A Weak Structure Model for Regular Pattern Recognition Applied to Facade Images

Radim Tyleček and Radim Šára

Center for Machine Perception
Faculty of Electrical Engineering,
Czech Technical University,
Prague, Czech Republic

Abstract. We propose a novel method for recognition of structured images and demonstrate it on detection of windows in facade images. Given an ability to obtain local low-level data evidence on primitive elements of a structure (like window in a facade image), we determine their most probable number, attribute values (location, size) and neighborhood relation. The embedded structure is weakly modeled by pair-wise attribute constraints, which allow structure and attribute constraints to mutually support each other. We use a very general framework of reversible jump MCMC, which allows simple implementation of a specific structure model and plug-in of almost arbitrary element classifiers. The MC controls the classifier by prescribing it "where to look", without wasting too much time on unpromising locations.
We have chosen the domain of window recognition in facade images to demonstrate that the result is an efficient algorithm achieving performance of other strongly informed methods for regular structures like grids, while our general model covers loosely regular configurations as well.

1 Introduction

Recent development in construction of virtual worlds like Google Earth or Bing Maps 3D heads toward higher level of detail and fidelity. Popularity of application such as Street View shows that reconstruction of urban environments plays an important role in this area. While acquisition of extensive data in high resolution for this purpose is feasible today, their automated processing is now the limiting factor for delivering more realistic experience and it is a task for computer vision at the same time. In urban settings, typical acquired data are images of buildings' facades and their interpretation can help discover 3D structure and reduce the complexity of the resulting model; for example, it would allow going beyond planar assumptions in dense street view reconstruction presented by [1]. Complexity is particularly important when the representation has to scale with the size of cities in applications such as [2] who plan to combine range data with images. The work of [3] dealing directly with structural regularity in 3D data also supports our ideas.

While facades as man-made scenes exhibit intensive regularity and structure when compared to arbitrary natural scenes, they still present a great variety of styles, configurations and appearance. The design of a general facade model that is able to cover their range is thus a challenging problem, and several approaches have been proposed to deal with it.

Shape grammars, as introduced in [4] and later picked up by [5], are the basic essence for all recent methods based on procedural modeling to overcome the limitations of traditional segmentation techniques. The idea of shape grammars is that image can be explained by combining rules and symbols.

Some aspects of probabilistic approach were first discussed in [6], including the use of Reversible Jump Markov Chain Monte Carlo (RJMCMC). The proposed grammar is simple, based on splitting and the results are demonstrated for highly regular facades only. In a similar fashion [7] determines the structure by splitting facade to a regular grid of individual tiles and subdividing them. Meyer and Reznik [8] presented a pipeline for multi-view interpretation, where heuristics based on interest points were designed to detect positions of windows, and subsequently used MCMC to localize their borders. Ripperda [9] has designed a comprehensive dictionary of rules, on which the proposed method substantially depends; the results presented on simple facades show this approach has difficulty to achieve good localization.

The most recent method of [10] combines trained randomized forest classifiers with shape grammar to segment Haussmannian facades into eight classes. Their model assumes windows form a grid while allowing different intervals. In the second step, positions of rows and columns are stochastically estimated by a specific random walk algorithm that does not propose dimension changes. They evaluated their results quantitatively on a limited dataset of Haussmannian facades in Paris which is available online.

The majority of the mentioned algorithms for single-view facade interpretation work with hard constraint on grid configurations of windows and employ strong domain-specific heuristics. Additionally, they require user design of specific grammar or training, while both processes are prone to overfitting. Our contribution is in the design of segmentation framework with the following properties:

- a general model allows a simple implementation avoiding strong domain specific heuristics,
- structure is not modeled by a global grid, but softly by local pair-wise constraints, allowing loosely regular configurations,
- different element classifiers can be conveniently plugged in,
- efficient interpretation is achieved as the classifier is guided by the sampler and need not even visit all image pixels in practice,
- the number, spacing and exact size of facade elements need not to be known in advance and does not rely on preprocessing that can fail i.e. in irregular cases like in Fig. 4.

Since windows are the most prominent elements of a facade, we choose detection of window-like image elements to be the target of this paper.

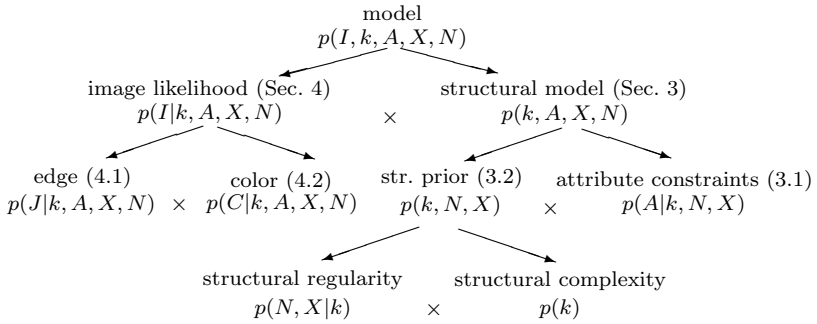


Fig. 1. Hierarchy in probability model, numbers in brackets are section references.

2 Structural Recognition Framework

We consider the problem of recognizing elements in an image, like windows in a facade. Our model parameters (variables) consist of complexity k (the number of windows), shape attributes A (i.e. size, aspect), location attributes X (window center locations) and element neighborhood relation N. The recognition task can then be formulated as follows: Given image data I, we search for model parameters $\theta = (k, A, X, N)$ by finding the mode of the following joint distribution $p(I, \theta)$

$$\theta^* = \arg\max_{\theta} p(I|\theta)p(\theta), \tag{1}$$

which is computed with Bayes theorem from data likelihood $p(I|\theta)$ and structural model prior $p(\theta)$. We will decompose our probability model hierarchically as shown in Fig. 1 and propose pdfs specific for the task of window detection in facade images. Then we can apply stochastic RJMCMC framework to find the optimal value θ^* by effectively sampling from the space of possible combinations of parameters θ. More details on its implementation will be given in the following sections.

3 Structural Model

The structural model is based on pair-wise element neighborhood and attribute constraints, yielding bottom-up approach. We are given a set of $k \in \mathrm{N}$ element locations $X = \left\{x_i \in \mathrm{R}^2; i = 1, \ldots, k\right\}$. Our neighborhood representation is based on a planar graph $G(X) = \{V(X), D(X)\}$, where vertices $V(X) = \{v_i;\ i = 1, \ldots, k\}$ correspond to elements and edges $D(X) = \{(u, v);\ u, v \in V(X)\}$ to relative neighborhood relationship between them.

Since we are dealing with image elements attributed by their locations X in image plane, we can limit the edge set $D(X)$ to a reasonable planar subgraph

and *Relative Neighborhood Graph* (RNG) turns out to be a natural choice [11]. It is defined by the following condition: Two points u and v are connected by an edge whenever there does not exist a third point r that is closer to both u and v than they are to each other (in Euclidean metric). It is known that RNG is a unique subgraph of *Delaunay Triangulation* (DT), and can be computed from it efficiently, in $O(n)$ time. This choice defines a function $X \mapsto G(X)$, where the graph is uniquely constructed from a set of element locations X.

We define neighbors as elements that are in immediate proximity of each other and such that they share some attributes. This neighborhood N is to be recovered as a part of the solution, and we represent it by binary labels $N = \{l_{uv} \in \{0,1\}\,;\, (u,v) \in D(X)\}$ for edges indicating mutual neighborhood of two elements when $l_{uv} = 1$. Such two elements are then members of the same structural component, where all connected elements are related by attribute similarity constraints. Labels $l_{uv} = 0$ allow the existence of dissimilar elements in proximity of each other.

An edge (u, v) has an orientation attribute $o_{uv} \in \{\mathtt{h}, \mathtt{v}\}$, which is a function of locations x_u, x_v of elements on its endpoints. It is given by the angle ψ between vertical direction and line connecting element locations. The case of $|\psi| < \frac{\pi}{4}$ determines vertical orientation (h), the other case is horizontal (v). This choice defines a function $D(X) \mapsto \{\mathtt{h}, \mathtt{v}\}$.

The prior probability model $p(k, N, X, A) = p(A|k, N, X)p(k, N, X)$ splits into attribute constraints $p(A|k, N, X)$ and structure prior $p(k, N, X)$. The parameters of the underlying distributions were chosen empirically.

3.1 Attribute Constraints

The attribute constraints evaluate the similarity of two neighboring elements (in terms of N); such attributes can be shape or appearance.

For facades, we assume our elements can be represented by a rectangular shape template with its borders parallel to image borders. The shape attributes $A = \{W, H, T\} = \{(w_i, h_i, t_i)\,; i = 1, \ldots, k\}$ are described in Fig. 2 and the column width $t_i = t$ is given and fixed. Our attribute constraints will then

Fig. 2. *Left:* Window shape template is parametrized by its width $w_i \in (0, 1)$, height $h_i \in (0, 1)$, both relative to image height I_h, and the width of the central column $t_i \in (0, 1)$ relative to the window width. *Right:* Shape template (red) is matched with image edges (blue).

reflect the fact neighboring windows most probably have the same dimensions. We start by decomposition

$$p(A|k,N,X) = p(W|H,k,N,X)p(H|k,N,X)1(A|X), \tag{2}$$

where $p(W|H,k,N,X) = \prod_{i=1}^{k} p(w_i|h_i)$ is the aspect ratio with distribution $p(w_i|h_i) = \beta(\frac{w_i}{w_i+h_i}, \alpha_r, \beta_r)$. When any of the windows overlap with another, we set unit function $1(A|X) = 0$, effectively avoiding such window configuration.

To model constraints on heights H, we introduce a set of latent variables h_c, one for each component c of graph $G(X)$ with neighborhood N. The height similarity within components is enforced in

$$p(H|k,N,X) = \prod_{c} \left(p(h_c) \prod_{i \in V_c} p(h_i|h_c) \right), \tag{3}$$

where c is from the set of all components, V_c is the set of windows in the component c and $p(h_c) = \beta(h_c, \alpha_h, \beta_h)$ is the common height prior. Each height in a component c should be most probably equal to h_c, which is expressed by $p(h_i|h_c) = \mathcal{N}(h_i - h_c, 0, \sigma_h)$.

3.2 Structural Prior

The structure prior $p(k,N,X) = p(N,X|k)p(k)$ combines structural regularity $p(N,X|k)$ and complexity $p(k)$.

Structural Regularity. In order to model multiple assumptions on $p(N,X|k)$, we express it as a probability mixture [12]:

$$p(N,X|k) = \omega_1 p_a(X|N)p(N) + \omega_2 p_s(X|N)p(N) + \omega_3 p_c(N|X)p(X), \tag{4}$$

where $\sum_{i=1}^{k} \omega_i = 1, \omega_{123} = \frac{1}{3}$ and k was omitted in $p(\cdot)$ for simplicity. We assume element locations in $p(X)$ are mutually independent and uniformly distributed in image. The neighborhood prior $p\,(N) = \prod_{(u,v)} p(l_{uv})$ takes into account the possibility of suppressing an edge where $p(l_{uv} = 0) = p_{sup}$, $p(l_{uv} = 1) = 1 - p_{sup}$ and $p_{sup} = 0.01$ is the probability of a suppressed edge.

Alignment. The first assumption on the position of elements is that neighboring elements should be horizontally or vertically aligned. We model this by measuring angles $\varphi(x_u, x_v) \in (-\frac{\pi}{4}, \frac{\pi}{4})$ between the line connecting element locations $x_u x_v$ and horizontal ($o_{uv} = \texttt{h}$) resp. vertical ($o_{uv} = \texttt{v}$) direction, and express them in

$$p_a(X|N) = \prod_{(u,v) \in D(X)} p(x_u, x_v|l_{uv}), \tag{5}$$

where $p(x_u, x_v|l_{uv} = 1) = \beta(\varphi'(x_u, x_v), \beta_\varphi, \beta_\varphi)$, $\beta_\varphi = 50$ and $\varphi'(x_u, x_v) = \frac{2}{\pi}(\varphi_{uv} + \frac{\pi}{4}) \in (0,1)$ is the angle normalized to unit interval. The probability in the case of a suppressed edge is $p(x_u, x_v|l_{uv} = 0) = p_{a0}$.

Spacing. The second assumption is that the distance between elements in a horizontal or vertical neighborhood should most probably be equal. We model this by comparing distances to horizontal and vertical neighbors in

$$p_s(X|N) = \prod_{(u,v,z)\in D^2(X)} p(x_u, x_v, x_z | l_{uv}, l_{vz}) \tag{6}$$

where (u, v, z) denotes a pair of edges $(u, v), (v, z), u \neq z$ with the common vertex v and the same orientation. The distance term is expressed by $p(x_u, x_v, x_z | l_{uv} = l_{vz} = 1) = \beta(\frac{\Delta_{uv}}{\Delta_{uv}+\Delta_{vz}}, \beta_\Delta, \beta_\Delta)$, where $\beta_\Delta = 50$ and $\Delta_{uv} = |x_u - x_v|$ are distances to the neighbors. As in the previous case, the probability in the cases with any suppressed edge is $p(x_u, x_v, x_z | l_{uv} \neq 1 \vee l_{vz} \neq 1) = p_{s0}$.

Configurations. We model higher-order dependencies in the structure configurations with

$$p_c(N|X) = \prod_{i=1}^{k} p(l_{ij} | (i, j) \in D(X)), \tag{7}$$

where the probabilities $p(l_{ij}|(i, j) \in D(X))$ model the expected degree of a given vertex i, including orientation of edges (i, j) connected to it, i.e. the typical grid configuration is to have two vertical and two horizontal edges incident with vertex i.

With the grid assumption and the window size prior, we can estimate the number of rows $m = \frac{1}{2\mu_h}$ and columns $n = \frac{1}{2\mu_h r_h}$, assuming the space between the windows to be equal to the window size. This heuristic plays only a minor role in our model and helps us to derive the vertex configuration probability $p(l_{ij}|(i, j) \in D(X))$. It is given in Table 1, where rows and columns correspond to the number of horizontal and vertical edges connected to the window vertex. The maximum degree of a vertex in RNG is six with at most three horizontal and three vertical edges.

Table 1. Neighborhood configuration prior $p(l_{ij}|(i, j) \in D(X))$, where $deg_h(i), deg_v(i)$ are functions of neighboring labels l_{ij}. The $p_{c0} = 10^{-4}$ is the probability of a single (unstructured) window, $p_{c1} = 0.099$ is the probability of a single row or column of windows, $p_{c2} = 0.9$ is the probability of a window grid, $p_{c3} = 10^{-5}$ is the probability of more dense configurations.

$deg_h(i), deg_v(i)$	0h	1h	2h	3h
0v	p_{c0}	$\frac{1}{2}p_{c1}$	$\frac{1}{(m-2)}p_{c1}$	p_{c3}
1v	$\frac{1}{2}p_{c1}$	$\frac{1}{4}p_{c2}$	$\frac{2}{(m-2)}p_{c2}$	p_{c3}
2v	$\frac{1}{(n-2)}p_{c1}$	$\frac{2}{(n-2)}p_{c2}$	$\frac{1}{(m-2)(n-2)}p_{c2}$	p_{c3}
3v	p_{c3}	p_{c3}	p_{c3}	p_{c3}

Structural Complexity. The prior for number of elements can be modeled with Poisson distribution $p(k) = \text{Pois}(k, mn)$ based on the estimation of number of rows m and columns n given above.

4 Data Likelihood

The data likelihood $p(I|K, N, A, X)$ is solely task-specific and can be chosen arbitrarily as long as it can be evaluated by means of probability density or likelihood ratio.

In the task of window detection in facade images, the input is image $I = \{i;\, i = 1, \ldots, I_w \cdot I_h\}$ defined as a set of pixels and we assume it is rectified, i.e. the windows borders are parallel to the image borders, and I_w, I_h are image width and height.

We want to express the probability of observing image I if window parameters and structure are given. We combine two features: image edges J and color C in $p(I|k, A, X, N) = p(J|k, A, X, N)p(C|k, A, X, N)$. We use color to detect regions of interest and edge features for localization of the windows' borders.

4.1 Edge Likelihood

We assume that window borders correspond to edges, and use Canny detector to find them. However, this model will not fully hold in real world situations, when we obtain the input by detecting edges in a picture—there can be windows which do not have all pixels with underlying edges and vice versa, some edges do not belong to any windows at all. The latter case will typically prevail.

We use binary imaging model for window edges represented by oriented edge image $J = \{J_i \in \{0, 1, 2\}\,; i \in I\}$, where $J_i = 1$ if pixel i belongs to an horizontal edge detected in I (foreground), resp. $J_i = 2$ for vertical edge; otherwise $J_i = 0$ (background). We define $d(J) \in (0, 1)$ as a distance transform of the edge image J normalized by $\max(I_h, I_w)$. We use the gradient of $d(J)$ to distinguish between horizontal and vertical edges. Similarly, we introduce edge image $R(A, X)$ rendered from the current configuration specified by attributes A, X and the shape template in Fig. 2 with nearest neighbor discretization. Assuming pixel independence, we can write $p(J|A, X) = \prod_{i \in I} p(J_i|R_i(A, X))$ where the probability of observing a pixel i in the edge image J given the rendered configuration R is

$$\begin{aligned} p(J_i = 0|R_i = 0) &= p_{\text{TN}} = 1 - 2p_{\text{FN}}, \\ p(J_i \in \{1, 2\}\,|R_i = 0) &= p_{\text{FN}} = 0.1, \\ p(J_i = 0|R_i \in \{1, 2\}) &= p_{\text{FP}}(d(i))(1 - p_{\text{FX}}),\ d(i) > 0, \\ p(J_i = 1|R_i = 1) &= p(J_i = 2|R_i = 2) = p_{\text{TP}} = p_{\text{FP}}(0), \\ p(J_i = 2|R_i = 1) &= p(J_i = 1|R_i = 2) = p_{\text{FX}}, \end{aligned} \tag{8}$$

where $p_{\text{FP}}(d(i)) = \beta(d(i), \beta_{\text{FP}} = 500, 1)$ makes rectangles close to edges more probable and acts as a guide for directing the random walk. The $p_{\text{FX}} = 10^{-9}$ is

the probability assigned when the edge specified by the configuration crosses an image edge with opposite direction.

The edge likelihood can be efficiently evaluated from pre-computed integral edge images, one for each orientation, yielding constant computational complexity $O(1)$ per edge; this speed-up is possible thanks to rectified images and helps make random sampling (described in Sect. 5) very efficient.

4.2 Color Likelihood

A pixel color classifier matches the input RGB color image $C = \{c_i \in (0,1)^3; i = 1, \dots, k\}$ with a unimodal Gaussian distribution $\mathcal{N}(\bar{C}, \Sigma_C)$ for window pixels. Its mean $\bar{C} = (0.33, 0.36, 0.38) \in (0,1)^3$ and covariance Σ_C of window color were trained on a single representative facade image and correspond to dark colors; higher mean in blue channel is related to the reflection of sky in window glass. We use the classifier to segment pixels either to foreground (window) or background (non-window) sets $C_f \cup C_b = I$. Assuming pixel independence, the probability of observing segmented image is

$$p(C|A,X) = \prod_{i \in C_f} p_f(c_i|A,X) \prod_{j \in C_b} p_b(c_j|A,X), \tag{9}$$

where the foreground color model is expressed by $p_f(C_i|A,X) = \mathcal{N}(\bar{C}, \Sigma_C)$, the background probability $p_b(c_j|A,X) = p_b$ is constant and we evaluate foreground pixels only. Similarly to edge likelihood, color likelihood can be evaluated using pre-computed integral images in linear time.

5 Recognition Algorithm

We have chosen reversible jump Markov Chain Monte Carlo (RJMCMC) framework [13] that fits our task of finding the most probable interpretation of the input image in the terms of target probability $p(\theta, I)$ in (1), which has a very complex pdf as it is a joint probability of both attributes and structure. Our solution θ^* is found as the most probable parameter value the chain visits in a given number of samples.

While the MCMC algorithm is simple, we need to carefully design proposal distribution q that should approximate target distribution $p(\theta, I)$ well while it is easy to sample from it. We should point out that the quality of the resulting interpretation is determined by the probability model and the time necessary to reach the solution is influenced by the proposal distributions. It turns out that by exploiting the estimated structure we can efficiently guide the random walk of our chain by repeatedly sampling the new state θ' from the vicinity of the current state from conditional probability $q(\theta'|\theta)$.

We use an independent sampler $q(\theta|I)$ to initialize the Markov chain, which samples the initial state θ_0 either from the prior distribution $\theta \sim q(\theta)$ or exploits some image information in $\theta \sim q(\theta|I)$. This involves sampling the number

of elements $k \sim q(k)$ first and then their attribute values $(X, A) \sim q(X, A)$ independently. In practice we choose sampler to start with $k_0 = 1$.

The conditional sampler $q(\theta'|\theta, I) \to \theta'$ is a mixture of individual samplers such that each modifies a subset of parameters θ based on a specific proposal distribution $q_m(\theta'|\theta, I)$. The main sampler only chooses from $q(m)$ which of the individual samplers m will be used to propose the next move. We will now propose the set of samplers that will explore the space of parameters θ. Their design must fulfill Markov Chain properties of detailed balance and reversibility of all moves, i.e. given a move there must always exist a reverse move m', and their probability ratio must be reflected in the acceptance of Metropolis-Hastings (MH) algorithm:

$$A = \min\left\{1, \frac{p(\theta', I)}{p(\theta, I)} \cdot \frac{q(m'|\theta')}{q(m|\theta)}\right\}. \tag{10}$$

5.1 Metropolis-Hastings Moves

Moves introduced in this section do not modify the model complexity k and can be thus evaluated by a classical MH algorithm (10).

Attribute modification. This move picks up an element $i \sim \mathcal{U}(\{1, \ldots, k\})$ from discrete uniform distribution and perturbs some of its attributes values randomly. Additionally, attribute samplers can be designed to exploit image likelihood to increase the acceptance rate. In the window detection scenario, we have implemented three variants for this type of proposals:

- *Drift* - random variation of position $x'_i = x_i + \Delta$, $\Delta \sim \mathcal{N}(0, \sigma_\Delta)$ without changing the size,
- *Resize* - randomly pick up one of four window sides (left/right/top/bottom) and move it by Δ,
- *Flip* - fix one of the window sides and flip the window around it.

Element resampling. This move is a more radical variant of the previous one, we pick up an element i and change of all its attributes by sampling from the prior distribution $a'_i, x'_i \sim q(a_i, x_i)$ or $a'_i, x'_i \sim q(a_i, x_i|I)$ if possible.

Attribute constraint enforcement. This move proposes changes to the attributes according to the current neighborhood, $a'_i, x'_i \sim q(a_i, x_i|A, X, N)$. We pick up a random edge $(u, v) \sim \mathcal{U}(D(X))$ and direction ($u \Rightarrow v$ or $v \Rightarrow u$) and transfer attribute values over the edge from one element to another according to the specific constraints, i.e. $a'_u = a_v$. For facades, we transfer both position and size from one element to the other in dimension given by orientation of the connected edge, i.e. height and vertical position for horizontal edge.

Structure modification. We include move to allow changes to the neighborhood structure: it picks up a random edge $q_d \to (u, v)$ and changes its label $l'_{uv} = 1 - l_{uv}$, effectively suppressing or recovering the edge.

Proposals for latent heights h_c are performed similarly by choosing uniformly component c and then sampling $h_c \sim \mathcal{N}(\bar{h_c}, \sigma_h)$, where $\bar{h_c} = \frac{1}{|V_c|} \sum_{i \in V_c} h_i$ is the mean height in the component.

5.2 Reversible Jump Moves

We also need to find the number of elements k, that controls the dimension of parameters A, X. In order to compare the models in different dimensions, we need to define dimension matching functions $q_\to, q_\leftarrow$ for both direct and reverse moves. Then the acceptance ratio can be calculated as $A = \min\{1, \alpha\}$, where

$$\alpha = \frac{p(\theta', I)}{p(I)} \cdot \frac{q(m|\theta')}{q(m'|\theta)} \cdot \frac{q_\leftarrow(u_\leftarrow|\theta')}{q_\to(u_\to|\theta)} \cdot J_\to, \tag{11}$$

where $\to$ refers to direct move, $\leftarrow$ to reverse move, u are dimension matching variables and $J_\to = \left|\frac{\partial f_\to(\theta, u_\to)}{\partial(\theta, u_\to)}\right|$ is the Jacobian of the transformation, following the notation given in [13]. There are three moves:

Birth. By inserting a new element into our model we propose an increase of dimension $k \to k' = k + 1$. We choose the communication variables to be $u_\to = [a_*, x_*]$, where we sample the attributes of the new element $a_*, x_* \sim q(a, x)$ and obtain a new state where $A' = \{A, a_*\}$ and $X' = \{X, x_*\}$. The corresponding dimension matching function is $f_\to(A, X, u_\to) = f_\to(\{A, X\}, [a_*, x_*])$, which inserts a_* into the set, and its Jacobian $J_\to = 1$. We will use the following notation within this paper: terms in [ ] refer to communication variables and terms in { } to parameters. The reverse move is *death*, for which we have no communication variable $u_\leftarrow = [\,]$, only choose an element i to be removed from the set. To establish reversibility, we define inverse matching function as $f_\leftarrow(A', X', u_\leftarrow) = f_\leftarrow(\{A', X'\}, [\,])$, where a_i, x_i are the removed attributes and $A = A' \setminus a_i$, $X = X' \setminus x_i$. The corresponding birth move acceptance is then

$$\alpha_{birth} = \frac{p(\theta', I)}{p(I)} \frac{q(m|\theta')}{q(m'|\theta)} \cdot \frac{q(i|k')}{q(*|k)} \cdot \frac{1}{q_\to(a^*|A)} \cdot 1, \tag{12}$$

where $q_\to(a_*|A) = p(a)$ is the prior probability of the new window, $q(i|k') = \frac{1}{k'}$ and $q(*|k) = \frac{1}{k}$ are the probabilities of selecting the windows a_*, a_i.

Death. By removing an existing element from the set we propose a decrease of dimension $k \to k' = k - 1$, and choose a window $i \sim \mathcal{U}(1, k)$ to be removed. With an appropriate change of labeling, the derivation of death move will be the same as for birth, except for the inversion of ratios in (12).

Replicate. This is a special case of the *birth* jump that exploits the structure for predicting values for the new elements according to attribute constraints, which can be generally described as sampling from $a_*, x_* \sim q(a, x|N)$. For facades, we uniformly sample an edge $(u, v) \sim \mathcal{U}(D(X))$ and place the new window to the position according to $x_* = x_u + \alpha(x_v - x_u)$, where we choose $\alpha \sim \mathcal{U}\left(\left\{\frac{1}{2}, \frac{1}{3}, \frac{2}{3}, 2, -1\right\}\right)$ and calculate the new height by $h_* = \frac{1}{2}(h_u + h_v)$ and the width w_* analogically.

5.3 Convergence and Complexity

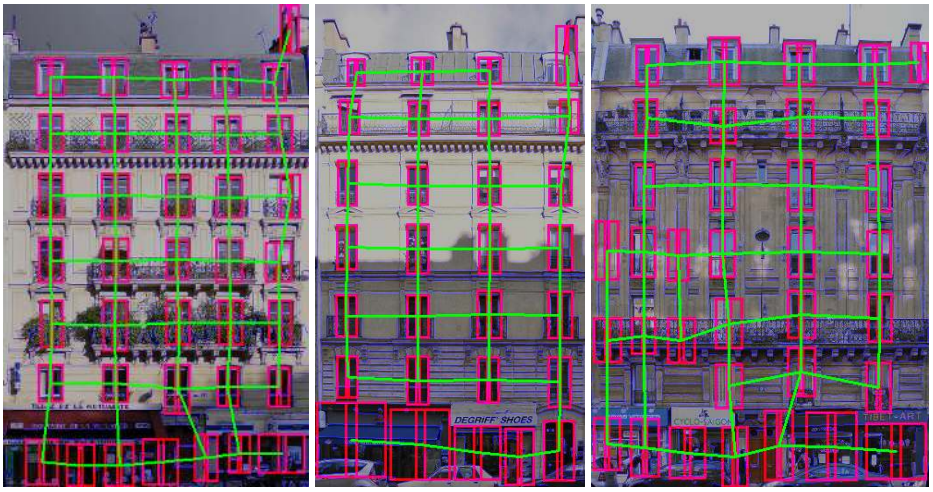
We have found that the typical necessary number of MCMC samples (classifier calls) is proportional to image size in pixels $|I|$ (from 30% for easy instances to 200% for difficult ones). This is a good news, we expected that the number will grow exponentially with scene complexity. As a result, we fixed the number of samples in our current method to a pessimistic estimate, but our experiments suggest that significantly shorter sampling time could be achieved with suitably designed stopping condition.

6 Experimental Results

We have performed a number of experiments with the implementation of window detection in facades of various styles to demonstrate the universality of our approach. We have run the Markov Chain for $5{\cdot}10^5$ iterations in our experiments, which roughly equals to visiting all pixels in the analyzed images.

Because of a very recent appearance of a first public dataset known to us with quantitative results in [10], we are among the first to compare with them. The test part of the dataset consists of 10 rectified and annotated images of facades from a street in Paris, which share attributes of Haussmannian style but differs in lightning conditions. Direct comparison is not possible, because they segment facade pixels into eight different classes of elements and our window detector defines only two (window/non-window). To deal with this issue, we have merged the columns of confusion matrix given in [10] into two, and the results are given in Table 2. All parameters of our model were fixed for this experiment, specifically the size prior was set such that the most probable relative window height is $h = 0.1$ and aspect ratio $r = 0.5$.

The numbers in Table 2 for *window* and *wall* classes show that our weak structure model slightly outperforms Procedural Segmentation (PS) framework [10]. This is clearly a success, because PS benefits from a randomized forest combining 8 classifiers, trained on 15×15 pixel patches in 20 images from the same street as the test data, and a grammar specifically designed for Haussmannian style. In contrast, our method is guided by far weaker cues: color of individual pixels, rectangular shape matching with image edges and size prior. In our case the dominant role plays the weak structural model that emerges from the data: it is able to select among objects of interest proposed by local classifiers and, at the same time, support windows completing the structure even where



a) Monge No. 13 b) Monge No. 43 c) Monge No. 50

Fig. 3. Visualization of results on part of Parisian dataset [10], facade a) is occluded by plants, in facade b) cast shadow is present. False positive windows in c) are also window-like regions: They have good response from both classifiers and match with the neighbors. Detected windows are shown in red, neighborhood edges in green and image edges are emphasized in blue. Results on the complete test set are available as supplemental material.

the classifier response is low. This allows us to achieve good results even when illumination varies and partial occlusion of windows is present, as shown in Fig. 3. Poor results of Randomized Forest (RF) segmentation from [10] included in Table 2 give an idea how entirely unstructured approaches perform on this data.

For classes different than *window* and *wall* the results cannot be directly compared with the other methods, but allow us to analyze the behavior of our method in such classes. Balconies are typically overlapping windows in Haussmannian style, but such overlaps are somehow randomly annotated as *window* or *balcony* in the ground truth [10], even when the appearance is the same, introducing some amount of ambiguity in the results. The *shop* class areas are

Table 2. Quantitative results on Haussmannian dataset [10] shown in percentage of pixels from class specified in a row. Second column displays the percentage of pixels of given class in the whole test set. RF stands for Randomized Forest, PS for Procedural Segmentation. Our window detection rate of 83% is comparable to 81% rate for PS (in bold face).

ground truth[10]		RF [10]		PS [10]		proposed		mapping of our classes	
class	area	hit	miss	hit	miss	hit	miss	*window*	*non-window*
window	11	30	70	**81**	19	**83**	17	•	
wall	48	38	62	83	17	84	16		•

a) Modern facade b) Irregular facade c) Sparse structure

Fig. 4. Results on facade images from Prague.

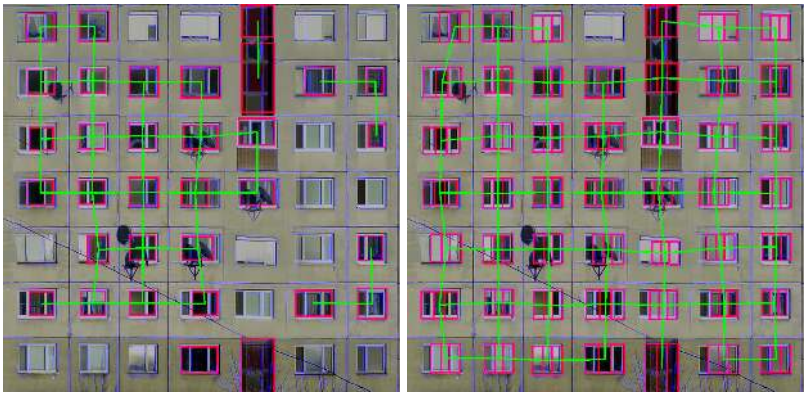


Fig. 5. Interpreted facades of a modern building. *Left:* Simple shape template with $t = 1$ fails to detect light windows. *Right:* Change to $t = 0.33$ improves the result significantly as the response from edge likelihood is stronger.

actually formed by shop-windows and the wall around them, and the visualized results show that our detector follows this interpretation. The roof area was difficult for our approach, since the color classifier considers them window-like.

While the authors in [10] claim their segmentation framework generalizes on some mild variants of Haussmannian facades, we can say our framework is not limited to any particular style at all. To prove this, we demonstrate results on modern buildings in Fig. 5 and 4 a).

Finally, we have made experiments with loosely regular facade of Frank Gehry's *Dancing House* shown in Fig. 4 b), where window alignment shows significant deviation from grid structure. We were successful in correctly locating all windows lying on the major plane as well as their neighborhood. The ability to handle sparse regular structures is presented on the right in Fig. 4 c).

7 Conclusion and Future Work

We have presented a recognition framework that uses a weak structure model to locate elements in images, and demonstrated its potential in the task of window detection in facades. Our experiments have demonstrated that structural regularity given by pair-wise attribute constraints can efficiently guide a stochastic process that estimates element locations and neighborhood at the same time. We have shown that the conjunction of a weak non-specific classifier and a weak structural model can lead to performance that would be hardly achievable by a well-trained specific classifier. Despite the seemingly complex description of the model, the ideas are simple and the implementation is straightforward.

In our future we would like to endow our recognition framework with more powerful classifiers and an ability to handle relations on multiple levels that would i.e. allow two different structural components to overlap.

Acknowledgment. This work has been supported by Google Research Award, by the Czech Ministry of Education under project MSM6840770012 and by Grant Agency of the CTU Prague under project SGS10/278/OHK3/3T/13.

References

1. Micusik, B., Kosecka, J.: Piecewise planar city 3D modeling from street view panoramic sequences. In: Proc. CVPR. (2009)
2. Hohmann, B., Krispel, U., Havemann, S., Fellner, D.: CITYFIT: High-quality urban reconstructions by fitting shape grammars to images and derived textured point cloud. In: Proc. of the International Workshop 3D-ARCH. (2009)
3. Pauly, M., Mitra, N., Wallner, J., Pottmann, H., Guibas, L.: Discovering structural regularity in 3D geometry. Transactions on Graphics **27** (2008) 43–43
4. Gips, J.: Shape grammars and their uses. Birkhäuser (1975)
5. Zhu, S., Mumford, D.: A stochastic grammar of images. Foundations and Trends in Computer Graphics and Vision **2** (2006) 362
6. Alegre, F., Dellaert, F.: A probabilistic approach to the semantic interpretation of building facades. In: International Workshop on Vision Techniques Applied to the Rehabilitation of City Centres. (2004)
7. Müller, P., Zeng, G., Wonka, P., Van Gool, L.: Image-based procedural modeling of facades. Transactions on Graphics **26** (2007) 85
8. Mayer, H., Reznik, S.: Building facade interpretation from uncalibrated wide-baseline image sequences. ISPRS Journal of Photogrammetry and Remote Sensing **61** (2007) 371–380
9. Ripperda, N., Brenner, C.: Data driven rule proposal for grammar based facade reconstruction. Photogrammetric Image Analysis **36** (2007) 1–6
10. Teboul, O., Simon, L., Koutsourakis, P., Paragios, N.: Segmentation of building facades using procedural shape prior. In: Proc. CVPR. (2010)
11. Toussaint, G.T.: The relative neighbourhood graph of a finite planar set. Pattern Recognition **12** (1980) 261 – 268
12. McLaughlan, G.J.: Finite Mixture Models. Wiley (2000)
13. Green, P.J.: Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. Biometrika **82** (1995) 711–732