

Specification of Windowpane IPM for Use in SCENIC

Jan Čech

cechj@cmp.felk.cvut.cz

tel. +420 2435 5726

Center for Machine Perception
Prague, Czech Republic

TN-eTRIMS-CMP-02-2006

version 1.0

January 16, 2007

This document describes basic functionality of the windowpane detector for easier incorporation of this image processing module (IPM) into the SCENIC system. We are using a language defined in [2] which should be in accordance with [1].

1 Principle of the detector

The windowpane detector is formulated as maximum a posteriori labeling problem. The algorithm finds (approximately) the most probable labeling in the language of non-overlapping rectangles.

Each pixel $t \in T$ of the input image has one of possible labels $x \in X = \{E, I, L, R, B, T, TL, TR, BL, BR\}$, Fig. 1. There are rules defining configurations of neighbouring labels which are allowed, such that the labels always form a set of non-overlapping rectangles. For instance, right to T only another T or TR is allowed.

The IPM solves the following problem

$$\mathbf{x}^* = \arg \max_{\mathbf{x} \in X^{|T|}} p(\mathbf{x}|\mathbf{d}), \quad (1)$$

where \mathbf{x} is the image labeling and \mathbf{d} is the (image) data. Using the Bayes law and the assumption of independence, this leads to

$$\mathbf{x}^* = \arg \max_{\mathbf{x} \in X^{|T|}} \frac{p(\mathbf{d}|\mathbf{x})p(\mathbf{x})}{p(\mathbf{d})} = \arg \max_{\mathbf{x} \in X^{|T|}} \prod_t p(d|x_t) \prod_{t,t'} p(x_t, x_{t'}) \prod_t p(x_t), \quad (2)$$

E	E	E	E	E	E	E	E	E
E	E	TL	T	T	T	TR	E	E
E	E	L	I	I	I	R	E	E
E	E	L	I	I	I	R	E	E
E	E	L	I	I	I	R	E	E
E	E	L	I	I	I	R	E	E
E	E	BL	B	B	B	BR	E	E
E	E	E	E	E	E	E	E	E

Figure 1: Labeled rectangle. Labels are marked as characters at pixels: E - external label (represents a facade), I - internal windowpane label, L, R, T, B are left, right, top, bottom edge of the windowpane respectively and TL, TR, BL, BR are its corners.

where the term $p(d|x)$ is the *image model*, it is a probability distribution of the image given the label. It is a statistical model of pixel color and edgeness for each of the labels, e.g. the windowpane label (I-label) is usually of dark or glossy color with low edgeness. It is learnt from annotated images.

The term $p(x, x')$ is the *structure model*, it represents the neighbouring label compatibility, i.e. the rules that generate a language of non-overlapping rectangles. These rules can be equivalently written as a 2D grammar. This structure of the model is fixed at present, its parameters (transition probabilities) are learnt from annotated results.

The term $p(x)$ stands for the *external prior* probability of the label. It can be set from an external process, e.g. the SCENIC system may deduce that in some region there should be a window, or must not be a window else. By default it is kept of a uniform probability distribution over all 10 labels.

Applying logarithm to (2), we obtain the max-sum labeling problem

$$\mathbf{x}^* = \arg \max_{\mathbf{x} \in X^{|T|}} \sum_t g_t(x_t) + \sum_{t, t'} g_{tt'}(x_t, x_{t'}), \quad (3)$$

where

$$\begin{aligned} g_t(x_t) &= \log p(d|x_t) + \log p(x_t), \\ g_{tt'}(x_t, x_{t'}) &= \log p(x_t, x_{t'}). \end{aligned} \quad (4)$$

We use Tomas Werner’s implementation of the max-sum labeling problem solver, [3].

2 Operator description

The default entities this IPM can handle are windowpanes. They are internally represented in the language of 10 labels. The external representation is only binary (windowpane and non-windowpane) such that the windowpane regions are axis parallel non-overlapping rectangles.

The output of the detector is a list of rectangular image regions of windowpanes with associated confidence values C_R . The confidence is a real number

which is the higher the better the model explains the data. It is simply the sum of the labeling in the windowpane region (and a 2-pixel neighbourhood) R normalized by the area of the region

$$C_R = \frac{1}{|R|} \left(\sum_{t \in R \subset T} g_t(x_t^*) + \sum_{t, t' \in R \subset T} g_{tt'}(x_t^*, x_{t'}^*) \right). \quad (5)$$

There is another confidence which we call overall energy. It reflects how the model explains the entire image. It is the sum of the labeling over the whole image. It is the value of the maximum in (3), where the exponential of this value is the posterior probability, (1), up to a constant.

Prerequisites to run the IPM

The whole IPM currently works on rectified images of facade scenes only. The usage of the IPM on non-rectified images is discouraged, since the windowpanes are not projected as axis-parallel rectangles, but due to perspective distortion as general quadrilaterals.

Provided interface operators

init Creates options structure filled with default values. This structure can be modified and is used as one of the attributes to the operators. It controls internal parameters of the operators. This operator is called at system startup or during re-initialization.

find object Given an image, image model, structure model, and a region of interest, it finds all windowpanes.

find object with external prior Given an image, image model, structure model, region of interest, and external prior, term $p(x)$, Eq. (2), (from SCENIC) to enforce or attenuate windowpane evidence at specified image locations, it finds all windowpanes.

The external prior term is expected as a list of axis-parallel rectangular locations in the image with label log-probabilities.

$$E = \{(r_t, c_l, r_b, c_r, x | \log p(x))\}, \quad (6)$$

where r_t, c_l are row, column image coordinate of the upper left corner of the axis-parallel rectangle, r_b, c_r row, column image coordinate of the lower right corner, $x \in X = \{E, I, L, R, B, T, TL, TR, BL, BR\}$ is the label identifier. The $\log p(x)$ is the logarithm of the prior probability of the label, (2). The values of $\log p(x)$ must be from interval $[-\infty, 0]$. The smaller it is the more the label is suppressed. Extreme cases are: $\log p(x) = -\infty$ means the label is forbidden, and $\log p(x) = 0$ means the solution must contain the label. For undefined locations the external prior is set automatically to a uniform distribution over the undefined labels.

Type	Description
filename	Name of (usually) binary file (used for images, data files, etc.)
XMLfilename	Name of a text file in XML format
structure	Structure containing several fields
number	Numerical value
rectangle	Description of a rectangle
string	Text string

Table 1: Operator parameter types

get entity confidence Given an image, image model, structure model, and hypothesized windowpane positions (by SCENIC), it returns their confidence values.

get most probable image Given hypothesized windowpane positions and an image model, it returns the most probable image according to the image model.

learn image model Given a set of images and hypothesized windowpane positions (annotated manually or by SCENIC), it learns the statistical image model, term $p(d|x)$, Eq. (2). Negative examples, i.e. entities which are not windowpanes, may also be set. If not, complement of the windowpanes to the image is considered as a facade - negative example.

learn structure model Given hypothesized windowpane positions in several images (annotated manually or by SCENIC), it learns parameters of probabilities of the statistical structure model, term $p(x, x')$, Eq. (2), transition probabilities of 2D grammar.

draw entities Given hypothesized windowpane positions, it draws rectangles into an image. It is for visualization purposes only.

shutdown This operator is called at system shutdown. It currently has null functionality.

What parameters can be given to operators

Parameters given to the operators can be of several types listed in Table 1. In the following operator list, input and output parameters are specified. The parameter types are indicated in the parenthesis behind the parameters.

The entity description contained in the XML files have to contain 1. entity as a rectangular polygon, 2. object type (windowpane in this case), and 3. image filename to which the entity is related to (each entity can be from different image!). A star (*) next to the XMLfilename type indicates that the entities in the file have also a confidence value tag. The XML file DTDs are the same as in the Adaboost IPM [2], they are defined in the appendix there.

Each operator returns the error status besides the result. If some error occurs, an error message string is returned, otherwise the error status is null.

init

Input: configuration file name (filename)
Output: default options (structure), error status (string)

find object

Input: rectified image (filename), image model (filename), structure model (filename), region of interest (rectangle), options (structure)
Output: hypothesized windowpanes (XMLfilename*), overall energy (number), error status (string)

find object with external prior

Input: rectified image (filename), image model (filename), structure model (filename), external prior term (filename), region of interest (rectangle), options (structure)
Output: hypothesized windowpanes (XMLfilename*), overall energy (number), error status (string)

get entity confidence

Input: rectified image (filename), image model (filename), structure model (filename), hypothesized windowpane positions (XMLfilename), options (structure)
Output: hypothesized windowpanes (XMLfilename*), overall energy (number), error status (string)

get most probable image

Input: hypothesized windowpane positions (XMLfilename), image model (filename), options (structure)
Output: most probable image according to a current statistical model (filename), error status (string)

learn image model

Input: hypothesized windowpane positions (XMLfilename), negative examples (XMLfilename), options (structure)
Output: image model (filename), error status (string)

learn structure model

Input: hypothesized windowpane positions (XMLfilename), options (structure)
Output: structure model (filename), error status (string)

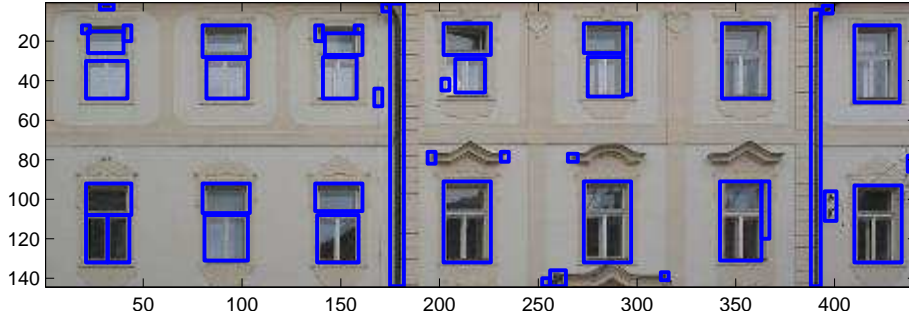


Figure 2: The result of **find windowpanes** operator.

draw entities

Input: rectified image (filename), hypothesized windowpane positions (XMLfilename), options (structure)

Output: image with entities sketched (filename), error status (string)

shutdown

Input: none

Output: error status (string)

3 Examples

Let us have a simple scenario. The task is to detect all windowpanes in the image of a facade.

First, call the **init** operator to initialize the system.

Then, call the **find windowpanes** operator to detect the windowpanes. The result is in Fig. 2. We can see that most of the windows are detected, however the 4th window from left in the top row is misdetections.

This result comes from the optimal labeling which is consistent with the data. Higher level information (given by the SCENIC system) can be incorporated into the detector. The SCENIC gets the result of the previous step and may suggest to increase the likelihood of the windowpane label in the suspicious region. Then **find windowpane with external prior** operator with prior term E is called. It is set $E = \{(30, 205, 45, 210, 'I' | 0)\}$, the rest of E is left undefined. It means the I-label, internal windowpane label) is forced in the rectangular area: rows from 30 to 45, columns from 205 to 210. The detector solves the updated optimization problem and the result of the detector is then in Fig. 3.

Further, the SCENIC may deduce, that all windows in one row should be the same. For example all windows in the bottom row, should be similar to the first three windows from the left. Therefore the SCENIC suggests to increase the likelihood of the facade label (non-windowpane label) in the whole horizontal line going in between the windowpanes there to enforce the windows to break into two windowpanes. The **find windowpane**



Figure 3: The result of **find windowpanes with external prior** operator to support one misdetectd window.



Figure 4: The result of **find windowpanes with external prior** operator to support breaking of the windows in the lower row.

with external prior operator with prior term E is called. It is set $E = \{(30, 205, 45, 210, 'I' | 0), (108, 1, 109, 441, 'I' | -100)\}$. The first element is the same as in the previous case, the second element means the I-label is suppressed there by setting a very low prior probability. The rest is again left undefined. The result is in Fig. 4. Each window in the bottom row really broke into two windowpanes. Note that, the vertical pipes stayed unbroken, which demonstrates this is not a hard enforcement of the label, but really a soft decrement of the likelihood and solving the optimization problem.

Similarly, the false detection which are not windowpanes can also be filtered out by the SCENIC, e.g. from their low confidence, small size (remember, we have the image scale), wrong aspect ratio, irregularity of their alignment, etc.

Finally **shutdown operator** is called.

4 System requirements

- The IPM requires the Matlab environment with the Image Toolbox installed. It has been developed under Matlab 7.2. but is easy to upgrade

for a newer version or downgrade for an older version (not below 5.0).

- Parts of the module are written in C++, so also the compiler – gcc for Linux or Visual .NET for Windows operating system – is needed. Each Matlab version requires different gcc version so the appropriate one has to be used. The Visual .NET version 7.1 has been tested.
- Both Windows XP/2000/NT and Linux operating systems are supported. Windows XP and Gentoo Linux have been tested but the IPM should be easily transferred to similar operating systems.
- The IPM does not require any additional libraries.
- Expected computation time for detection strongly depends on the complexity of the scene. If the data are in a good agreement with structure model, it is usually very fast (few seconds for facade images 500×500 pixels). It is not recommended to run this IPM on large images which contain other scene structure beside the facade, e.g. trees, cars, roofs, etc., without setting a region of interest to focus to the facade.

References

- [1] Lothar Hotz. Template for describing image processing modules. Technical notes, Hamburg Informatics Technology Center, November 2006.
- [2] Jan Sochman. Specification template of the AdaBoost IPM for use in SCENIC. Technical notes TN-eTRIMS-CMP-01-2006, Center for Machine Perception, December 2006.
- [3] Tomáš Werner. An approach to max-sum problem based on equivalence and triviality. Research Report CTU–CMP–2005–25, Center for Machine Perception, K13133 FEE Czech Technical University, Prague, Czech Republic, October 2005.