Stochastic Recognition of Regular Structures in Facade Images

Radim Tyleček and Radim Šára

Abstract

We present a method for recognition of structured images and demonstrate it on the 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 attributes 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.

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.

I. Introduction

Recent development in the construction of virtual worlds like Google Earth™, Bing Maps 3D™ or Nokia Maps 3D™ heads toward a higher level of detail and fidelity. The 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 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 reconstructions presented by [12]. The complexity is particularly important when the representation has to scale with the size of cities in applications such as [9] who plan to combine range data with images. The work of [14] dealing directly with structural regularity in 3D data also supports our ideas.

While facades as man-made scenes exhibit strong 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 [7] and later picked up by [19], are the basic essence for all recent methods based on the procedural modeling to overcome the limitations of traditional segmentation techniques. The idea of shape grammars is that image can be explained by terminal symbols (objects) obeying a set of rules.

Some aspects of probabilistic approach were first discussed in [1], 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 [13] determines the structure by splitting the facade to a regular grid of individual tiles and subdividing them. Meyer and Reznik [10] 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 [15] has designed a comprehensive dictionary of domain-specific rules; the results presented on simple facades show this approach has difficulty to achieve good localization.

A recent method of [17] combines trained randomized forest classifiers with a shape grammar to segment Haussmannian[1] facades into eight classes. Their model assumes the windows form a grid while allowing different intervals. In the second step, positions of rows and columns in the grid are stochastically estimated by a specific random walk algorithm that does not propose dimension changes. Subsequently they proposed a new parser based on reinforcement learning to speed up the process in [16]. They evaluated their results quantitatively on a limited dataset of Haussmannian facades in Paris which is available online. In the same domain, the work of [4] demonstrates how a specific segmentation algorithm can be engineered for a particular regular style.

The majority of the mentioned algorithms for single-view facade interpretation work with hard constraints on grid configurations of windows and employ strong domain-specific heuristics. Additionally, they require the user design of a specific grammar or training, whereas both processes are prone to overfitting. In [18] there was presented a segmentation framework, where the structure is modeled softly by local pair-wise constraints, allowing loosely regular configurations like those in Fig. 5. The present paper revisits the weak structure model [18], it proposes several simplifying modifications, particularly it removes some unnecessary complexity in the model, makes it more flexible and improves its modeling power. The changes resulted in a significant improvement of performance, as discussed in Sec. VI.

II. Structural Recognition Framework

We consider the problem of recognizing elements in an image, specifically windows in a facade, we will call them *terminal elements*. We assume the input image is orthographically rectified, as in Fig. 4. Our model parameters (variables) consist of complexity k (the number of terminal elements), shape attributes A (size, aspect ratio, etc.), 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 joint distribution $p(I, \theta)$ with

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

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 will 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.

[1]Architectural style widely used during the reconstruction of Paris in 19th century.

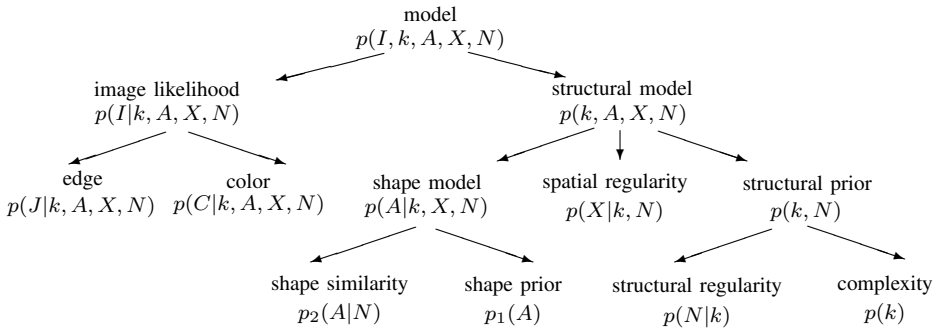


Fig. 1. Hierarchy in the probability model. Pdf of a node is a product of its leaves.

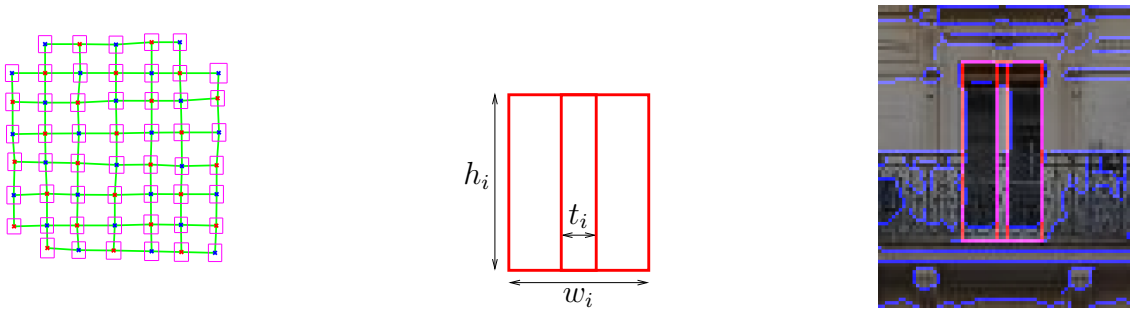
III. Structural Model

We design a probabilistic structural model $p(k, N, A, X)$ in which (k, N, A, X) is a configuration. The model captures the rules for appearance of a set of similar elements in image with a semi-regular spatial distribution. Rather than explicitly imposing a lattice or a similar global layout, the model is based on local pair-wise element neighborhood and attribute constraints. We are given a set of $k \in \mathbb{N}$ elements with locations $X = \{x_i \in (0,1)^2; i = 1, \ldots, k\}$ in the unit image plane. Our neighborhood representation is independent on the locations X and it is based on a graph $G = \{V, D\}$, where nodes $V = \{v_i; i = 1, \ldots, k\}$ correspond to terminal elements and edges $D = \{(u, v);\ u, v \in V\}$ to neighborhood relationship between them. Our goal is to define neighbors as terminal elements that are in proximity of each other and such that they share some attribute values.

We represent the edges D with adjacency matrix $N = \{l_{uv} \in \{0, 1\}; (u, v) \in V^2\}$ that is recovered as a part of the solution of (1).

A. Structural Prior

This prior describes a class of graphs that are similar to a lattice with a high level of flexibility. It combines structural regularity $p(N|k)$ and complexity $p(k)$.



a) b) c)

Fig. 2. a) A random sample close to the mode of a softly bipartite graph, nodes V on positions X are marked with crosses colored red or blue according to labels Z. Edges D are in green, terminal elements' shapes are in magenta. b) The 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. c) The shape template (red) is matched with image edges (blue).

Structural Complexity. The prior on the number of elements is modeled by Poisson distribution $p(k) = \mathrm{Pois}(k, \lambda_k)$ where λ_k is the expected number of terminal elements in an image.

Structural Regularity. We restrict G to a reasonable graph family. In [18] a planar *relative neighborhood graph* was used, but in certain situations it was too restrictive, preventing the inclusion of an edge where desirable. Instead, we choose now a more general *softly bipartite graph*. A bipartite graph is two-colorable, meaning that we can assign a binary label $c_i \in \{0, 1\}$ to every node such that every edge connects nodes with different labels. However, in our case we relax this condition by allowing edges connecting equally colored nodes but assigning them a low probability p_z (softness). We introduce a set of hidden variables $Z = \{z_i; i = 1, \ldots, k\}$ and model them with

$$p(Z|N) = \prod_{(u,v)\in D} p(z_u, z_v|l_{uv}), \text{where } p(\cdot|l_{uv} = 1) = \begin{cases} p_z, & z_u = z_v, \\ 1 - p_z, & z_u \neq z_v. \end{cases} \tag{2}$$

In the case of no edge we use uniform distribution $p(z_u, z_v|l_{uv} = 0) = \frac{1}{2}$. Note that the graph of a complete lattice (or its subgraph) is bipartite, but when some elements are missing in the middle, it may be not (chords can create odd cycles).

The preferred number of edges in the graph is modeled by binomial distribution

$$p(N|k) = \mathrm{B}\Big(\sum_{uv} l_{uv}, d_g(k), d_c(k)\Big), \tag{3}$$

where $d_g(k) = 2(k - \sqrt{k})$ is the number of edges in a complete square lattice with k nodes and $d_c(k) = \frac{1}{2}k(k-1)$ in a complete graph.

B. Spatial Regularity

This part of the model describes rules for the relative position of neighboring elements. We parametrize the spatial relation of two elements (u, v) in polar coordinates, i.e. by distance $\rho_{uv} = ||\mathbf{X}_u - \mathbf{X}_v||$ and angle $\varphi_{uv} = \mathrm{atan2}(y_u - y_v, x_u - x_v)$. We want to model multiple assumptions on $p(X|N, k)$ that are not independent, each of them represented with a pdf of the same variables X. Therefore we combine them in the form of a probability mixture [11]:

$$p(X|N) = \omega_a p_a(X|N) + \omega_s p_s(X|N) + \omega_c p_c(X|N), \tag{4}$$

where we have chosen $\omega_a = \omega_s = \omega_c = \frac{1}{3}$ and k was omitted in $p(\cdot)$ for simplicity of notation. The functions p_a, p_s, p_c are described next.

Alignment. The first assumption on the position of elements is that neighboring elements should be horizontally or vertically aligned parallel to axes of the rectified input image. We express it in

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

where $p(\phi_{uv}|l_{uv} = 1) = \mathrm{Beta}(\phi_{uv}, \beta_\varphi, 1)$, $\phi_{uv} = \frac{1}{2}(1 + \cos 4\varphi_{uv}) \in (0, 1)$ maps the angle to the unit interval. The probability in the case of a suppressed edge is uniform, $p(\phi_{uv}|l_{uv} = 0) = 1$.

Spacing. The second assumption is that the distance ρ_{uv} between elements in a neighborhood should most probably be equal. Let's first define a sorted circular list for a given terminal element u of its

n neighbors v_i $R(u) = \{v_i|(u,v_i) \in D, l_{uv_i} = 1, \varphi_{uv_i} \leq \varphi_{uv_{i+1}}\}$ ordered by angle, where no element is preferred as starting. Then we model the assumption by evaluating together distances to neighbors in

$$p_s(X|N) = \prod_{u=1}^{k} p\Big(\{\rho_{uv_i}\}_{v_i \in R(u)}|N\Big), \text{ in which} \tag{6}$$

$$p(\{\rho_{uv_i}\}|N) = \frac{1}{n}\mathrm{Dir}\Big(\big[\frac{\rho_{uv_1}}{\varrho_u}, \frac{\rho_{uv_2}}{\varrho_u}, \ldots, \frac{\rho_{uv_n}}{\varrho_u}\big], \alpha_\rho\Big) \cdot p\Big(\rho_u\Big), \tag{7}$$

where the factor $\frac{1}{n}$ is due to the free starting element, $\varrho_u = \sum_{v=1}^{n} \rho_{uv}$ is the sum of distances to neighbors; the mean distance $\rho_u = \frac{1}{n}\varrho_u$ has prior $p(\rho_u) = \mathrm{Beta}(\rho_u, \alpha_\rho, \beta_\rho)$. The symmetric Dirichlet pdf assigns the highest probability exactly when $\rho_{uv_1} = \rho_{uv_2} = \cdots = \rho_{uv_n}$.

Periodicity. Analogically, the next assumption for angles φ_{uv} is that the neighbors of a given element u should be evenly distributed around it, in terms of their relative angles. We have

$$p_c(X|N) = \prod_{u=1}^{k} p\Big(\{\varphi_{uv_i}\}_{v_i \in R(u)}|N\Big), \text{ in which} \tag{8}$$

$$p(\{\varphi_{uv_i}\}|N) = \frac{1}{n}\mathrm{Dir}\Big(\big[\frac{\varphi_{uv_2} - \varphi_{uv_1}}{2\pi}, \ldots, \frac{\varphi_{uv_n} - \varphi_{uv_{n-1}}}{2\pi}\big], \alpha_\varphi\Big), \tag{9}$$

where the Dirichlet pdf assigns the highest probability to configurations in which the differences between the neighbors' angles are equal to each other, i.e. π for two neighbors, $\frac{2\pi}{3}$ for three, $\frac{\pi}{2}$ for four, etc. This assumption partly replicates the alignment rules in p_a, however its purpose is to softly suppress multiple neighbors with closely similar φ_{uv} but different ρ_{uv}.

C. Shape Attributes

Aside from the locations X, the appearance of terminal elements is described with shape attributes. Our terminal elements are represented by a rectangular shape template with its borders parallel to image borders. The shape attributes $A = \{W, H, T\} = \{A_i = (w_i, h_i, t_i) \in (0,1)^3; i = 1, \ldots, k\}$ are described in Fig. 2, the relative central column position attribute t is specific to the 'window' class elements.

The attribute model can be described as a Markov Random Field on graph G with unary and binary factors, assuming element's attributes are conditionally independent of all other attribute variables given its neighbors (local Markov property). This is formulated in

$$p(A|k, N, X) = p_o(A|X) \underbrace{\prod_{i=1}^{k} p_1(A_i)}_{p_1(A)} \underbrace{\prod_{(u,v)\in D} p_2(A_u, A_v|l_{uv})}_{p_2(A|N)}, \tag{10}$$

where we additionally specify that when any two shape rectangles overlap each other, then $p_o(A|X) = 0$ effectively avoids such configuration (with the simplifying assumption of independence on p_1 and p_2).

Shape prior. The unary factors are attribute priors

$$p_1(A_i) = p(t_i|w_i, h_i)p(w_i|h_i)p(h_i), \tag{11}$$

where $p(t_i|w_i, h_i) = \mathrm{Beta}(t_i, \alpha_t, \beta_t)$, the aspect ratio has an asymmetric Dirichlet distribution $p(w_i|h_i) = \mathrm{Dir}\big([\frac{w_i}{w_i+h_i}, \frac{h_i}{w_i+h_i}], \{\alpha_a, \beta_a\}\big)$, and the height prior is $p(h_i) = \mathrm{Beta}(h_i, \alpha_h, \beta_h)$.

TABLE I
STRUCTURAL MODEL PARAMETERS.

parameter	λ_k	p_z	β_ϕ	α_ρ	$\alpha_\varrho, \beta_\varrho$	α_ϕ	α_t, β_t	α_h, β_h	α_a, β_a	α_s
value	50	0.01	10	10	5, 20	10	20,10	2,40	20,20	3

Shape similarity. Our attribute constraints reflect the fact that neighboring elements most probably have the same shape. This can be described with binary factors in

$$p_2(A_u, A_v|l_{uv}) = \begin{cases} p(w_u, w_v)p(h_u, h_v)p(t_u, t_v), & \text{if } l_{uv} = 1, \\ 1, & \text{if } l_{uv} = 0, \end{cases} \tag{12}$$

where $p(w_u, w_v) = \mathrm{Dir}\big([\frac{w_u}{w_u+w_v}, \frac{w_v}{w_u+w_v}], \alpha_s\big)$ is a symmetrical distribution with its mode at $w_u = w_v$, in the case of $l_{uv} = 0$ the distribution is uniform. Analogically we define the pdfs for h and t.

D. Parameter Learning

The parameters of the structural model were first learned by fitting the respective distributions to values computed on the annotated training image set described in Sec. VI. However, for some small or highly regular datasets, the variance of the regularity and similarity variables can be very small, approaching a Dirac pdf, resulting in numerical and sampling problems with the fitted pdfs. For such variables, we specified the minimum variance shown in Tab. I based on empirical tests, what also helped to establish balance between individual parts of the model. In the future we would like to find a solution to the problem of complex model tuning such that requires less interaction.

For this setting we have verified our model $p(k, N, A, X)$ by constructing a random sample generator from the distribution, generating a sequence of 10^6 samples and selecting the most probable sample in the sequence. As expected, we got a regular configuration shown in Fig. 2a).

IV. DATA LIKELIHOOD

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

The data likelihood model $p(I|k, A, X, N)$ is similar to [18]. We express the probability of observing an image I given a configuration (k, A, X, N). We combine two independent 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 window borders.

A. Edge Likelihood

We assume that window borders correspond to edges and represent them by oriented edge image $J = \{J_i; i \in I\}$, which segments the image into horizontal edge, vertical edge and background regions. It is then matched with the edge image $R(A, X)$ rendered from the current configuration specified by attributes A, X and the shape template in Fig. 2. The underlying pdf $p(J|A, X) = \prod_{i \in I} p(J_i|R_i(A, X))$ is described in detail in [18]. It is 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 (Sect. V) very efficient.

B. Color Likelihood

We extend the simple color model from [18] and model the input color image $C = \{c_i \in (0,1)^3; i = 1, \ldots, k\}$ with a multivariate Gaussian mixture distribution with $m = 3$ components that targets the 'window' class. We use the configuration (A, X) to partition pixels either to foreground (window) set C_f or background (non-window) set C_b such that $C_f \cap C_b = 0$. Assuming pixel independence, the probability of observing segmented image is

$$p(C|A,X) = \prod_{i \in C_b} p_b(c_i) \prod_{j \in C_f} p_f(c_j) = p_b^{|I|} \prod_{j \in C_f} \frac{p_f(c_j)}{p_b}, \tag{13}$$

where the background probability $p_b(c_i) = p_b = 10^{-8}$ is uniformly constant, $|I|$ is the image size in pixels and the foreground color model is expressed by $p_f(c_j) = \sum_{i=1}^{m} \omega_j \mathcal{N}(c_j|\mu_i, \Sigma_i)$. The mixture parameters $\omega_j, \mu_i, \Sigma_i$ are learned as ML estimates obtained with the EM algorithm [5] by fitting color of 'window' class pixels sampled from the annotated training image set.

Like in edge likelihood, color likelihood is evaluated using pre-computed integral images in linear time, and as (13) suggests, we evaluate foreground pixels only.

V. Recognition Algorithm

We have chosen Reversible Jump MCMC framework [8] 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 so that it is easy to sample from it. We should point out that the quality of the resulting interpretation is determined by the probability model, on the other hand 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 using conditional probability $q(\theta'|\theta)$.

The conditional sampler $q(\theta'|\theta, I) \rightarrow \theta'$ 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 top-level sampler only chooses from $q(m|\theta)$ which of the individual samplers m will be used to propose the next move. We use the set of samplers from [18] to explore the space of parameters θ and extend them to fit the specific needs of the new structural model and to improve the acceptance rate. 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 ratio of Metropolis-Hastings (MH) algorithm:

$$\alpha = \frac{p(I|\theta')p(\theta')}{p(I|\theta)p(\theta)} \cdot \underbrace{\frac{q(m|\theta')}{q(m'|\theta)}}_{\alpha_m} \cdot \underbrace{\frac{q_m(\theta|\theta')}{q_m(\theta'|\theta)}}_{\alpha_q} \cdot \underbrace{\frac{q_{\leftarrow}(u_{\leftarrow}|\theta')}{q_{\rightarrow}(u_{\rightarrow}|\theta)}}_{\alpha_u} \cdot J_{\rightarrow}, \tag{14}$$

where α_m reflects the choice of individual samplers, α_q is the proposal density ratio ($\alpha_q = 1$ when the proposals are symmetric), α_u and $J_{\rightarrow}$ are related to dimension changes and will be described in Sec. V-C. The proposed move is accepted with probability $A = \min\{1, \alpha\}$. The chain is initialized with $k_0 = 0$, then the only allowed proposal is to add a new element (see Sec. V-C).

A. Proposal Selection

The sampler mixture distribution $q(m)$ is constructed hierarchically, we first choose a probability $q_{RJ} = 0.1$ of reversible jump proposals, from which follows that the ordinary MH jumps have $q_{MH} = 1 - q_{RJ} = 0.9$. In the second step, we choose uniformly one of the jumps from the appropriate set of proposals (either $q(m|MH)$ or $q(m|RJ)$) presented in Sections V-B and V-C.

Proposing dimension changes is expensive, therefore we adapt the proposal distribution according to the current state to achieve a speed up by reducing reversible jumps. This is done by constructing a conditional distribution $q_t(RJ|\theta_t) = q_{RJ} + Te^{-\frac{t}{\tau}}$, we choose in practice $T = \frac{1}{4}$, $\tau = 10^4$. The vanishing adaptation (i.e. $q_t(RJ|\theta_t) \to q_{RJ}$) guarantees convergence of the chain even if it is no longer ergodic due to its adaptation [2].

B. Metropolis-Hastings Moves

The moves introduced in this section perform attribute modifications, thus do not modify the model complexity k and can be evaluated by a classical MH algorithm (14), where the ratio has $\alpha_u = 1$ and $J_\to = 1$.

We pick up an element $i \sim \mathcal{U}(\{1, \ldots, k\})$ from discrete uniform distribution and perturb some of its attribute values randomly. We have adopted the *drift, resize, flip* and *resample* from [18], here we propose modifications and extensions for the new model.

Enforce attribute constraints. 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 $(u, v) \sim \mathcal{U}(D)$ and transfer a randomly selected attribute (h,w or t) value over the edge from one element to another according to the specific constraints, i.e. $a_u' = a_v$.

Modify neighborhood. We include a move to allow changes to the neighborhood structure: It picks up a random edge $(u, v) \sim q(u, v|X)$ and changes its label $l_{uv}' = 1 - l_{uv}$, effectively suppressing or recovering the edge. The edge proposal $q(u, v|X)$ is an empirical distribution on $\{\frac{1}{\rho_{uv_i}}\}_{v_i \in R(u)}$ to prefer nodes closer to each other, reflecting the idea of proximity of neighbors.

Modify node coloring. This move picks up a random node $i \sim q_z(k|N)$ and changes its node color to $z_i' = 1 - z_i$. The distribution $q_z(k|N)$ is constructed to prefer nodes i from a set where the two-coloring property of softly bipartite graph is violated, i.e. some of its neighbors u have the same color $z_u = z_i$. We choose from this set with $q_z = 0.9$.

C. Reversible Jump Moves

We also need to find the number of elements k, that controls the dimension of the vector 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 in (14) 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 [8].

Birth and Death. By inserting a new element into our model we propose an increase of dimension $k \to k' = k + 1$, or in the case of *death* a decrease $k \to k' = k - 1$. The derivation of the acceptance ratios is given in [18].

a) basic b) flipped

Fig. 3. Splitting scenarios.

In the basic case of *birth* the new position is sampled uniformly, $x_* \sim \mathcal{U}$ and the new attributes from the prior $a_* \sim p_1(A)$. The jumps below are special cases of *birth* that exploit the structure for predicting values for the new elements, which can be generally described as sampling from $a_*, x_* \sim q(a, x|N)$. We designed them to sample from the marginal distributions of the structural model where possible.

Append. In this case of the *birth* jump we first choose uniformly an existing terminal element $i \sim \mathcal{U}(k)$ and place the new element relatively to its position according to $x_* = x_i + \rho\nu(\varphi)$, where $\nu(\varphi) = \big[\sin(\varphi)\cos(\varphi)\big]$ and we sample $\rho \sim p(\rho_{uv})$ and $\varphi \sim p(\varphi_{uv}|l_{uv} = 1)$ from the regularity marginals. Its attributes a_* are sampled relatively to a_i from the marginal Beta pdf of similarity by $\delta \sim p_2(A|N)$ and then $a_* = a_i\frac{1-\delta}{\delta}$. We explicitly set the edge $l_{i*} = 1$ and the Jacobian here is $J_\rightarrow = \rho$.

Replicate. This jump is similar to *append*, but we directly sample an edge $(u, v) \sim \mathcal{U}(D)$ and set the new window position to $x_* = x_v + \rho_{uv}\nu(\varphi_{uv})$ where ρ_{uv} and φ_{uv} are replicated from the sampled edge. The Jacobian is here $J_\rightarrow = \rho$.

Extend. In this case we add two new elements $*_1, *_2$ at once and connect them with edges to create a new face (4-cycle) in the graph G. We sample an edge $(u, v) \sim \mathcal{U}(D)$ and set the new positions to $x_{*_1} = x_u + \rho_{uv}\nu(\varphi_*)$ and $x_{*_2} = x_v + \rho_{uv}\nu(\varphi_*)$ where $\varphi_* = \varphi_{uv} \pm \frac{\pi}{2}$, the sign is chosen randomly. The attribute values are replicated from a_u to a_{*_1} and a_v to a_{*_2}. The face is completed by adding edges $l_{u*_1} = l_{v*_2} = l_{*_1*_2} = 1$.

Split and Merge. The *split* move proposes increase of dimension $k \rightarrow k' = k + 1$, where an existing element is transformed into two new ones. Its purpose is to create a shortcut in the parameter space, because an equivalent concatenation of the above moves has a small acceptance. The general split scenarios are shown in Fig. 3. We choose the element $v \in \{1, \ldots, k\}$ to be split, the split direction (horizontal/vertical) and sample the split factors $s_{ij} \in (0, 1)$ from Beta distribution as the communicating variables $u_\rightarrow = [\ s_{11}\ \ s_{12}\ \ s_{21}\ \ s_{22}\ ] = \mathbf{s}$. The alpha and beta parameters are chosen according to a given split scenario, i.e. for the vertical scenario $s_{11}, s_{12} \sim \mathrm{Beta}(2, 2),\ s_{21} \sim \mathrm{Beta}(1, 10),\ s_{22} \sim \mathrm{Beta}(10, 1)$. We work with the element rectangles represented by upper-left and lower-right corners $B(X, A) = \big\{B_i = \big[b_{11}\, b_{12}\, b_{21}\, b_{22}\big] = \big[x_i - \frac{1}{2}w_i, y_i - \frac{1}{2}h_i,\ x_i + \frac{1}{2}w_i, y_i + \frac{1}{2}h_i\big]\big\}$. The corresponding dimension matching function is then

$$f_\rightarrow(B, u_\rightarrow) = f_\rightarrow\big(B, [s_{11}\, s_{12}\, s_{21}\, s_{22}]\big) = \big(\{B_{\neg v}, B'_v, B^*\}, []\big) = \big(B', []\big), \tag{15}$$

where for horizontal orientation the rectangles are set to

$$B^* = \big\{\{b_{11} + s_{12}w_v, b_{12} + s_{22}h_v\}, \{(1 - s_{12})w_v, (1 - s_{21})h_v\}\big\}, \tag{16}$$

$$B'_v = \big\{\{b_{11}, b_{12}\}, \{s_{11}w_v, s_{22}h_v\}\big\}, \tag{17}$$

which inserts B^* into the set. The case for vertical orientation is derived analogically. The Jacobian (for both orientations) $J_\rightarrow = \left|\frac{\partial(B_{\neg v}, B'_v, B^*)}{\partial(B, B^*)}\right| = w_v^2 h_v^2$ is calculated from the variables that actually change.

The inverse move is *merge*, for which we have no communication variable $u_{\leftarrow} = []$ (it is deterministic), and choose the two neighboring elements $B_v, B^{\dagger} \in B'$ to be merged into one. To establish reversibility, we define inverse matching function as

$$f_{\leftarrow}(B', u_{\leftarrow}) = f_{\leftarrow}\big(\{B_{\neg v}, B'_v, B^{\dagger}\}, []\big) = \big(\{B\}, \mathbf{s}\big) \sim \big(B, u_{\rightarrow}\big), \tag{18}$$

where $B^{\dagger}$ is the removed element and B_v is the merged element, $B = \{B' \setminus B^{\dagger}\}$. The split configuration is detected and ratios $\mathbf{s}$ are calculated from the affected element pair $B_v, B^{\dagger}$, inversely to (17). In the split move acceptance we now have $\alpha_u = \frac{1}{q_{\rightarrow}(\mathbf{s})}$, where $q_{\rightarrow}(\mathbf{s}) = p(s_{11})p(s_{12})p(s_{21})p(s_{22})$ is the prior probability of the split and $\alpha_q = \frac{k}{k+1}$ reflects terminal element selection.

For *merge*, where $k \rightarrow k' = k - 1$, the merged element rectangle B'_v is a bounding box of the merged elements $B_v, B^{\dagger}$ and the Jacobian is now $J_{\rightarrow} = \left|\frac{\partial(B', \mathbf{s})}{\partial(B)}\right| = \frac{1}{w_v^2 h_v^2}$. Again, with appropriate change of labeling, the derivation of *merge* move is the same as for *split*, except for the inversion of ratios, i.e. $\alpha_q = \frac{k+1}{k}$ and $\alpha_u = q_{\rightarrow}(\mathbf{s})$, where the corresponding split factors $\mathbf{s}$ must be calculated from the input configuration.

D. Convergence and Complexity

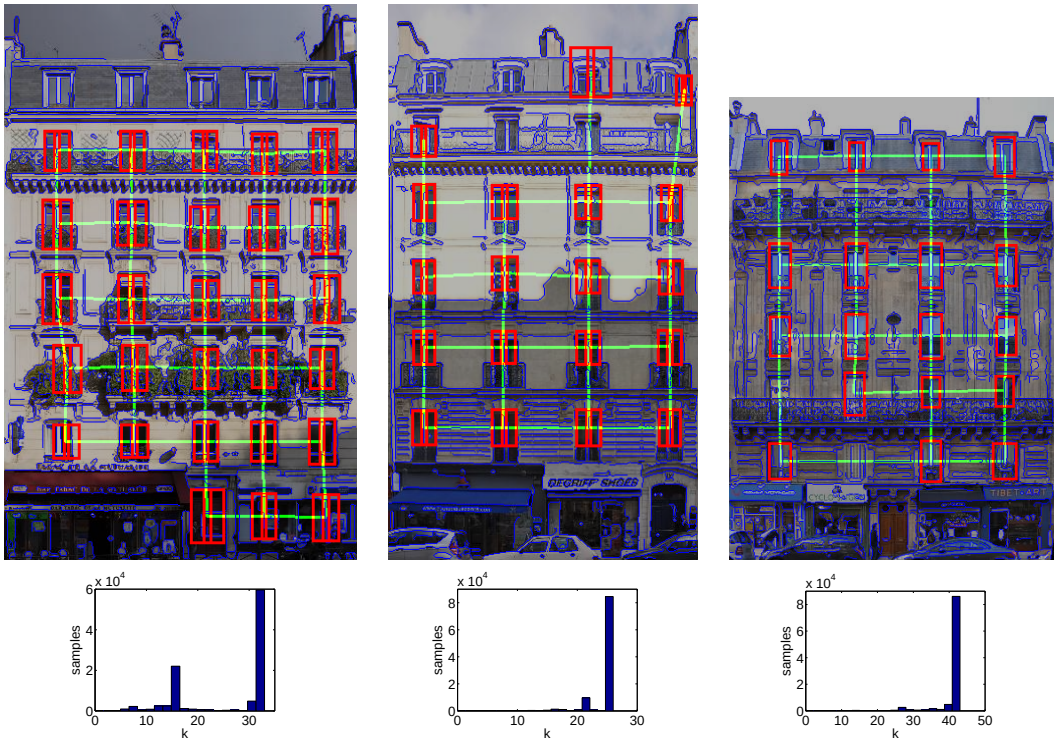
We have found that the typical necessary number of MCMC samples (classifier calls) is proportional to image size in pixels $|I|$ (from 30% for easy instances to 200% for the difficult ones). As a result, we fixed the number of samples in our current method to a pessimistic estimate, but our experiments suggest that significantly shorter sampling time could be achieved with a suitably designed stopping condition. Another option is to use a more efficient sampling scheme, i.e. [6] for the continuous part or [3] for the discrete variables (labels).

VI. 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 5×10^5 iterations in our experiments, which roughly equals to visiting all pixels in the analyzed images. With our Matlab implementation, the running time was under one minute on a standard 2 GHz CPU.

The only public dataset known to us that allows quantitative comparison in this area has been provided by [17]. The dataset consists of 30 rectified and annotated images of facades from a street in Paris, which share attributes of Haussmannian style but differ in illumination conditions. We have trained our model on 20 of them and 10 were used for testing. 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 used a similar reduction as in [18] and merged the columns of confusion matrix given in [17] into two, treating all original classes other than *window* as our background (non-window).

The results in Tab. II for *window* and *wall* suggest that the proposed method is performing better in the terms of high specificity when compared both to the procedural segmentation (PS) framework [17] and the weak structural model constrained to relative neighborhood graphs (RNG) [18], also see Fig. 6. We attribute this to the extended color likelihood model with Gaussian mixtures in the HSV color space (which is less sensitive to the illumination changes), on the other hand, it resulted in a small drop in



$H = 1.3705$ a) Monge No. 13

$H = 0.6632$ b) Monge No. 43

$H = 0.6920$ c) Monge No. 50

Fig. 4. *Top row:* Visualization of selected results from Parisian dataset [17], facade a) is occluded by plants, in facade b) a cast shadow is present. False positive windows 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. *Bottom row:* Posterior histograms for complexity k.

sensitivity to the window class. The new bipartite structural model with parameters learned from the annotations also contributed to the results, it is able to support windows completing the structure even where the likelihood response is low. This allows us to achieve good results even when the illumination varies and partial occlusion of windows is present, as shown in Fig. 4.

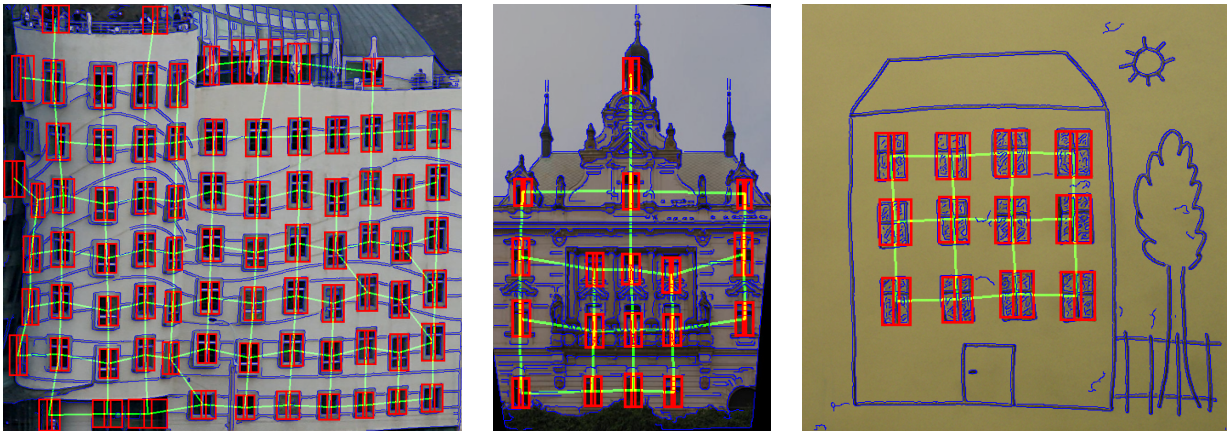
Posterior histograms shown in Fig. 4 for complexity k demonstrate different difficulty of the images, which is quantified by estimated entropy H. In the case of a) there is another less probable interpretation for $k = 15$ (missing some rows of windows), resulting in higher H.

To prove our framework is not limited to a particular style, we demonstrate results on modern buildings and even hand drawn images in Fig. 7 and Fig. 5. Note the appearance of edges in Fig. 7a) connecting the 'shifted' middle column, which was not possible in [18] due to the RNG constraint. The shape parameter t in Fig. 7b) which was fixed in [18] is now inferred along with the other parameters of the model.

Finally, we have made experiments with loosely regular facade of *Dancing House* shown in Fig. 5a), where window alignment shows significant deviation from the grid structure and we were successful in correctly locating all windows lying on the major plane as well as their neighborhood.

VII. Conclusion

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



a) Irregular facade b) Sparse structure c) Hand-drawn

Fig. 5. Results on non-standard facade images.

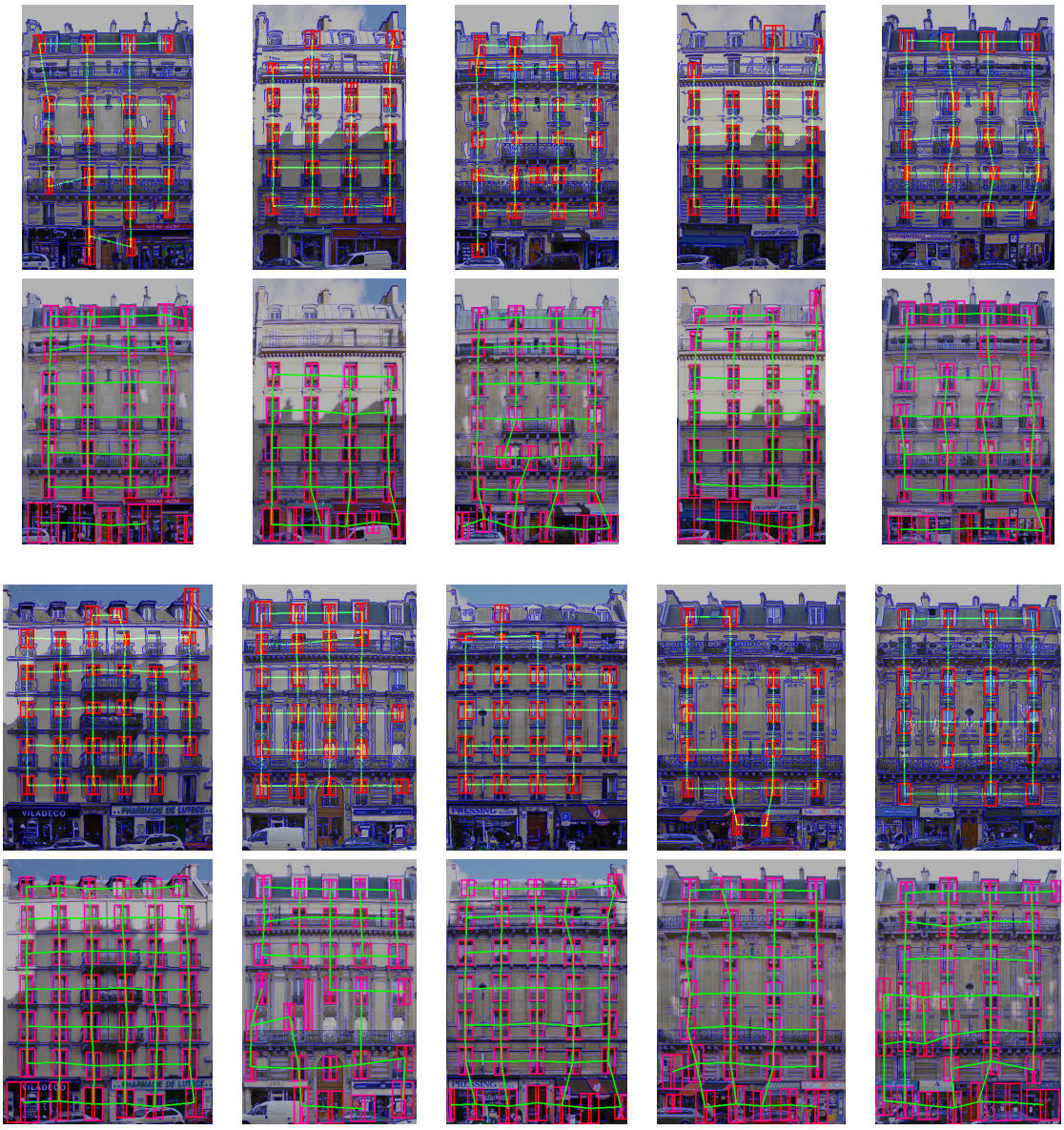


Fig. 6. *Rows 1 and 3:* Results of the proposed method on the ten test images in the Parisian dataset. *Rows 2 and 4:* Results on the same set from [18].

TABLE II

QUANTITATIVE RESULTS ON THE PARISIAN DATASET [17] SHOWN AS PERCENTAGE OF PIXELS FROM EACH CLASS SPECIFIED IN A ROW. THE AREA IS THE PERCENTAGE OF PIXELS OF A GIVEN CLASS IN THE WHOLE TEST SET. PS STANDS FOR PROCEDURAL SEGMENTATION [17], RNG FOR RELATIVE NEIGHBORHOOD GRAPH [18].

ground truth [17]		PS [17]		RNG [18]		proposed	
class	area	hit	miss	hit	miss	hit	miss
window	11	81	19	83	17	76	24
wall	48	83	17	84	16	98	2
balcony	12	72	28	*60*	*40*	*89*	*11*
door	1	71	29	*65*	*35*	*100*	*0*
roof	4	80	20	*51*	*49*	*95*	*5*
chimney	1	0	100	*83*	*17*	*96*	*4*
sky	7	94	6	*99*	*1*	*100*	*0*
shop	14	95	5	*60*	*40*	*99*	*1*
other	2	0	100	*61*	*39*	*96*	*4*
area-weighted		81	19	77	23	93	7

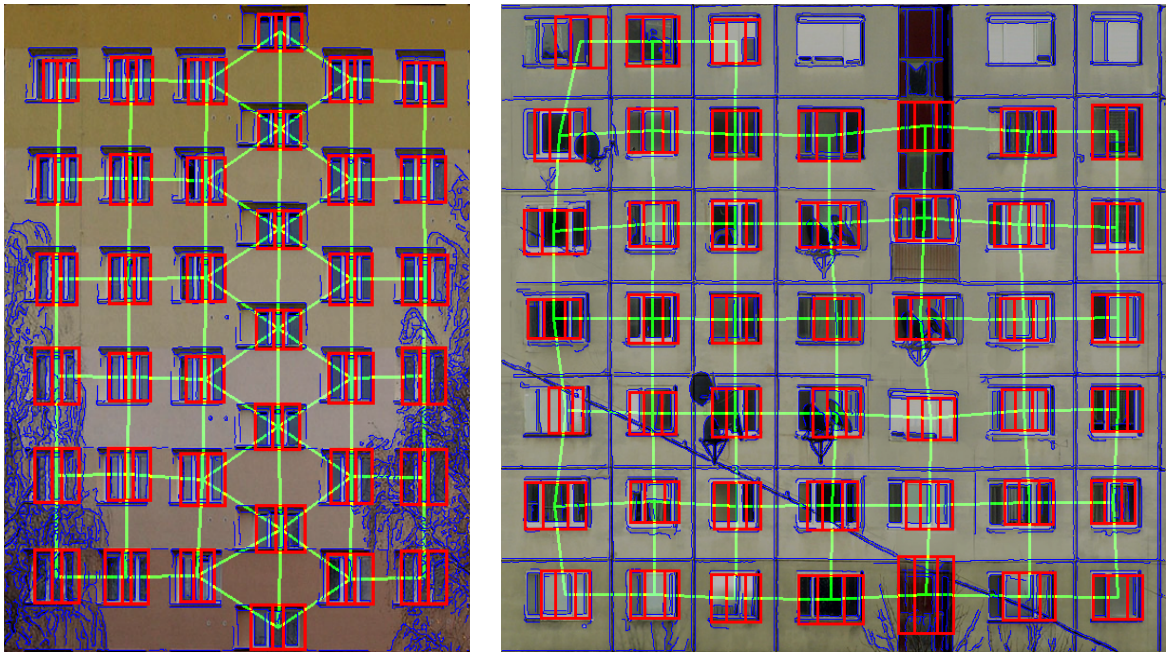


Fig. 7. Interpreted facades of modern buildings.

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.

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

ACKNOWLEDGMENT

This work was supported in part by the Czech Science Foundation under Project P103/12/1578.

REFERENCES

[1] F. Alegre and F. Dellaert. A probabilistic approach to the semantic interpretation of building facades. In *International Workshop on Vision Techniques Applied to the Rehabilitation of City Centres*, 2004.

[2] C. Andrieu and J. Thoms. A tutorial on adaptive MCMC. *Statistics and Computing*, 18(4):343–373, 2008.

[3] A. Barbu and Song-Chun Zhu. Generalizing swendsen-wang to sampling arbitrary posterior probabilities. *PAMI*, 27(8):1239 –1253, aug. 2005.

[4] Liu Chun and André Gagalowicz. 3D modeling of Haussmannian facades. In *Proc. of the 5th international conference on CV/CG collaboration techniques*, MIRAGE'11. Springer-Verlag, 2011.

[5] Dempster, A.P., Laird, N.M., and Rubin, D.B. Maximum likelihood from incomplete data via the EM algorithm. *J. R. Statist. Soc.*, (39):1–38, 1977.

[6] Simon Duane, A. D. Kennedy, Brian J. Pendleton, and Duncan Roweth. Hybrid Monte Carlo. *Physics Letters B*, 195(2):216–222, September 1987.

[7] J. Gips. *Shape grammars and their uses*. Birkhäuser, 1975.

[8] Peter J. Green. Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. *Biometrika*, 82:711–732, 1995.

[9] B. Hohmann, U. Krispel, S. Havemann, and D. Fellner. 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.

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

[11] G. J. McLaughlan. *Finite Mixture Models*. Wiley, 2000.

[12] B. Micusik and J. Kosecka. Piecewise planar city 3D modeling from street view panoramic sequences. In *Proc. CVPR*, 2009.

[13] P. Müller, G. Zeng, P. Wonka, and L. Van Gool. Image-based procedural modeling of facades. *Transactions on Graphics*, 26(3):85, 2007.

[14] M. Pauly, N.J. Mitra, J. Wallner, H. Pottmann, and L.J. Guibas. Discovering structural regularity in 3D geometry. *Transactions on Graphics*, 27(3), 2008.

[15] N. Ripperda and C. Brenner. Data driven rule proposal for grammar based facade reconstruction. *Photogrammetric Image Analysis*, 36(3/W49A), 2007.

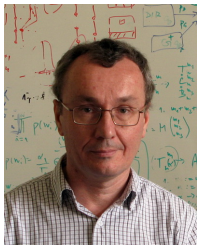
[16] O. Teboul, I. Kokkinos, L. Simon, P. Koutsourakis, and N. Paragios. Shape grammar parsing via reinforcement learning. In *Proc. CVPR*, 2011.

[17] O. Teboul, L. Simon, P. Koutsourakis, and N. Paragios. Segmentation of building facades using procedural shape prior. In *Proc. CVPR*, 2010.

[18] Radim Tyleček and Radim Šára. A weak structure model for regular pattern recognition applied to facade images. In *Proc. ACCV 2010*, volume 6492 of *LNCS*. Springer, 11 2011.

[19] S.C. Zhu and D. Mumford. A stochastic grammar of images. *Foundations and Trends in Computer Graphics and Vision*, 2(4):362, 2006.

Radim Tyleček is a PhD student at the Center for Machine Perception at the Faculty of Electrical Engineering, Czech Technical University in Prague. He received his master degree there in 2008. Since his master studies, he has been interested in computer vision, his focus is on structural recognition and reconstruction.



Radim Šára is an associate professor at the Czech Technical University in Prague since 2008. He received his PhD degree in 1994 from the Johannes Kepler University in Linz, Austria. From 1995 to 1997 he worked at the GRASP Laboratory at University of Pennsylvania. In 1998 he joined the Center for Machine Perception. His research interests are in computer vision, include robust stereovision, shape-from-X, and structural object recognition.