Fully Bottom-Up Blob Extraction in Building Facades

M. Jahangiri, M. Petrou

Imperial College London, Communication and Signal Processing Group, maria.petrou@imperial.ac.uk

A fully bottom-up algorithm for extracting blobs in building facades is proposed. Prototype discovery, hypothesis testing and verification are used to identify missing blobs or split the agglomerated ones, when repeated patterns are present in the facade. A novel similarity measure is used for assessing the similarity of subpatterns. The proposed algorithm can detect 80% of the 3589 hand segmented components in 300 images.

1. Introduction

The eTRIMS[1] project is concerned with the automatic identification of building parts, with the ultimate aim to be incorporated in a tool that constructs 3D models of cities. In this paper, we present a methodology for identifying the subparts of a building facade, including windows, doors, chimneys, balconies, dormers and canopies as blobs that will then be presented to a higher level module for classification. For this purpose, our system is totally automatic and totally bottom up.

2. Methodology

In the first stage we apply an interest operator for enhancing blob-shaped structures in the image. We use thresholding, morphological operators and connected component analysis to extract blobs from the constructed map. The first step of the algorithm is fully automated, but it constitutes a straight forward image processing methodology and we do not discuss it here due to lack of space. Two example outputs of the results of the preliminary blob extraction are shown in Fig 1.

The main contribution of this paper is in the steps which use the preliminary regions as input and automatically recognize the presence of repeated structures and subsequently identify the missed ones or split the agglomerated ones. In the following sections we present these steps in more detail.

[1] http://www.ipb.uni-bonn.de/projects/etrims/

Fig 1: Examples of the output of the first step of the algorithm. Note that some parts are missed or agglomerated.

2.1. Prototype Discovery

Our task here is to recognize automatically the presence of repeated structures. To achieve that, we first create a binary map that contains all the extracted blobs. Then we perform a sort of autocorrelation as follows.

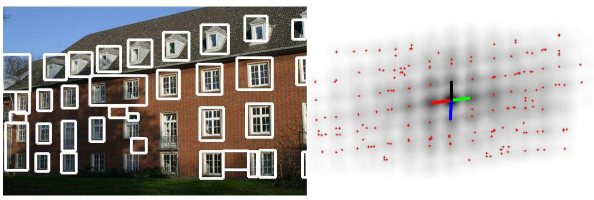
1. If the image is $N \times M$, create an empty grid of size $2N \times 2M$, say *Cor*.
2. Shift the binary image so as the centre of each extracted region coincides with the centre of *Cor* in turn, and accumulate the values
3. Identify the peaks of matrix *Cor*. Each peak in association with the centre of the grid constitutes a possible prototype.

The *Cor* matrix which is created in this way is very noisy. This is due to various reasons including i) fragments of regions of interest extracted instead of full regions, ii) incorrect blobs extracted and iii) perspective effects. So, for peaks to be identified in these maps, some smoothing has to be first applied. From the blobs which have already been extracted, we create the histogram of the areas of the extracted regions and identify the mode of the area. The mode area is defined as the average area of the blobs that fall in the most populated bin. Let us call it A_m. We then use an averaging window of size $\lfloor\sqrt{A_m}+0.5\rfloor\times\lfloor\sqrt{A_m}+0.5\rfloor$ to smooth the corresponding *Cor* map.

After smoothing the *Cor* map, the local maxima are identified. We search for different local maxima in four different directions by considering the local maxima in 4 non-overlapping angular bins with 22.5° tolerance, and with respect to the centre of matrix *Cor*, about the horizontal and vertical mean directions. Let us consider two vectors, say $\vec{a}$ and $\vec{b}$ which are emanating from the centre of matrix *Cor*, pointing to the nearest local maxima with respect to the centre. These vectors are accepted as a discovered prototype if:

$$\frac{\left|\vec{a}\cdot\vec{b}\right|}{\left|\vec{a}\right|\cdot\left|\vec{b}\right|}>T_1 \quad \text{and} \quad \frac{\left|P_{\vec{a}}-P_{\vec{b}}\right|}{\left|P_{\vec{a}}+P_{\vec{b}}\right|}<T_2 \qquad (4)$$

where $P_{\vec{a}}$ and $P_{\vec{b}}$ are the values of the points of matrix Cor to which vectors $\vec{a}$ and $\vec{b}$ are pointing. In our experiments we chose T_1 and T_2 to be equal to 0.9 and 0.1, respectively. These criteria mean that the central point and a neighboring maximum are accepted as forming a prototype if there is a symmetrically placed maximum of roughly equal strength on the other side of the central point. Discovered prototypes from the extracted regions of image shown in Fig 4a is shown in Fig 2b. It is note worthy to underline that in the case that there is no prototype in the extracted regions, we do not proceed and no more regions may be extracted from the building facade.



(a) (b)

Fig 2: The discovered prototypes of the extracted regions in (a) are shown by the lines emanating from the centre of the Cor map in (b).

2. 2 Hypothesis Generation and a new Region Similarity Measure

A prototype we discovered from the extracted regions may be used to locate regions in the original image similar to those that have been already extracted. In other words, once we know where to look to find a region similar to one we already identified, we have to verify its existence by using a similarity measure. We consider that the images we wish to compare may be dissimilar in some details, but similar in their overall appearance. For example, we would like to discover a hypothesized window paired through prototype discovery with an already identified one, even if this second window is open or has its shutter closed or has a curtain. So, what we would like to compare between an already identified region and a hypothesized one is the basic appearance rather than the detail.

Therefore, we introduce a similarity measure based on the eigenimages of two subimages. We first decompose the two regions, which we want to compare, into their eigenimages, using singular value decomposition (SVD) [2]. Two subimages, say I_1 and I_2 that we wish to compare may be written as:

$$I_1=\sum_{i=1}^{r_1}\underbrace{\sqrt{\lambda_1^i}\,\vec{u}_1^i\vec{v}_1^{i^T}}_{E_1^i},\ I_2=\sum_{i=1}^{r_1}\underbrace{\sqrt{\lambda_2^i}\,\vec{u}_2^i\vec{v}_2^{i^T}}_{E_2^i} \qquad (5)$$

where λ_1^i and λ_2^i are the eigenvalues of square matrices $I_1I_1^T$ and $I_2I_2^T$, respectively, and u_1^i and u_2^i are the corresponding eigenvectors. Parameters r_1 and r_2 are the ranks of the same matrices, and v_1^i and v_2^i are the eigenvalues of $I_1{}^TI_1$ and $I_2{}^TI_2$. The eigenvalues in (5) are arranged in decreasing order, so successive approximations of the two subimages may be obtained by truncating these expansions by keeping only the first few terms. The similarity measure which we propose is:

$$SIM(I_1,I_2)=\frac{\sum_{i=1}^{k}\sqrt{\lambda_1^i\lambda_2^i}\,e^{-\alpha\left\|E_1^i-E_2^i\right\|}}{\sum_{i=1}^{k}\sqrt{\lambda_1^i\lambda_2^i}} \qquad (6)$$

where $\|\cdot\|$ is the norm operator. This criterion is bounded between 0 and 1 and the higher the value of *SIM* the more similar the two sub-images are expected to be. Parameter α is used in order to avoid the saturation of the exponential function. Some pilot experiments showed that for $P \times Q$ subimages $\alpha = \min(P, Q)$ restricts the saturation of the exponential function, satisfactorily. Furthermore, E_1^i and E_2^i are scaled to have values from 0 to1.

In equation (6) parameter k can be used for determining to what extent the details of the two subimages have to match. The more details have to match, the higher the required value of k. At the limit, where $k = \min(r_1, r_2)$, the two images have to match exactly. In section 4 we compare the performance of this similarity measure with the normalized mutual information and the correlation coefficient.

3.5 Hypothesis Verification

A generated hypothesis is verified as follows.

1. Consider an initially extracted region, say A, in the original image.
2. Consider one of the discovered vectors connecting the centers of the regions that form prototypes.
3. Around the position pointed by the vector, which starts from the centre of region A, consider in the original image a region, say B, of the same size as region A.
4. If region B corresponds to one of the previously extracted regions, then ignore it and consider another extracted region and go to step 2. Otherwise go to the next step.
5. Compute the similarity measure between A and B.
6. If the similarity between A and B is more than a threshold, T, then consider B as a new region of interest.
7. Consider all extracted regions (including the new ones) and repeat steps 2 to 6 until no more hypothesized regions are verified.

Threshold T is selected as follows.

For each image we consider all pairs of initially extracted regions which are placed in relation to each other according to the discovered prototypes. Let us assume that there are P pairs of such regions. Next, we select as threshold of similarity the X percentile of the distribution of the similarity values. To identify this, we sort the similarity values in an ascending order. We then consider as threshold the value that corresponds to the region at position $\lfloor XP + 0.5 \rfloor$. For example, let us say that we have $P = 73$ pairs of regions. For $X = 0.25$, the threshold we shall use will be the similarity of the 18th pair, since $\lfloor 0.25 \times 73 + 0.5 = 18 \rfloor$. A region that is more similar to a given region by this threshold is postulated as a true region of interest.

After generating new regions, the overlapping regions are removed.

4. Experimental Results

For evaluating the performance of the proposed algorithm, 300 images of building facades were used. These images were annotated by the contribution of 10 persons and are freely available in [1].

Two different experiments are presented in this section. In the first part we evaluate the performance of different similarity measures in the hypothesis verification stage of the algorithm. Next we evaluate the performance of the algorithm in detecting subparts of building facades.

Hypothesis verification: Prior to comparing different similarity measures, we first consider the performance of the proposed similarity measure (6) as a function of the eigenimages retained. The ROC curves for different number of eigenimages turned out to be very similar. The curves were constructed by varying the percentile X used for deciding the similarity or not of the two compared sub-images. So, we select to use only one eigenimage.

Next, we construct the ROC curve of the proposed similarity measure as well as that of the correlation coefficient and the normalised mutual information, as shown in Fig 3. It can be seen that the proposed similarity measure has better performance in comparison with the correlation coefficient and the normalized mutual information for our specific task.

Evaluating the performance of the algorithm: In this part we evaluated the performance of the proposed methodology in detecting subparts of building facades. We compared the results of the proposed algorithm with hand segmented images of [1]. The segmented regions which we included in our experiments are windows, doors, chimneys, dormers, balconies and canopies. For testing whether an extracted region corresponds to a segmented image or not, we define:

$$O = \max_{S}\left\{ \frac{\#\{pixels \in (R \cap S\}}{\max\{\#(pixels \in R), \#(pixels \in S)\}} \right\} \quad (6)$$

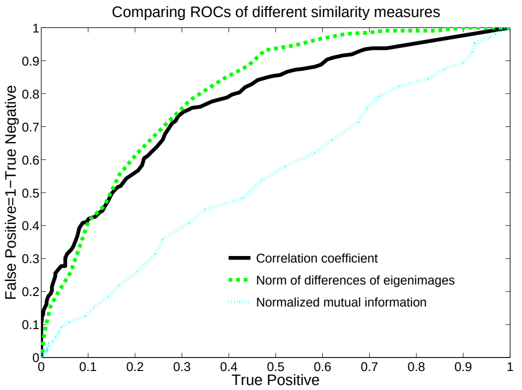


Fig 3: Comparing the ROC curves of different image similarity measures

where R is the set of pixels which belong to an extracted region of interest, and S is the set of pixels which make up a hand segmented region of interest. If O is more than 0.6 then we infer that the region of interest corresponds to one of the subparts of the building facade. The results of this experiment are listed in table 1 according to category, although our algorithm does not identify the class of each region. Some results are shown in Fig 4.

Table 1: Results of comparing the proposed methodology with the hand segmented images

Regions	*Number of regions*	*Blob extraction*
Window	3295	**82%**
Door	109	**56%**
Chimney	55	**35%**
Dormer	75	**74%**
Canopy	30	**33%**
Balcony	25	**75%**
All Regions	3589	**79.8%**

5. Conclusions

In this paper we proposed a fully bottom up approach for extracting blobs in images of building facades. Preliminary blobs are recognized using some image processing techniques.

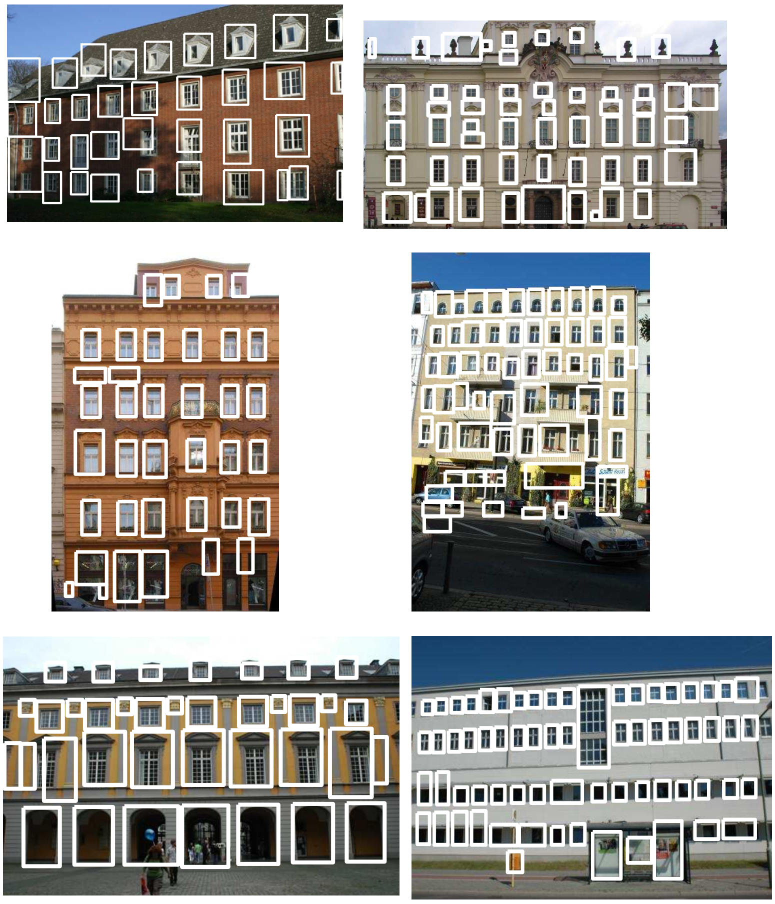


Fig.4. Bounding boxes of identified blobs are shown in white.

Prototype discovery, hypothesis generation and verification are used to identify any missed blobs or split the agglomerated ones when repeated patterns are present in an image. A novel similarity measure is proposed for assessing the similarity of subpatterns. We evaluated the performance of the proposed blob detection using 300 manually annotated images of building facades, containing more than 3500 subparts. Our experiments showed that about 80% of the hand segmented subparts were identified with the proposed methodology.

Acknowledgements: This work was supported by the EU project eTRIMS.

References

1. https://www.ipb.uni-bonn.de/svn/etrims-img-dbs/
2. Maria Petrou and Panagiota Bosdogianni: Image Processing: The Fundamentals, John Wiley, 1999, ISBN: 978-0-471-99883-9