 3D world through a 3D transformation called a *scope* to become an *atomic shape*. A rule replaces an atomic shape LHS by a set of atomic shapes RHS, possibly under a boolean condition. We denote the rules using the following notation:

$$condition : LHS \rightarrow RHS \tag{1}$$

The RHS part of the rule is composed of atomic shapes, that are generally the outputs of procedures called *operators*. The supported operators are:

Transformation operators These are the usual 3D transformations: translation, rotation and scaling along one of the principal axes, in the local coordinate system of the atomic shape the operator is applied to. They only act on scopes.

Splitting operators These operators are called *split*, *repeat* and *mirror*. They subdivide the scope of the LHS into smaller scopes along one of the local axis and according to the specification in RHS.

Of these the split operator is the most commonly used. For example the rule $A \rightarrow \text{split}(x; B, C; 0.35, 0.65)$ splits the scope of the basic shape with symbol A with length l, along the X axis, into two pieces with lengths

$0.35l$ and $0.65l$ and assigns the symbols B and C respectively.

Component split operator This operator decomposes a 3D mesh into the set of its faces and creates one atomic shape for each of them.

Extrude operator This operator can be thought as the reverse of the component split operator. It takes a 2D polygon and creates a new atomic shape having a 3D mesh, with top and bottom faces being copies of the original polygon.

Roof operators *Mansard* or *hipped*. Just like component split and extrude, they operate on the meshes of the atomic shapes. They use the top face of the mesh of LHS and compute a weighted straight skeleton [2, 4] of the corresponding polygon. This can be easily turned into a 3D mesh of the roof.

2.2. The Generation Process

The generation process is tied to a tree structure. The nodes of this tree correspond to the atomic shapes. The process begins with a specific atomic shape called *axiom* as the root of the tree, which usually represents a polygon that is the footprint of the building. At each step of the generation one of the leaves of the tree is selected. Based on the symbol of its atomic shape, a rule is randomly selected among a set of compatible ones (i.e. rules that have the specified symbol on their LHS). When applying the rule, the resulting atomic shapes are added as children of the current leaf node and the shape tree keeps growing during the generation process. The generation ends when no more rules can be applied. Only the information at the leaves takes part in the geometry and the appearance of the final shape. The internal nodes keep track of the construction steps of the tree, but do not take part in the appearance of the shape itself.

2.3. Description of the Parametric Grammar

In this section we describe the way the facades we examine, are modeled by a parametric grammar. The grammar is actually a simple sequence of five rules: the first extrudes the footprint and puts a roof placeholder, the second turns the volume into facades. Then the remaining rules create the mansard geometry for the roof, split facades into floors and split floors into windows and wall patches. The rules are shown in equation 2. Except for the component split and the mansard rules (second and third rule), the other rules carry parameters that have to be optimized for later reconstruction:
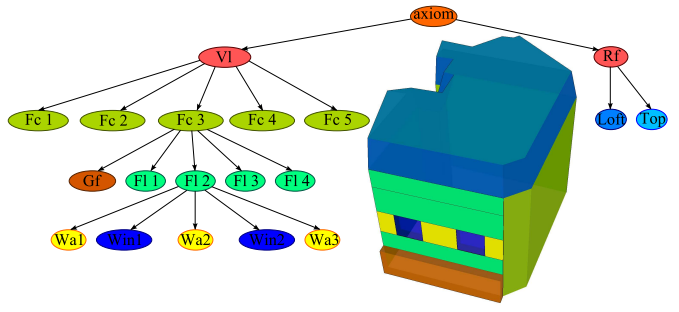

Figure 2. A partial derivation tree and the corresponding 3D model.

$$
\begin{aligned}
axiom \rightarrow &\text{scale}(z; \, Vl; \, h_b), \, \text{translate}(z; \, Rf; \, h_b) \\
Vl \rightarrow &\text{component-split}(Fc_1, \ldots, Fc_n) \\
Rf \rightarrow &\text{mansard}(Loft, Top; \, h_l, h_t) \\
Fc_i \rightarrow &\text{split}(z; \, Gf, Fl_1, \ldots Fl_m; \qquad (2) \\
& \quad h_0, h_1, \ldots h_m) \\
Fl_i \rightarrow &\text{split}(x; \, Wa_1, Win_1, \ldots, Win_k, Wa_{k+1}; \\
& \quad w'_1, w_1, \ldots, w_k, w'_{k+1})
\end{aligned}
$$

The tree corresponding to this grammar is shown in Fig. 2. The symbols appearing in the above grammar are:

- Vl the volume of the building
- Fc_i the facades
- Gf, Fl_i the ground floor and the rest of the floors.
- Wa_i, Win_i wall patches and windows.

The parameters of this grammar are the following:

- h_b, the height of the building.
- h_0 and $h_1, \ldots, h_n$, the height of the ground floor and the n floors of the facade.
- w'_i and w_i, for $1 \le i \le k$, the width of the wall patches and the width of the windows.

The procedure of optimization of these variables based on image measurements is described in section 3.

3. Grammar MRF

As discussed in the previous section, a facade can be expressed using a generic shape grammar. A specific choice of the parameters of these generic rules leads to a specific facade. A facade can be represented as a sequence of three rules, an extrusion, a vertical split and an horizontal split.
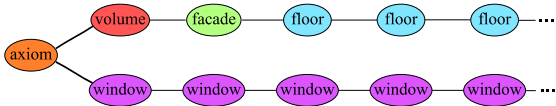
Figure 3. The MRF structure for optimization. Using a grammar enables us to decouple the optimization of the horizontal split with the optimization of the vertical one. Thus the grammar represents a huge dimensionality gain.

Modeling the facade with a grammar has three main advantages for automatic reconstruction: structure, flexibility and dimensionality reduction. The first one is the capability to intrinsically handle repetitions and well-defined structures. Therefore the result of optimizing the grammar parameters with respect to the image, will always produce a well-defined building. As for flexibility, shape grammar allows to handle irregular facade rhythms in both directions. Unlike [12], we are not limited to search for regular repetitions of a unique pattern. Finally, according to the grammar, some rules are completely independent from one another. This is the case of the two splits. Consequently, these rules should be optimized independently. The grammar reduces tremendously the complexity of our problem by intrinsically decoupling the optimization.

3.1. MRF Formulation

A graphical model such as Markov Random Field turns out to be an appropriate way of formulating our problem. Each rule can be seen as a simple MRF in which the nodes are the basic shapes lying on the right-hand side of the rule definition. The MRFs of two dependent rules are linked together. Fig. 3 shows the structure of the final MRF. The exact geometry of an element is unknown. The edges of the graph represent grammatical constraints that directly come from the operators. Sections 3.2, 3.3, 3.4 explain in detail how to build these MRFs.

Each node q is a basic shape of the grammar with a defined semantics and an undefined geometry. Therefore, the label x_q of node q represents the scope of the atomic shape in the relevant direction of the operator. For instance, a vertical split will only take care of the vertical position and the vertical dimension of the scopes. Those labels are chosen among a label set $\mathcal{L}_i = (l_1, \ldots, l_k)$. To each node q we associate a cost for each label $D_q(x_q)$ or *singleton potential*. To each pair of nodes q and r linked by an edge and for every possible pair of labels x_q and x_r we associate a cost $V_{qr}(x_q, x_r)$ or *pairwise potential*, that measures the appropriateness of a joint configuration. Consequently, for each labeling $l = (x_1, \ldots, x_n)$ of the n nodes of the graph, we can associate the cost :

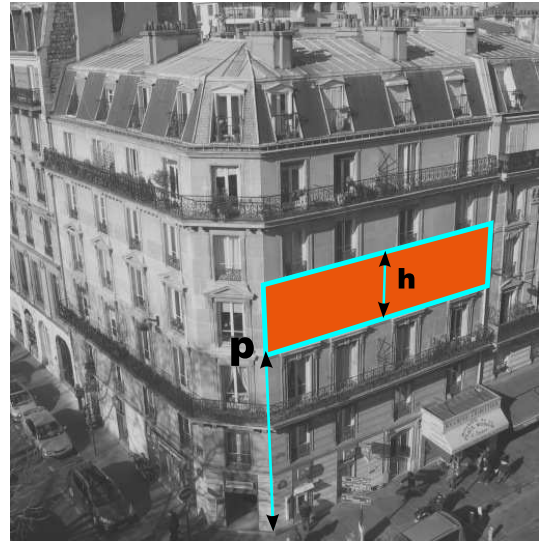$$E(l) = \sum_q D_q(x_q) + \sum_{qr} V_{qr}(x_q, x_r) \quad (3)$$

Figure 4. Definition of the labels : height and position of the floor.

Minimizing this energy with respect to l provides the labeling l^* that represents the optimal choice for the parameters of the rules to be applied on the axiom, in order to generate the building seen in the image.

$$l^* = arg \min_l E(l) \quad (4)$$

The next sections explains how to build the MRF for each rule. Before entering into the details, one might wonder why not to use the derivation tree defined by the grammar as a MRF (such as in Fig. 2). Actually such a graph does not take advantage at all of the grammar's assets. The graph defines a fixed topology (even though the number of floors may vary for instance which is all but flexible. Besides, such a graph faces the curse of dimensionality. The number of possible scopes of the leaf nodes (the window for instance) is as big as the number of instances of buildings that can be generated by the generic grammar. While appealing the derivation tree is not a good candidate to define the MRF.

3.2. Vertical Split Operator

Graph description: Each node at this level represents a floor. Two adjacent nodes are linked by an edge, so that the graph is reduced to a chain. The label of a floor is a pair (p, h) that represents the position and the height of the floor (see Fig. 4). We allow a floor to be of height zero, which means that it does not exist. This way we can handle buildings with different numbers of floors with the same graph topology. Each label (p, h) can be associated to a part of the image by projecting the procedural model onto the image using the rectification homography. (see Fig. 4).

Pair-wise potentials: They ensure the consistency between the operator and the image. The first term in equation 5 enforces a natural ordering constraint so that a floor sticks to the bottom of the next one. The second term pushes non existing floors at the end of the chain so as to avoid degenerated configurations. Eventually, the last one ensures both visual and geometric similarities of consecutive floors.

$$V_{q,r}(l_a, l_b) = \begin{cases} \infty & p_b - p_a - h_a \neq 0 \\ \infty & h_b > 0, h_a = 0 \\ \alpha S(l_a, l_b) + \beta|h_b - h_a| & h_a > 0, h_b > 0 \end{cases} \quad (5)$$

For $S(l_a, l_b)$ we use the *sum of square differences* (SSD) between sub images as a simple similarity measure. If the two compared floors share the same width W, but are of different heights h_a and h_b, then we only compute the SSD on the overlapping parts, and normalize this measure by the size of the overlap (see equation 6). The SSD is normalized for comparison purpose between varying size of sub images. The graph structure ensures a smooth and feasible configuration. It might be the case that two floors have locally low SSD value, but the associated configuration is very unlikely.

$$\begin{aligned} S(l_a, l_b) &= S((p_a, h_a), (p_b, h_b)) \\ &= \frac{\sum_i \sum_j (\mathcal{I}_{i,p_a+j} - \mathcal{I}_{i,p_b+j})^2}{W \times \min(h_a, h_b)} \end{aligned} \quad (6)$$

Singleton Potential: it tends to attract the boundaries of the semantic elements to the boundaries of the image. It can easily be modeled as a singleton potential as in equation (7). $\mathcal{B}$ can be a non increasing function of $||\nabla I||$ or the result of a pre-segmentation like the one in section 3.4.

$$D_q(l) = D_q((p, h)) = \alpha \frac{\sum_{j=1}^{W} \mathcal{B}(p, j)}{W} \quad (7)$$

Based on this model, we discretize the facade with a step of 10cm, and we consider ranges of floor heights between 2.4m and 4.2m. For a building with a height of 18m (typical in Paris), we have about 3000 labels. Due to the floor contingency, the chain is typically of length 8, since we cannot expect more than height floors on these kinds of buildings.

3.3. Horizontal Split Operator

For horizontal split, we could use the exact same graph as in section 3.2. However, we have decided to change it towards being closer to the architectural design. We globally keep the same structure, but now a node represents a vertical stripe containing a window, and therefore the space between two nodes can be non zero. As a consequence,

the first case of equation 5 has to be relaxed by turning the equality into an inequality.

As a gap might exist between two windows, we can estimate the appropriateness of it by adding an optional pairwise term. This term forces the gap regions (the walls) to be homogeneous and symmetric (see equation 8)

$$G_{q,r}(l_a, l_b) = \sum_i^{\frac{w_a + p_a + p_b}{2}} \sum_j |\mathcal{I}_{i,j} - \mathcal{I}_{p_b - i,j}|^2 + \text{var}(\mathcal{R}) \quad (8)$$

where $\mathcal{I}$ is the image, $\mathcal{R}$ the region between the two windows and $\text{var}(\mathcal{R})$ its variance.

3.4. Building Extrusion

The height of the building has to be determined. The image is not fully calibrated, therefore the solution will be defined up to a scale factor.

In order to do that we use the result of the segmentation obtained by running the algorithm described in [6]. We expect that in that image the different regions that are interesting for us, i.e. sky, roof and main building, will contain different labels. Due to noise, and the presence of different features (windows, balconies etc) we cannot expect the regions to be homogeneous. We tackle this problem by comparing the histograms of two consecutive rectangular regions.

The metric we used was the Earth Mover's Distance (EMD for short) as described in [15]. We are searching for an h^* such that:

$$h^* = arg \max_h EMD(H(\mathcal{R}_1), H(\mathcal{R}_2)) \quad (9)$$

where H is the histogram of the image region, $\mathcal{R}_1$ is the region from the bottom to height h and $\mathcal{R}_2$ is the region from height h to the top of the image. The same process is repeated for the main building and roof separation since the histogram of the roof is assumed to be different from the histogram of the volume.

4. Experimental Validation

Once the MRF has been correctly defined, the inference is solved using Belief Propagation [9], the fast message passing algorithm which offers the great advantage of providing the global optimal solution when the graph has no loops, as in our case. The optimal labeling is then easily turned into a specific grammar, whose derivation produces the expected 3D model (see Fig. 1).

We have tested our grammar-based segmentation and reconstruction approach on a set of building images, taken from the street with a standard pocket camera. We present a selection of the results obtained from this database. It gives a good overview of the wide variety of buildings that can be

processed with a simple framework, without any machine learning techniques. For each building we depict from left to right and from top to bottom : the rectified input image, the segmentation (walls are darken), then—in some cases— a wire-frame and a textured 3D model obtained from the grammar.

This selection focuses on demonstrating the robustness of the approach with respect to illumination, occlusions, viewpoint and architectural style. As for illumination, Fig. 7 presents two London buildings under different weather conditions. The first image was taken on a cloudy day while the sun is shinning on the second one. It spite of the weather conditions and an important occlusion due to a scaffolding in the first image, the segmentation is not affected. In Fig. 5, one can notice in the first facade an important local specular lighting effect which results on a saturation of the input image. However the global optimization ensures the segmentation to be correct.

Eventually, it goes without saying that the selected buildings belong to different architectural styles resulting in quite diverse visual aspects. To be more accurate, Fig. 6 and Fig. 5 show different facades from Paris. The two buildings of Fig. 6 as well as the two first ones of Fig. 5 were built in the second half of the 19th century and belong to the Haussmannian style. The two last buildings in Fig. 5 were built around 150 years earlier and are typical examples of respectively Louis XIV and Louis XII styles. Finally, the two last buildings in Fig. 7 are taken from London and belong to the Victorian style. One should also notice the different rhythms of facade. For instance, in the second image of Fig. 6, the windows are repeated regularly whereas the first facade presents an irregular window grid.

5. Conclusion and future work

In this work, we have described a way to use shape grammars along with Markov Random Field framework in order to retrieve the semantics and the geometry of challenging facades. The outcome of this process enables us to bridge the gap between 2D segmentation and 3D reconstruction from a single view.

The proposed method inherits three main limitations: interactions between the grammar elements are constrained to be on the pair level, the nature of the process is static and simplistic features being used in the inference process that are specific to the architecture. The first limitation can be addressed with the use of higher order MRFs [8]. In order to address the two remaining limitations we have adopted a different theoretical framework. To this end, image features are now obtained through responses of trained classifiers using machine learning techniques while instead of using an MRF framework that requires static graph, we proposed a random walk approach which optimizes both the sequence of derivation rules and their corresponding parameters [18].
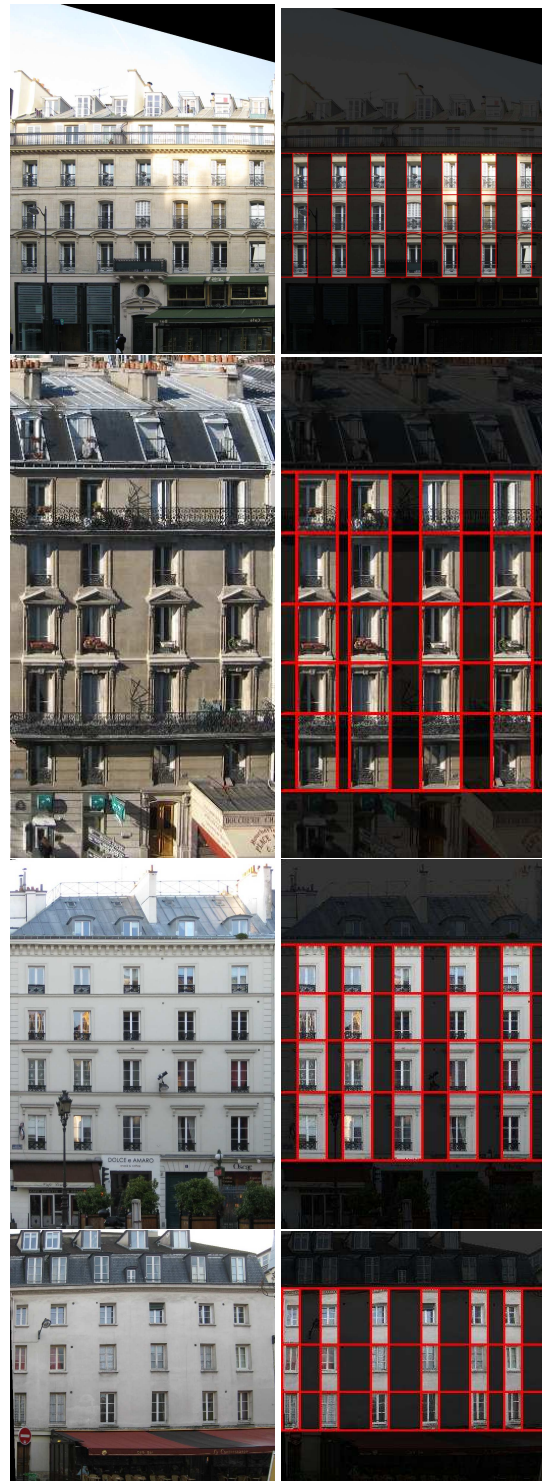

Figure 5. Segmentation of Parisian facades from different styles, built between 1640 and 1860. Note the important specular effect on the first facade and the irregular rhythm of the windows on the second and the last one.

Besides, the use of more efficient optimization methods, adapted to the dynamic nature of our problem like evolutionary computational algorithms [3], or multi-armed bandits [17] could greatly improve and enhance the applicability of the method. Furthermore, inferring/customizing specific grammars from images is a challenging direction. The idea will be to automatically determine the set of basic shapes and the set of rules being able to described a specific architecture using a set of training examples coming from the same architectural class and a generic shape grammar. Such a direction could enormously impact the applicability of shape grammars for 3D modeling and reconstruction. Ultimately, we will look into large scale reconstruction of entire urban environments. Once one has such a complete database of 3D models, many applications can be envisioned such as scene generation and understanding, image geo-tagging, or rich texture mapping on the fly, with numerous applications fields, like cinematography, navigation, games, etc.

References

[1] F. Alegre and F. Dellaert. A probabilistic approach to the semantic interpretation of building facades - georgia tech's institutional repository. Technical report, Georgia Institute of Technology, 2004.

[2] F. Aurenhammer, D. Alberts, and B. G. Artner. A novel type of skeleton for polygons. *Journal of Universal Computer Science*, 1(12):752–761, 1995.

[3] T. Bäck, D. Fogel, and Z. Michalewicz. *Handbook of evolutionary computation*. Oxford Univ. Press, 1997.

[4] J. Erickson. Raising roofs, crashing cycles, and playing pool: Applications of a data structure for finding pairwise interactions. In *Proc. 14th Annu. ACM Sympos. Comput. Geom*, pages 58–67, 1999.

[5] J. Gips. *Shape Grammars and Their Uses*. Birkhäuser, 1975.

[6] E. Grinias. *Bayesian Flooding for Image and Video Segmentation*. PhD in Computer science, Computer Science Department, University of Crete, P.O.Box 2208, Heraklion, Crete, GR-714 09 GREECE, 2009.
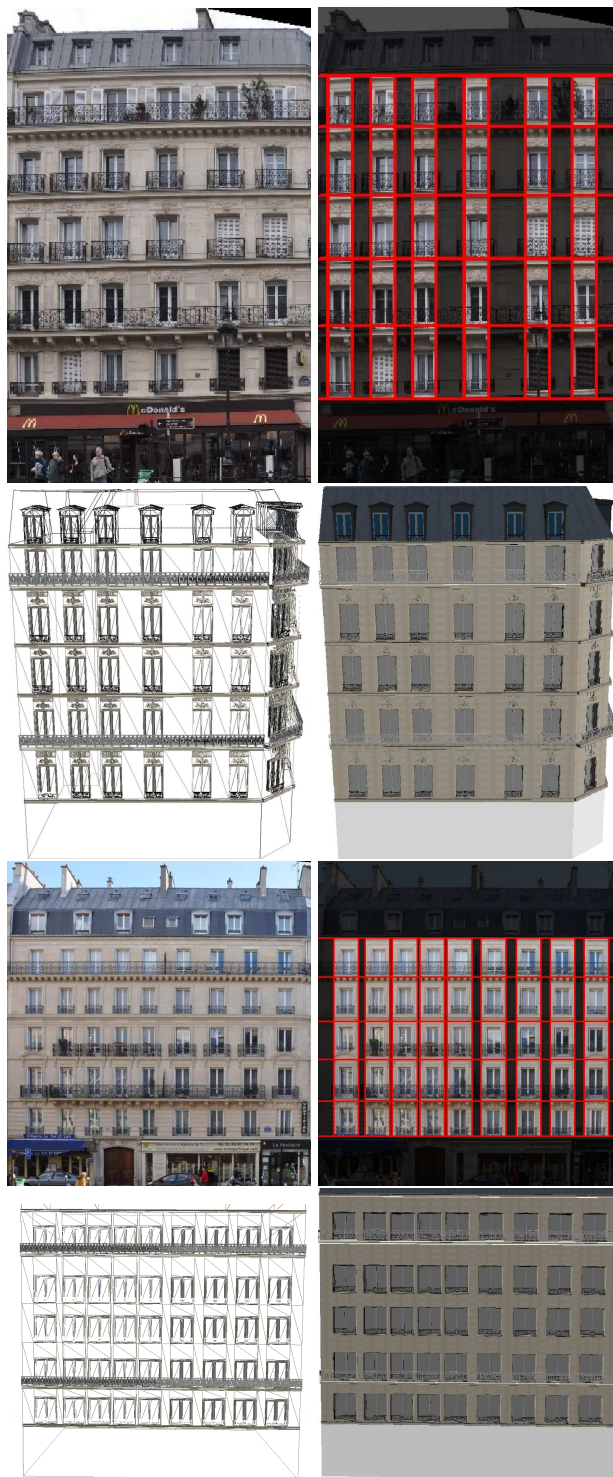
Figure 6. Segmentations and reconstructions using Grammar MRF on two Parisian Haussmannian facades.

[7] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, Cambridge, UK, second edition, 2003.
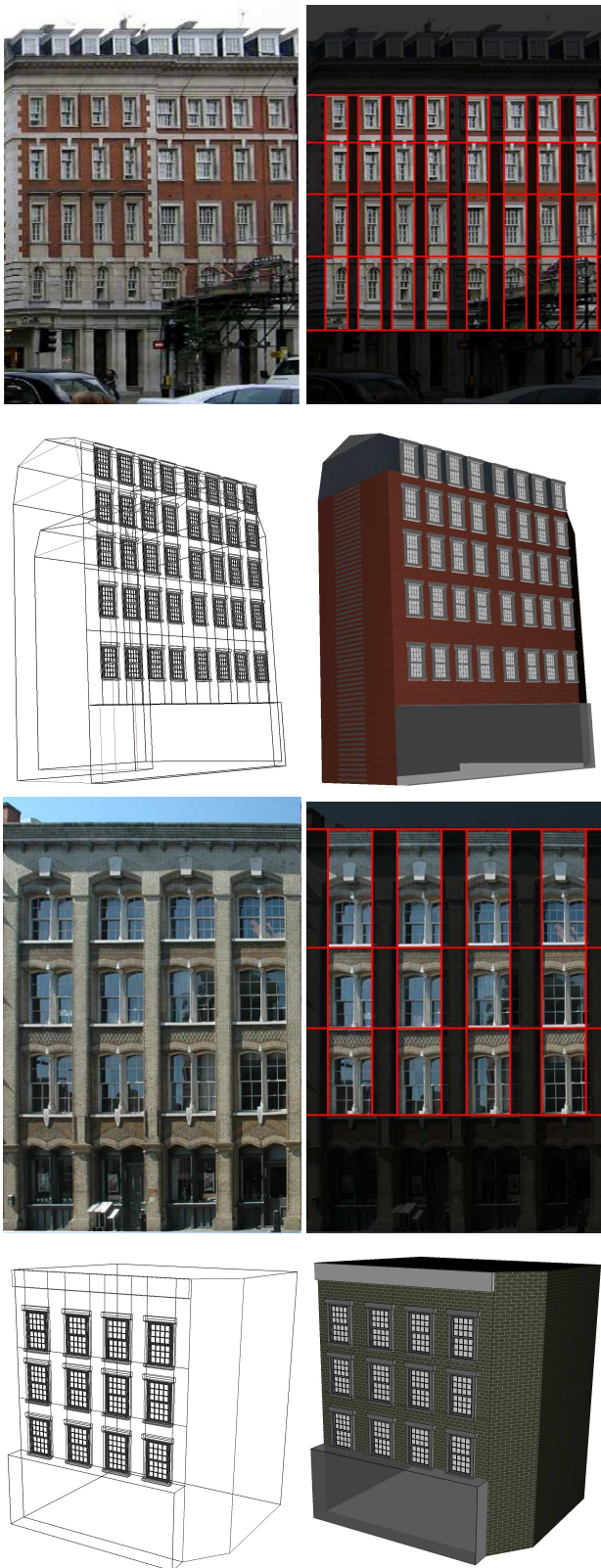
Figure 7. Segmentations and reconstructions of London buildings from Victorian style. Note the occlusion in the first image.

[8] N. Komodakis and N. Paragios. Beyond pairwise energies: Efficient optimization for higher-order mrfs. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*. IEEE Computer Society, 2009.

[9] F. R. Kschischang, B. J. Frey, and H. andrea Loeliger. Factor graphs and the sum-product algorithm. *IEEE Transactions on Information Theory*, 47:498–519, 2001.

[10] D. Liebowitz and A. Zisserman. Metric rectification for perspective images of planes. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pages 482–488, June 1998.

[11] P. Müller, P. Wonka, S. Haegler, A. Ulmer, and L. V. Gool. Procedural modeling of buildings. *Proceedings of ACM SIGGRAPH 2006 / ACM Transactions on Graphics*, 25(3):614–623, 2006.

[12] P. Müller, G. Zeng, P. Wonka, and L. V. Gool. Image-based procedural modeling of facades. *Proceedings of ACM SIGGRAPH 2007 / ACM Transactions on Graphics*, 26(3), 2007.

[13] Y. Parish and P. Müller. Procedural modeling of cities. In *SIGGRAPH '01: Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 301–308. ACM Press, 2001.

[14] N. Ripperda and C. Brenner. Data driven rule proposal for grammar based facade reconstruction. In *PIA07*, pages 1–6, 2007.

[15] Y. Rubner, C. Tomasi, and L. J. Guibas. A metric for distributions with applications to image databases. In *ICCV '98: Proceedings of the Sixth International Conference on Computer Vision*, Washington, DC, USA, 1998. IEEE Computer Society.

[16] G. Stiny. *Pictorial and Formal Aspects of Shape and Shape Grammars*. Birkhäuser, 1975.

[17] R. S. Sutton and A. G. Barto. Reinforcement learning i: Introduction, 1998.

[18] O. Teboul, L. Simon, P. Koutsourakis, and N. Paragios. Procedural modeling and image-based 3d reconstruction of complex architectures through random walks. Technical report, Ecole Centrale Paris, September 2009. International Journal of Computer Vision (submitted), 2009.

[19] P. Wonka, M. Wimmer, F. Sillion, and W. Ribarsky. Instant architecture. *ACM Transactions on Graphics*, 22(4):669–677, July 2003.