Learning Domain Knowledge for Façade Labelling

Dengxin Dai[1,2], Mukta Prasad[1], Gerhard Schmitt[2], and Luc Van Gool[1]

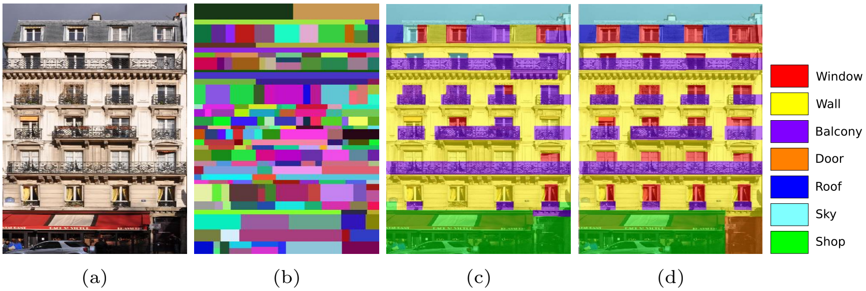
[1] Computer Vision Lab, ETH Zürich
[2] Chair for Information Architecture, ETH Zürich

Abstract. This paper presents an approach to address the problem of image façade labelling. In the architectural literature, domain knowledge is usually expressed geometrically in the final design, so façade labelling should on the one hand conform to visual evidence, and on the other hand to the architectural principles – how individual assets (e.g. doors, windows) interact with each other to form a façade as a whole. To this end, we first propose a recursive splitting method to segment façades into a bunch of tiles for semantic recognition. The segmentation improves the processing speed, guides visual recognition on suitable scales and renders the extraction of architectural principles easy. Given a set of segmented training façades with their label maps, we then identify a set of meta-features to capture both the visual evidence and the architectural principles. The features are used to train our façade labelling model. In the test stage, the features are extracted from segmented façades and the inferred label maps. The following three steps are iterated until the optimal labelling is reached: 1) proposing modifications to the current labelling; 2) extracting new features for the proposed labelling; 3) feeding the new features to the labelling model to decide whether to accept the modifications. In experiments, we evaluated our method on the ECP façade dataset and achieved higher precision than the state-of-the-art at both the pixel level and the structural level.

1 Introduction

Following handwriting, faces and pedestrians, architectural scenes are becoming another example that merits special consideration. This follows from the fact that they all have specific structures and important practical applications. This paper addresses the problem of labelling all building assets that an image façade contains, such as wall, doors, windows and balconies. The task is of broad interest due to its wide spectrum of applications, such as digitalization of existing cities and abstraction of grammar rules for building recreation [1,2]. Although numerous methods have been developed for labelling everyday objects in images and they can do this with reasonable pixelwise accuracy [3,4,5,6], few of them can achieve satisfactory results for our task. This is because in addition to pixelwise precision, our task is also very demanding in terms of structural precision – whether the structural layout of inferred labels form a "valid" façade. In this

A. Fitzgibbon et al. (Eds.): ECCV 2012, Part I, LNCS 7572, pp. 710–723, 2012.



(a) (b) (c) (d)

Fig. 1. The typical steps of our method: From (a) the input façade, through (b) façade segmentation and (c) the primary labelling via visual recognition, to (d) the labelling after learning the meta-features. (Best viewed in color.)

paper we aim at obtaining more precise labelling at both the pixel level and structural level, by specialising on images of façades.

For façade labelling, our key observations are: 1) urban façades can be explained by a bunch of mutually-exclusive and collectively-exhaustive rectangular tiles, each with a semantic label; 2) the semantic labels of the tiles are not distributed arbitrarily over the façades, but rather are governed by architectural principles. So, façade labelling can benefit from visual recognition at the tile level and from leveraging the architectural principles embedded in façades. With this in mind, we first propose a recursive splitting method to segment façades into a set of compact tiles, which are used as the units for semantic recognition. We then identify a set of meta-features on the segmented tiles and their label maps to capture both the visual evidence and the architectural principles. The weights of these features are learned from the training façades and are then used to label new ones. The façade specific segmentation method allows for visual recognition at the suitable scale and the meta-features leverage the architectural principles, which together lead to the higher precision of our method than competing methods, at both pixel and structural levels. The typical steps of our method are shown in Fig. 1.

The remainder of this paper is structured as follows: § 2 summarizes related work and § 3 describes our splitting segmentation method. In § 4 we first identify the meta-features and then integrate them into an energy function, which is followed by the inference and learning. § 5 shows the experimental results and § 6 concludes the paper.

2 Related Work

Berg *et al.* [7] presented a parsing method for architectural scenes by first parsing images at a coarse level into regions of sky, foliage, buildings, and street, and then at a finer level identifying the positions of windows and doors. Zhao *et al.* [8] also developed a two-stage algorithm by first parsing images into buildings, sky and grass, and then partitioning the buildings into separate façades. Later on, Wendel *et al.* [9] presented an unsupervised method for separating specific façades from each other in street-side images. Recently, a very intuitive method was developed for discovering the basic structure of façades by analyzing the cumulative effect of edges [10]. Our work is designed for labelling all assets in isolated and rectified image façades, so these works are complementary to our method.

Xiao *et al.* [11,12] addressed large scale reconstruction of façades captured along streets. The method first represents a façade as a Directed Acyclic Graph and then augments its nodes with depth values by structure from motion. Dick *et al.* [13] proposed a probabilistic approach to 3D reconstruction of architectural scenes, where a building is represented as a set of walls embedding a 'Lego' kit of parameterized primitives. Li *et al.* [14] presented a method for depth-layer decomposition of façades by fusing two complementary data acquisition modes: 2D photographs and 3D LiDAR scans. Very recently, Musialski *et al.* [15] proposed an interactive editing framework, where the symmetries across façades are exploited to minimize human interaction to obtain detailed 3D models of façades. While the goal of these works are very similar to ours, none of them explicitly learns the architectural principles for façade modeling.

Moreover, there has been a distinguished stream of research for façade modeling with procedural grammars. Müller *et al.* [2] detected symmetries and repetitions in façades using Mutual Information and converted the final hierarchical subdivisions into a procedural model. The state-of-the-art results along this thread was reported by Teboul *et al.* [16,17], where input façades were assumed to be particular instances of a predefined grammar, and the façade labelling problem was casted to a parameter optimization problem. While great successes have been achieved with these approaches, they have two major limitations. One is that if the defined grammar is incomplete or the wrong one for the input façades, the possibility of obtaining precise models drops dramatically. Another is that experts are needed to create the grammars. In contrast, our method learns automatically from training data.

Our work can also be regarded as a special case of learning context information for façade labelling. A lot of work has been proposed for exploring context information in image labelling [3,18,5,4]. But, such work was designed to capture the very general context information in images such as neighboring relationships and object co-occurrences, which are not all too helpful for façades and cannot leverage architectural principles. One notable exception is given by Recursive Neural Networks (RNN) [19]. They were developed to capture the hierarchical structure found in scenes by learning how to combine neighboring segments into super-segments, and so on until the whole image is covered.

3 Façade Segmentation

For the purpose of façade labelling, we would like to create a set of candidate segments to work with, thus reducing the need to process all the pixels but also allowing for the easier extraction of architectural principles and for visual recognition at desirable scales. Therefore, we want these candidates to have certain properties. Their boundaries need to conform to the construction principles of façades and cannot be shaped arbitrarily, while their interiors should have the characteristics of building assets[1]. Superpixels from established segmentation methods [20,21] and patches from dense sampling do not have this merit.

In this section, we design a recursive segmentation method to recursively partition each façade into K local tiles x_k. See Fig. 1(b) for an example of the segmentation results. At each step, one of the current leaf façade blocks X (an image block of $H \times W$ pixels) is partitioned along a line $\mathbf{l}$ into two child-blocks X_l^1 and X_l^2. $\mathbf{l}$ is restricted to be either a horizontal or a vertical line (a row or column of the image), since most patterns in common (rectified) façades are aligned with these two directions. Which block X will be chosen for splitting and where $\mathbf{l}$ will be placed is decided by evaluating which choice has the strongest data support. We consider two types of data support: edge support and content support. Edge support means that the split happens along prominent edges, so that 2D assets are not broken up and lie on either side. Content support means that the two resulting sub-blocks are both homogeneous within but different from each other. Edge statistics have been widely exploited in previous façade modeling systems [2,11,10] and we formulate them in the way of [10]. Since edge statistics are local and sometimes fragile, we supplement them with content statistics for content support.

Edge support. For X, the edge strength s_{ij} and direction $\mathbf{f}_{ij}$ are computed at every pixel f_{ij} and the direction consistency between a pixel edge and a given line, is computed as: $\left(\mathbf{l}^\top \mathbf{f}_{ij}\right)^2$. The distance of a pixel to a given line $\mathrm{dist}(f_{ij}, \mathbf{l})$ is weighted non-linearly in order to weaken the contribution from points further away from $\mathbf{l}$. This is given by $\phi(i,j,\mathbf{l}) = \exp(-\mathrm{dist}(i,j,\mathbf{l})/\sigma_e)$. The ϕ term allows for a small error in pixel position while contributing to $\mathbf{l}$, where σ_e controls the speed of the drop in contribution with the distance (we use $\sigma_e = 5$ in this paper). For a given $\mathbf{l}$, its agreement with image edge is measured by:

$$S^+(X,\mathbf{l}) = \frac{\sum_{i,j} \mathrm{s}_{ij} \cdot \phi(i,j,\mathbf{l}) \cdot \left(\mathbf{l}^\top \mathbf{f}_{ij}\right)^2}{|\mathbf{l}|} \tag{1}$$

where $|\mathbf{l}|$ is the length of the segment of $\mathbf{l}$ in X, in pixels. Similarly, the disagreement between $\mathbf{l}$ and image edges is measured by considering the above expression for $\mathbf{f}_{ij}^{\perp}$, i.e. the direction perpendicular to $\mathbf{f}_{ij}$. The edge support for $\mathbf{l}$ then is:

$$S(X,\mathbf{l}) = S^+(X,\mathbf{l}) - \lambda \cdot S^-(X,\mathbf{l}) \tag{2}$$

where λ is a weighing parameter.

Content support. The content support $\Upsilon(X_\mathbf{l}^1, X_\mathbf{l}^2)$ for line $\mathbf{l}$ is measured as the inverse of the normalized cut $ncut(X_\mathbf{l}^1, X_\mathbf{l}^2)$ (as defined in Eq.2 of [20]) of $\mathbf{l}$ over a special graph. Now, let us define the graph. For simplicity, we assume that $\mathbf{l}$ is a horizontal splitting line (for vertical ones, the following works on the transposed X). We interpret each pixel row $\mathbf{r}$ (a vector concatenating RGB

[1] Building assets are the atomic elements in buildings such as doors, windows, balconies.

values for subsequent pixels) as a node and every two nodes are connected by an edge. The weight of the edge connecting node i and j is:

$$\alpha_{ij} = \exp^{-\left(\frac{\mathrm{d}(\mathbf{r}_i,\mathbf{r}_j)}{\sigma_r}\right)^2} \tag{3}$$

where $\mathrm{d}(\mathbf{r}_i, \mathbf{r}_j) = ||\mathbf{r}_i - \mathbf{r}_j||$ the value of σ_r is set to 10% of the total range of $\mathrm{d}(.,.)$. The $ncut(X^1_\mathbf{l}, X^2_\mathbf{l})$ is then calculated for each possible $\mathbf{l}$ explicitly on this graph. Constructing the graph in this way, the perceptual grouping is forced to conform to the rectilinear structure of façades.

After having defined the edge term and content term, the data support from X for splitting line $\mathbf{l}$ is:

$$\mathrm{support}(X, \mathbf{l}) = S(X, \mathbf{l}) \cdot \Upsilon(X^1_\mathbf{l}, X^2_\mathbf{l}). \tag{4}$$

The splitting line with the strongest data support is chosen for further splitting and such splitting applied recursively leads to a segmentation of the façade.

4 Façade Labelling

Given the training set $\mathcal{D} = \{(X_n, Y_n), n \in \{1, ..., N\}\}$ with N segmented façades X_n and their label maps Y_n, we aim at learning a labelling model which is precise at both pixel level and structural level. Let $\mathbf{x}_{nk}$ denote the visual feature of tile x_{nk} and y_{nk} denote its label with $y_{nk} \in \{1, ..., C\}$. C is the number of labels considered, *e.g.* the 7 categories in Fig. 1. A visual recognition module is trained on all the $\mathbf{x}_{nk}$ from $\mathcal{D}$ and the recognition confidence for tile x is denoted by $P(y = c|\mathbf{x})$. Having defined these terms, we will first identify the meta-features and then define our labelling model, followed by the inference and learning of the model.

4.1 Meta-features for a Good Labelling of Façades

We identify a variety of properties that a good labelling of façades should possess:

- Content: The labels should have strong support from the visual recognizer.
- Boundary: The boundaries of the labels should have strong edge support.
- Similarity: Tiles having the same label should be similar in appearance.
- Balance: The occurring frequency of the categories in the façade should conform to that of the "truth" (obtained from training label maps).
- Neighbours: The occurring frequency of neighbouring relationships should conform to that of the "truth" (obtained from training label maps).
- Shapeness: Assets should come with regular shapes, esp. rectangles.
- Alignment: Some types of assets should be aligned well, such as windows.
- Regularity: Assets of the same types should be evenly spaced, especially for those close to each other.

Though the corresponding meta-features (cf. § 4.2) are quite simple, they can effectively capture the architectural principles and the visual information in image façades, because all of them are non-accidental phenomena, rather resulting from manufacturing, functional, or aesthetic considerations. We do not claim that this list is all-inclusive though, and other useful features can be exploited and added.

4.2 Labelling Model

Now, we combine the aforementioned meta-features into the labelling model:

$$P(Y|X) \propto P(X|Y)P(Y) = \frac{1}{Z}\exp\left(-\sum_{i=1}^{3}\omega_i E_i(X;Y) - \sum_{j=1}^{6}\omega_j E_j(Y)\right) \tag{5}$$

where ω is the relative weight for each energy term, which reflects one of the features identified above, and Z is the partition function. Below, we detail the energy terms for each of the meta-features.

Content. The inferred labels should have strong support from the visual evidence. In particular, it is:

$$E_1(X;Y) = \frac{\sum_{k=1}^{K} E(\boldsymbol{x}_k, y_k)}{K} \tag{6}$$

where $E(\boldsymbol{x}_k, y_k)$ is the energy defined over the visual features and category labels on tile x_k, which is learned by a visual recognizer (See § 5 for details).

Boundary. The boundaries between different categories should agree with prominent edges of the images. For ease of representation, we decompose these rectilinear boundaries into their M linear fragments $\mathbf{b}_m$. Thus, we have:

$$E_2(X;Y) = \frac{-\sum_{m=1}^{M}\sum_{f_{ij}\in\mathcal{X}} \mathrm{s}_{ij}\cdot\phi(i,j,\mathbf{b}_m)\cdot\left(\mathbf{b}_m^{\top}\mathbf{f}_{ij}\right)^2}{\sum_{m=1}^{M}|\mathbf{b}_m|} \tag{7}$$

where $|\mathbf{b}_m|$ is the length of fragment m in terms of pixels, $\mathcal{X}$ is the set of all pixels in image X, and all other terms are the same as in Eq. 1.

Similarity. This energy term is set up to encourage that tiles with the same labels should be similar to each other visually. Formally, it is:

$$E_3(X;Y) = \frac{\sum_{k=1}^{K}\sum_{c=1}^{C} r_{kc}\cdot||\boldsymbol{x}_k - \boldsymbol{\mu}_c||^2}{K} \tag{8}$$

where r_{kc} is 1 if $y_k = c$, and 0 otherwise. $\boldsymbol{x}_k$ is a feature vector describing tile k's appearance (see § 5 for details) and $\boldsymbol{\mu}_c$ is the average of all the $\mathbf{x}_k$ with $y_k = c$.

Balance. Since façades serve practical functions, the occurring frequency of the category labels cannot be arbitrary in each façade. This energy term penalizes the difference between the occurring frequency $\mathbf{o}(Y)$ in the inferred label map Y and the frequency $\bar{\mathbf{o}}$ found in the groundtruth, which is obtained by averaging across all training samples. Specifically, we have,

$$E_1(Y) = \text{dist}(\boldsymbol{o}(Y), \bar{\boldsymbol{o}}) = \sum_{c=1}^{C} |\mathbf{o}(Y)_c - \bar{\mathbf{o}}_c| \cdot \log(\frac{1}{\bar{\mathbf{o}}_c}) \tag{9}$$

where $log(\frac{1}{\bar{\mathbf{o}}_c})$ is an inverse frequency weighting term which is used to attenuate the dominating effect from labels that appear most often in façades.

Neighbours. Some structural aspects of façades are carried by pairwise neighbouring relationships, such as ‘ceiling’ is supported by ‘wall’. This energy term is used to penalize labellings that violate these relationships. We consider four types of neighbouring relationships: above, below, left, and right, for all pairs of categories considered. The occurring frequencies are concatenated as a long vector $\boldsymbol{\tau}$ of dimension $H = 2C^2 + 2C = 4C(C+1)/2$. The penalty term is then based directly on the distance between the vector of inferred labels $\boldsymbol{\tau}(Y)$ and that of the groundtruth $\bar{\boldsymbol{\tau}}$ (again the average over all training labels). It is formulated as:

$$E_2(Y) = \text{dist}(\boldsymbol{\tau}(Y), \bar{\boldsymbol{\tau}}) = \sum_{h=1}^{H} |\boldsymbol{\tau}(Y)_h - \bar{\boldsymbol{\tau}}_h| \cdot \log(\frac{1}{\bar{\boldsymbol{\tau}}_h}). \tag{10}$$

Shapeness. This energy term is designed to penalize the appearance of building assets in Y with irregular shapes. For the sake of simplicity and without loss of generality, we encourage the shape of assets (*e.g.* doors and windows) to be rectangular. For all these assets Λ_a, $a = \{1, ..., A\}$, their boundaries $\partial\Lambda_a$ are extracted on Y and the minimum rectangles R_a that contain the Λ_a are also found. The energy term is based on the distance between $\partial\Lambda_a$ and ∂R_a:

$$E_3(Y) = \frac{\sum_{a=1}^{A} \sum_{f_{ij} \in \partial\Lambda_a} \min_{f_{\kappa\upsilon} \in \partial R_a} \text{d}(f_{ij}, f_{\kappa\upsilon})}{\sum_{a=1}^{A} |\partial\Lambda_a|} \tag{11}$$

where $|\partial\Lambda_a|$ denotes the length of $\partial\Lambda_a$ in pixels, and $\text{d}(\cdot,\cdot)$ is the Euclidean distance between the two pixels.

Alignment and Regularity. The assets of the same type should be placed in an aligned pattern and evenly spaced. The term is only used for the assets with repetition in façades like windows. In order to have enough flexibility while adding this constraint, we introduce the concept of *alignment group.* We encourage the formation of large *alignment groups*, but are still tolerant to the existence of small ones. For simplicity, this term is quantified on the basis of the enclosing rectangle R_a instead of Λ_a. Four ways in which the R_a can be aligned are considered: Left edges share the same horizontal position, 2) or so do right edges, 3) top edges share the same vertical positions, 4) or so bottom edges.

The *alignment groups* are identified by a very simple greedy searching method, applied directly to edge positions. Two edges of the same type (i.e. both left, right, top, or bottom) are put into one group if their 'shared' positions are within a specified distance ε ($\varepsilon = 15$ pixels in this paper). For each asset category considered here, the method runs four times, once for each of the four alignment ways. The energy terms are then defined as:

$$\underbrace{E_4(Y) = \frac{\sum_{g\in\mathcal{G}} \sigma_a(g)}{\#(\mathcal{G})}}_{\text{alignment}}, \underbrace{E_5(Y) = \frac{\sum_{g\in\mathcal{G}} \sigma_r(g)}{\#(\mathcal{G})}}_{\text{regularity}}, \underbrace{E_6(Y) = \#(\mathcal{G})}_{\text{group number}} \tag{12}$$

where $\mathcal{G}$ is the set of all the *alignment groups* found, $\sigma_a(g)$ denotes the standard deviation of edge locations in group g and $\sigma_r(g)$ denotes the standard deviation of the gaps between adjacent edges in group g.

With the labelling model in place, we now can describe the inference and the learning method.

4.3 Inference

Given a façade X and the parameter vector $\boldsymbol{\omega}$, the optimal labelling Y^* can be determined by the optimization $Y^* = \text{argmax}_Y P(Y|X)$. Since also some features from Y are involved, the maximization cannot be obtained analytically. Here, we adopt the Swendsen-Wang Cut (SWC) [22] method for efficient sampling. To obtain Y^* by the SWC, the next 3 steps are iterated until convergence.

(i) *Graph Construction* An adjacency graph $G =< V, E >$ is constructed in this step, where $V = \{v_1, v_2, ..., v_K\}$ is the set of nodes (the segmented tiles), and E is the set of edges connecting neighboring tiles. Each edge $e \in E$ is associated with a Bernoulli random variable $\mu_e \in \{on, off\}$ indicating whether the edge is turned on or off, and a weight reflecting the possibility of doing so. In this work, for each edge $e =< v_i, v_j >$, we define its turn-on probability as:

$$q_e = p(\mu_e = on|\boldsymbol{x}_i, \boldsymbol{x}_j) = \exp\{-(KL(\boldsymbol{x}_i, \boldsymbol{x}_j) + KL(\boldsymbol{x}_j, \boldsymbol{x}_i))T/2\}, \tag{13}$$

where $KL(\cdot, \cdot)$ denotes the KL divergence and T is a temperature factor.

(ii) *Graph Clustering.* Given the current label map, it removes all edges between tiles of different categories. Then all the remaining edges are turned on independently with the probability q_e. Thus, we have a set of connected components (CCPs) $\varPi$'s, in which all tiles have the same category label.

(iii) *Graph Flipping.* It randomly selects a CCP $\varPi_i$ from the set formed in step (ii) with a uniform probability, and then flips the labels of all tiles in $\varPi_i$ to category $c \in \{1, 2, ..., C\}$ with the probability:

$$P(\varPi_i = c) = \frac{1}{\#(\varPi_i)} \sum_{v_k \in \varPi_i} P(y_k = c|\boldsymbol{x}_k), \tag{14}$$

where $P(y_k = c|\boldsymbol{x}_k)$ is the score of visual recognition. The flip is accepted with probability:

$$P(Y \to Y') = \min(1, \frac{Q(Y' \to Y)P(Y'|X)}{Q(Y \to Y')P(Y|X)} \tag{15}$$

where $Q(Y' \rightarrow Y)$ and $Q(Y \rightarrow Y')$ are proposal probabilities for state jumping and their ratio $\frac{Q(Y' \rightarrow Y)}{Q(Y \rightarrow Y')}$ can be derived as Theorem 2 in [22] shows.

Algorithm 1. Learning Algorithm

```
Input:
— training set $\mathcal{D} = \{(X_n, Y_n)_{n=1}^{N}\}$
— empty set of competing low energy labellings: $\mathcal{S} = \emptyset$
— initial parameters: $\boldsymbol{\omega} = \boldsymbol{\omega}_0$
Repeat until $\boldsymbol{\omega}$ is unchanged
for $n = 1 \rightarrow N$ do
    1. Find the optimal labelling of sample $n$ using the inference described in § 4.3:
    $Y^* = argmax_Y \; P(Y|X)$
    2. Add $Y^*$ to the constraint set: $\mathcal{S}_n \leftarrow \mathcal{S}_n \bigcup \{Y^*\}$
    3. Update $\boldsymbol{w}$ to maximize the energy margin between training label maps and
    competing label maps:
        $\min_{\boldsymbol{\omega}} \frac{1}{2}||\boldsymbol{\omega}||^2 \quad \text{such that} \qquad (16)$
        $E(X_n, Y) - E(X_n, Y_n) \geq 1 \quad \forall Y \in \mathcal{S}_n \quad \forall n.$
end for
```

4.4 Learning

Given the training set $\mathcal{D}$, we are looking for the parameter vector $\boldsymbol{w}$ so that,

$$E(X_n, Y_n) \leq E(X_n, Y) \;\; \forall Y \neq Y_n \;\; \forall n. \tag{17}$$

where $E(X, Y) = -\text{log}P(X, Y)$. Since the inequalities in (17) may have no solution or have multiple solutions, we follow [23] to introduce an energy margin γ so that the inequalities are satisfied with the largest margin. The max-margin concept provides robust solutions for the inequalities in (17). By introducing the margin, (17) can be rewritten as:

$$\begin{gathered} \max_{\boldsymbol{\omega}:||\boldsymbol{\omega}||=1} \gamma \quad \text{such that} \\ E(X_n, Y) - E(X_n, Y_n) \geq \gamma \quad \forall Y \neq Y_n \quad \forall n \end{gathered} \tag{18}$$

where the $\boldsymbol{w}$ has been constrained to have a unit norm so that the weights cannot diverge to arbitrary large values. Using the transformation $||\boldsymbol{\omega}|| \leftarrow \frac{1}{\gamma}$, (18) can be expressed as a standard quadratic optimization problem:

$$\begin{gathered} \min_{\boldsymbol{\omega}} \frac{1}{2}||\boldsymbol{\omega}||^2 \quad \text{such that} \\ E(X_n, Y) - E(X_n, Y_n) \geq 1 \quad \forall Y \neq Y_n \quad \forall n. \end{gathered} \tag{19}$$

It is infeasible to optimize the problem in (19) directly, as there are an exponential number of Ys. In this work, we adopt the method developed in [23] to find the solution, in which only a small set of "promising" Y need to be examined. The learning process is very similar to that described in Fig. 1 of [23]. However, in order to be self-contained, we summarize it in Alg. 1. As long as the SWC inference method gets the optimal solution (it does when given enough time), this learning method can learn the optimal parameters [23].

5 Experiments

5.1 Experiment Setup

We evaluated our method on the Ecole Centrale Paris (ECP) Façade Database [24]. This dataset contains 104 images of rectified and cropped façades of Haussmannian style. 7 different categories are considered in the dataset: window, wall, balcony, door, roof, sky and shop. As the annotations of some images in the dataset is imprecise, we chose 80 images out of 104 for our experiments, with 40 images (randomly selected) for training and the rest for test. All reported results represents averages over 10 training-test partitions. Randomized Forests (RFs) [25] were adopted as the visual recognizer, and an ensemble of 10 decision trees were trained. Since the semantic categories appearing in façades are imbalanced (e.g. doors are less frequent than windows), we adopted the scheme proposed in [5] of balancing the categories to avoiding unexpected bias towards some categories. For this dataset, the shapeness term is applied for doors, windows and balconies. The alignment and regularity terms are applied for windows and balconies. The λ in Eq. 1 is set to 0.2 as the agreement is more important than the disagreement. Each façade is segmented into $K = 200$ tiles.

In order to evaluate our method, we compare against 5 competing methods: 1) RFs directly on densely-sampled patches (RFs(P)); 2) RFs directly on superpixels (RFs(S)); 3) RFs directly on the tiles from our segmentation method (RFs(T)); 4) the state-of-the-art scene labelling method RNN [19]; and 5) the state-of-the-art façade labeling method via a shape grammar (SG) [17]. For the densely-sampled patches, we used the optimal patch size of 16×16 pixels with step size of 4 pixels (obtained empirically). For the superpixels, we used the method of [21] with the default parameters. As to the feature representation, though Eq. 6, Eq. 8, and Eq. 13 use the same notation $\boldsymbol{x}$, speed-accuracy trade-offs mean that different features may be optimal in each stage. We adopted a 48-bin color histogram as the $\boldsymbol{x}$ for Eq. 8 and Eq. 13, with 16 bins for each of the RGB color channels. For the $\boldsymbol{x}$ of Eq. 6 (i.e. the features for RFs), we used a concatenation of various features: PHOG [26] with a 2-layer pyramid, PACT [27] with a 2-layer pyramid, the 48-bin color histogram, and the vertical position of the tile in the image. RFs(P) and RFs(T) used the same feature set as our method did. RFs(S) and the RNN used the superpixel features designed in [4] for image labelling on superpixels.

5.2 Results

Table 1 lists the labelling performance of all the methods. In the table, we compare all the methods in terms of the precision of separate categories as

Table 1. The labelling accuracy on the ECP Façade dataset. The top panel compares our method with others on individual categories and the bottom on the overall performance. Numbers in green denote the superior of our method, and red for the reverse. Average(p) and average(c) denote the average precision over pixels and over categories respectively.

	Our method							RFs(P)	RFs(S)	RFs(T)	RNN[19]	SG[24]
window	**72**	11	9	3	5	0	0	+14	+15	+12	+12	+10
wall	6	85	6	0	2	0	1	+15	+13	+9	-3	+3
balcony	14	10	**71**	0	5	0	0	+7	+10	+4	+9	+12
door	6	4	5	**65**	0	0	20	+8	+12	+7	+17	+15
roof	7	6	5	0	80	2	0	+17	+18	+9	+9	-3
sky	2	1	0	0	4	93	0	+0	-2	+0	-3	-4
shop	0	2	0	6	0	0	**92**	+34	+30	+15	+9	+1
average(p)	**83.0**							67.5	68.7	74.3	79.8	78.9
average(c)	**80.1**							67.0	67.0	73.0	72.3	75.1
	(a)							(b)	(c)	(d)	(e)	(f)

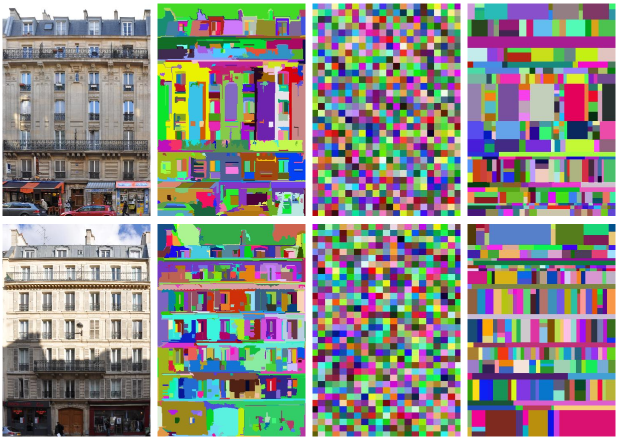

(a) (b) (c) (d)

Fig. 2. Examples of façade segmentation on Haussmannian buildings from the ECP façade dataset: (a) two façades, (b) segmentation results by [21], (c) results by densely-sampling, and (d) partitioning results of our method (Best viewed in color)

well as of the overall performance. For the overall performance, we calculated the average precision over pixels (average(p)) and the average precision over categories (average(c)). Table 1 shows that our method outperforms all other methods in overall precision and in the precision of most individual categories, even compared with the SG method [24], which is provided with additional information through a style-specific grammar. The reason for the better results is that the labelling model learned on the identified feature set, maintains the structural correctness of façades and resolves the visual ambiguities between categories. Table 1 also shows that our method performs specially well in the

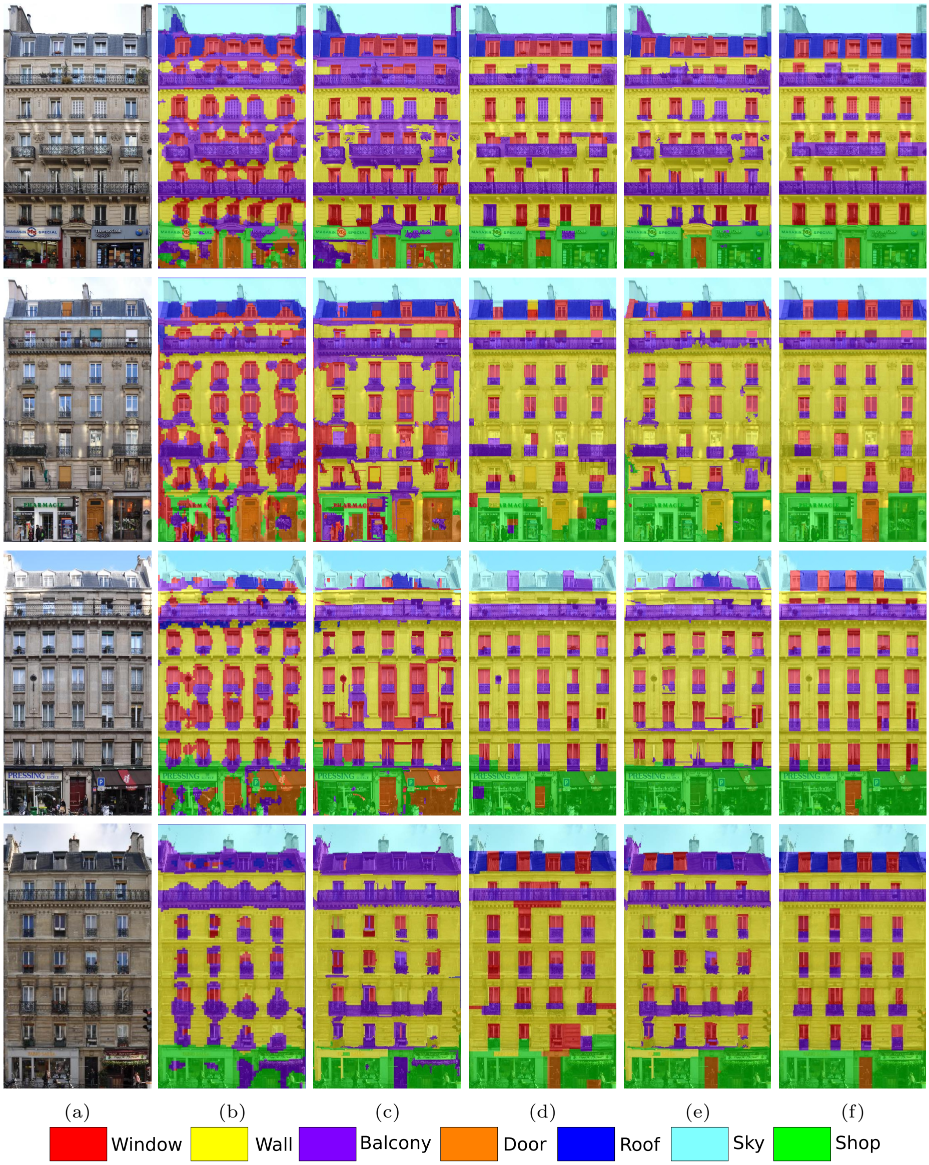


Fig. 3. Examples of façade labelling for Haussmannian buildings from the ECP façade dataset: (a) four façades, (b) results by RFs(P), (c) results by RFs(S), (d) results by RFs(T), (e) results by RLP [4] and (f) results by our method (Best viewed in color)

categories 'windows', 'doors' and 'balconies'. Though occupying the smallest areas in façades, the above categories are critical to maintain the structural correctness of façades. Our method benefits from its configurational features, while still making more lenient assumptions than imposed by a grammar.

Another reason that our labelling method is superior can be attributed to the good performance of our segmentation method. Fig. 2 shows two examples of the segmentation on the ECP façade dataset. For comparison, we also show the results of the method [21] and of dense sampling. From Fig. 2, it is evident that principled segmentation methods like [21] and dense sampling lose most of the structural information at an early stage and their final segments rarely correspond to the atomic elements of façades. In contrast, our method maintains the structure very well and segments out most of the atomic elements. The benefits of this for labelling can be seen by comparing the performance of RFs(P), RFs(S) and RFs(T) in Table 1.

In order to show how good our method is at maintaining the structural integrity of façades, we illustrate the labelling results of four façades from the ECP façade dataset in Fig. 3. From the figure, one can see that our method has a good performance both in terms of pixelwise and structural precision. If we compare Fig. 3 (d) and (f), it is evident that the learned domain knowledge is able to correct most of the erroneously inferred labels. Another conclusion we can draw is that our method yields cleaner structures. This makes it much easier to extract procedural rules from the labelling results for building modelling [1].

6 Conclusion

This paper has tackled the problem of precise façade labelling without using any prior knowledge. In order to guarantee the structural correctness and avoid visual ambiguities, we leveraged the power of architectural principles embedded in façades. We presented a simple yet effective façade segmentation method to segment façades into a bunch of compact tiles, so that visual recognition can be performed at suitable scales and the architectural principles can be extracted easily. A set of meta-features were identified to capture both visual information and the architectural principles, and were used to train a precise façade labelling model. Experiments show that our method is more precise than the state-of-the-art at both pixel level and structural level. Our long-term goal is to automatically extract grammar rules from the labelling results and recreate detailed 3D building models with the rules. This is quite straightforward for our method as the labelling results have cleaner structures and can easily be modeled by procedural rules.

Acknowledgement. We gratefully acknowledge support by Future Cities Laboratory in the Singapore-ETH Center, EC Integrated Project 3D-Coform, and ERC Advanced Grant VarCity.

References

1. Müller, P., Wonka, P., Haegler, S., Ulmer, A., Gool, L.V.: Procedural modeling of buildings. In: SIGGRAPH (2006)
2. Müller, P., Zeng, G., Wonka, P., Gool, L.V.: Image-based procedural modeling of facades. In: SIGGRAPH (2007)

3. Shotton, J., Winn, J.M., Rother, C., Criminisi, A.: *TextonBoost*: Joint Appearance, Shape and Context Modeling for Multi-class Object Recognition and Segmentation. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3951, pp. 1–15. Springer, Heidelberg (2006)
4. Gould, S., Rodgers, J., Cohen, D., Elidan, G., Koller, D.: Multi-class segmentation with relative location prior. IJCV 80, 300–316 (2008)
5. Shotton, J., Johnson, M., Cipolla, R.: Semantic texton forests for image categorization and segmentation. In: CVPR (2008)
6. Tighe, J., Lazebnik, S.: SuperParsing: Scalable Nonparametric Image Parsing with Superpixels. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part V. LNCS, vol. 6315, pp. 352–365. Springer, Heidelberg (2010)
7. Berg, A.C., Grabler, F., Malik, J.: Parsing images of architectural scenes. In: ICCV (2007)
8. Zhao, P., Fang, T., Xiao, J., Zhang, H., Zhao, Q., Quan, L., Buaa, V.: Rectilinear parsing of architecture in urban environment. In: CVPR (2010)
9. Wendel, A., Donoser, M., Bischof, H.: Unsupervised Facade Segmentation Using Repetitive Patterns. In: Goesele, M., Roth, S., Kuijper, A., Schiele, B., Schindler, K. (eds.) DAGM 2010. LNCS, vol. 6376, pp. 51–60. Springer, Heidelberg (2010)
10. Shen, C.H., Huang, S.S., Fu, H., Hu, S.M.: Adaptive partitioning of urban facades. In: SIGGRAPH Asia (2011)
11. Xiao, J., Fang, T., Tan, P., Zhao, P., Ofek, E., Quan, L.: Image-based façade modeling. In: SIGGRAPH Asia (2008)
12. Xiao, J., Fang, T., Zhao, P., Lhuillier, M., Quan, L.: Image-based street-side city modeling. In: SIGGRAPH Asia (2009)
13. Dick, A., Torr, P., Cipolla, R.: Modelling and interpretation of architecture from several images. IJCV 60, 111–134 (2004)
14. Li, Y., Sharf, A., Cohen-or, D., Chen, B.: 2d-3d fusion for layer decomposition of urban facades. In: ICCV (2011)
15. Musialski, P., Wimmer, M., Wonka, P.: Interactive coherence-based facade modeling. In: Eurographics (2012)
16. Teboul, O., Simon, L., Koutsourakis, P., Paragios, N.: Segmentation of building facades using procedural shape priors. In: CVPR (2010)
17. Teboul, O., Kokkinos, I., Koutsourakis, P., Paragios, N.: Shape grammar parsing via reinforcement learning. In: CVPR (2011)
18. Tu, Z.: Auto-context and its application to high-level vision tasks. In: CVPR (2008)
19. Socher, R., Lin, C.C., Ng, A.Y., Manning, C.D.: Parsing Natural Scenes and Natural Language with Recursive Neural Networks. In: ICML (2011)
20. Shi, J., Malik, J.: Normalized cuts and image segmentation. PAMI 22, 888–905 (2000)
21. Felzenszwalb, P.F., Huttenlocher, D.P.: Efficient graph-based image segmentation. IJCV 59, 167–181 (2004)
22. Barbu, A., Zhu, S.C.: Generalizing swendsen-wang to sampling arbitrary posterior probabilities. PAMI 27, 1239–1253 (2005)
23. Szummer, M., Kohli, P., Hoiem, D.: Learning CRFs Using Graph Cuts. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part II. LNCS, vol. 5303, pp. 582–595. Springer, Heidelberg (2008)
24. Teboul, O.: Shape Grammar Parsing: Application to Image-based Modeling. PhD thesis, Ecole Centrale Paris (2011)
25. Breiman, L.: Random forests. Machine Learning 45, 5–32 (2001)
26. Bosch, A., Zisserman, A.: Bosch, A., Zisserman, A., Muñoz, X.: Image classification using random forests and ferns. In: ICCV (2007)
27. Wu, J., Rehg, J.: Where am i: Place instance and category recognition using spatial pact. In: CVPR (2008)