Facade Proposal for Facade Detection and Matching

Antoine Fond[1], Marie-Odile Berger[2] and Gilles Simon[1]

***Abstract*— We introduce several "facadeness" measures of image regions and show how to combine them to generate building facade hypotheses in images of urban environments. We demonstrate the interest of this procedure through two applications. First, a CNN-based method is proposed to detect facades from a restricted list of facade hypotheses. We show that this method outperforms the state-of-the-art techniques in term of adequation of the detected facades with a ground truth. In addition, the computational time is compatible with the navigation requirements. Second, we investigate image matching based on facade proposal. Considering a large set of data extracted from Google Street View, we show that matching based on Euclidean distances between CNN descriptors outperforms the classical SIFT matching based on RANSAC-homography calculation.**

I. INTRODUCTION

Planar building facades are semantically meaningful city-scale landmarks. Such landmarks are essential for localization and guidance tasks in GPS-denied areas which are common in urban environments [1], [2], [3], [4]. Detection of facades is also key in augmented reality systems that allow the annotation of prominent features in the user's view [5].

Building detection from monocular images is a challenging task due to perspective deformations, repetitive structures and partial occlusions. Two categories of methods have been proposed in the past for the detection of building facades. Geometric-based methods attempt to identify rectangle facades in images rectified thanks to orthogonal vanishing points [1], [6]. Various geometric and photometric criteria are then used to characterize facades. However, they are generally too strict to take into account the variability of the facades encountered in urban areas.

On the other hand, with the advent of learning based techniques for classification, several methods have been designed with the aim to classify pixels or superpixels into categories. Among theses works, [7], [8] show examples of classification, "building" being one of the category. Though promising, these methods do not allow to identify regions which belong to the same facade.

Object detection has made great strides in recent years but these techniques have not been yet applied to building detection to our knowledge. Object detectors apply image classifier as a sliding window detector. In order to decrease the number of windows each classifier has to consider, a cascade of classifiers was proposed in [9]. Most of the time, a two step process is used [10], [11]: a fast classifier is first designed to extract a limited number of candidate windows, whereas a more complex classifier is used in the second step for scoring. The idea of defining an "objectness measure" for the pre-selection step, designed to produce a small number of regions such that top-ranked regions are likely to contain some categories, is developed in [12], [10], [13]. Though some methods integrate that objects have well defined closed boundaries [13], these methods are too general to be applied to the preselection of facades.

We thus propose in this paper a "facadeness" measure of image windows that can be evaluated rapidly and integrates geometric and photometric constraints commonly encountered in building facades.

We are thus able to generate a restricted list of facade hypotheses in a few seconds. We demonstrate the interest of our method through two applications:

- First, a CNN based method is proposed for facade building detection. The classifier is used on the restricted list of facade hypotheses. We show that this method outperforms the state of the art techniques in term of adequation of the detected facades with the ground truth. In addition, the computational time is compatible with the navigation requirements.
- Second, we investigate the interest of our method for model-based pose computation. The idea is that in case of cluttered environments or when the model is not prominent in the considered image, considering matching only in areas which are likely to contain facades will make the pose procedure more robust. Several matching criteria are considered in our study. Considering a large set of data extracted from Google Street View, we show that the rectification step noticeably improve matching. In addition, we show that matching based on the CNN descriptor outperforms the classical RANSAC-homography criterion.

The paper is organized as follows: related works are described in section II. Our method for facade proposal is explained in section III. Applications of our method to facade detection and matching are presented in sections IV and V. Finally, some experimental results are given in section VI.

II. STATE OF THE ART

Several methods have been proposed in the past to detect rectangular structures in Manhattan worlds. In [1], line segments are automatically detected and intersected to generate hypotheses of rectangles in agreement with the vanishing points. For each hypothesis, the input image is orthorectified and a histogram of gradient (HOG) is computed inside the warped rectangle. Hypotheses whose HOG contains more

[1]Antoine Fond and Gilles Simon are with the Université de Lorraine, Loria, Vandœuvre-lès-Nancy, 54506, France antoine.fond@loria.fr, gilles.simon@loria.fr

[2]Marie-Odile Berger is with the Inria Nancy Grand Est, Villers-lès-Nancy, 54600 France marie-odile.berger@inria.fr

than two dominant horizontal and vertical directions are discarded. This method is computationally expensive generating many superfluous hypotheses. To keep the problem tractable and efficient, Micusik et al. formulate the detection of the rectangles on a restricted neighborhood structure given by Delaunay triangulation [14]. The problem is then expressed as a search for the maximum aposteriori probability solution of a Markov random field. In [6], right-angle corners are detected in the orthorectified image using a Support Vector Machine. A Delaunay triangulation is performed from the right corners and a min-cut-like algorithm is used to generate windows in which a high density of right corners is observed.

All these methods allow to detect rectangular structures appearing on facades, like windows or rows of windows, but not, in general, entire facades. Finally, only a few attempts have been made to detect entire facades. Motivated by recent advances in low-rank matrix recovery via convex optimization, a new type of invariant image feature, called transform-invariant low-rank texture (TILT) has been proposed. From these features, patches at fixed grid coordinates belonging to the same facade can be merged based on the following criterion. Two adjacent patches I_1 and I_2 are merged into a larger patch $I = [I_1; I_2]$, if the joint texture remains low-rank [15]. This method produces good results in relatively simple cases where a few unoccluded facades occupy the whole image. However, the accuracy of the results highly depends on the sampling of the grid and the orientation of the facades relative to the camera. To tackle this issue, authors of [16] propose to use superpixels instead of rectangular patches. Moreover, the superpixels are grouped using a multiscale low-rank analysis and texture similarity measures based on a 2D Gabor wavelet. However, this method performs poorly on complex images, as shown in section VI-A.

In [17], the Gini Index is used to form an edge-based regularity metric relating regularity and distribution sparsity. The facade region detection is treated as a regional regularity/sparsity maximization problem, which is solved using greedy adaptive expansion over a down-sampled grid. Integrer Quadratic Programming is then used to select a subset of facades that have maximum regularity score and facade coverage, with minimum overlap. This procedure outperforms TILT with the aerial images shown in [17], and also in our own experiments (see section VI-A). However, the method still suffers from the use of a grid, and the regularity assumption makes it more suitable to large building facades with many regularly spaced windows than to the various kind of facades we consider in this work.

III. Facade Proposal

Our algorithm for facade proposal consists in a two-stage procedure. A first set of facade candidates relying on contours is initialized. Had hoc facade features (facadness cues) are then evaluated on that set, and the best facade candidates are selected by combining the obtained values in a machine-learning framework. A database of 920 images labeled as "buildings" or "street" in the ImageNet database was used for learning purpose. Each image is ortho-rectified and the bounding boxes of the facades are provided manually. These bounding boxes are referred as ground truth (GT) in the following.

A. Geometry of the scene and rectification

Manhattan hypothesis are well suited for modeling urban environments. All the buildings of the scene are considered to be 3D boxes and shall be parallel one to another. Each of the box faces is a facade except for top and down faces. Thus the geometric shape of a facade is a rectangle and its texture is defined by different visual characteristics that will be described below.

The method of [18] is used to find the Manhattan vanishing points. Knowing the intrinsic parameters of the camera, homographies that warp the image of the scene to ortho-rectified images are computed. Doing so, all the vertical facades of the scene appear in either of these ortho-rectified images as in a frontal view.

B. Rectangular candidate sampling

The main hypothesis we made on facades is that they are rectangular shaped. As we work in ortho-rectified images we are explicitly looking for rectangles. We choose to rely on the contours of the image to generate a first set of candidates. Indeed, the border of a facade should create high gradient values on the image. Edges are detected thanks to Dollar's algorithm [19]. They are then accumulated in both a histogram of vertical-projected edges H_x and a histogram of horizontal-projected edges H_y. The product $H_x H_y^T$ can be seen as a corners likelihood map (see Fig. 1). The n local maxima of that map are used to generate $\frac{n(n-1)}{2}$ rectangles. Actually, as both (top-left,bottom-right) and (top-right,bottom-left) pair of corners define the same rectangle, only a set of $\frac{n(n-1)}{4}$ facade candidates are retained. For instance, for the 920 images in our GT database, the average number of facade candidates per image is 34240 at this very first step.

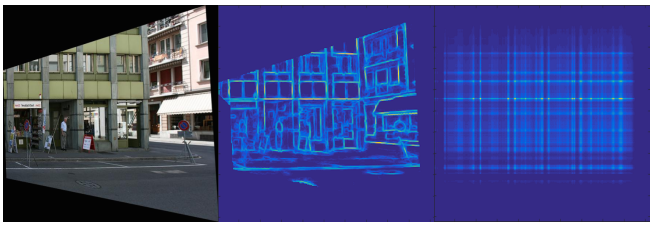

Fig. 1. Example image from our database (left), with the contour map (middle) and the corners likelihood (right).

C. Facadeness cues

Facades share several common visual characteristics. They are usually composed of rectangular features such as floors, windows, door, bricks. These features repeat themselves along the facade in both vertical and horizontal directions. Facades are also roughly symmetrical. Eventually facades are homogeneous in color at least compared to their background. For each of these facade candidates we evaluate 5 different

had hoc features (cues). We here reuse the color contrast cue defined in [10]. With respect to [10], [13] a stronger edge cue is defined which favors vertical and horizontal segments. Three new criteria are introduced aiming at characterizing shape, symmetry and repetitive patterns on facades. Subsequently the combination of all these cues enables us to discard the candidates that do not match our facade hypothesis and keep only the best ones. For each cue presented below, Fig. 2 shows the best rectangle obtained among all candidates, in an example image of our database (right column) and the probabilities of the cue values to be obtained on a facade (in green) or on a non facade (in red).

1) Shape cue: Facades are rectangular-shaped, but all rectangles are not as likely to be observed. Indeed, architectural rules allow just a few values of the facade aspect-ratio. Extremely thin facades are almost impossible for example. We have learned the probability distribution of two rectangular parameters (height and width) on our GT images (Fig. 2, top-right image). The shape cue is a coarse (24×24 bins) histogram H_s of that distribution :

$$s_{shape}(r) = H_s(24\frac{h}{h_I}, 24\frac{w}{w_I}) \quad (1)$$

where (h_I, w_I) are the height and width of the image I, respectively.

2) Contour cue: Facades are rectangular-shaped. We can expect a high gradient values along their border. Thus the contour cue is defined as :

$$s_{contour}(r) = \frac{1}{2(l+h)} \sum_{b(r,\alpha)} E_x|E_y \quad (2)$$

where α is the thickness of the band $b(r,\alpha)$ surrounding the rectangle r. E_x and E_y are the binary images of the horizontal and vertical Dollar's contour. The operator $|$ is the pixelwise logical or. (h,w) are the height and width of r, respectively.

3) Structure cue: Rectangular visual features repeat themselves along both vertical and horizontal directions on facades such as floor, windows, or bricks. The vertical edges projected on the horizontal axis are binned in an histogram H_x so are the horizontal edges projected on the vertical axis in H_y. The autocorrelation of both these two signals is sparse if there are strong repetitions of vertical edges and horizontal edges. We thus define the structure cue :

$$s_{struct}(r) = \frac{\sum_{peaks} \mathscr{F}^{-1}|\mathscr{F}(H_x^r)|^2}{\sum \mathscr{F}^{-1}|\mathscr{F}(H_x^r)|^2} + \frac{\sum_{peaks} \mathscr{F}^{-1}|\mathscr{F}(H_y^r)|^2}{\sum \mathscr{F}^{-1}|\mathscr{F}(H_y^r)|^2} \quad (3)$$

where H_x^r and H_y^r are the 64 bins-normalized-histograms defined above for the rectangle r. $\mathscr{F}$ and $\mathscr{F}^{-1}$ are respectively the Fourier transform and inverse Fourier transform. Peaks are local maxima of the signal.

4) Symmetry cue: Facades have a non-perfect axial symmetry. What we want is a metric that evolves continuously with the symmetrical aspect of the facade. For example the cross-correlation between the left and the right half of the image would be very high for even a small asymmetry. We propose to subdivise the rectangle into 16 patches. For each

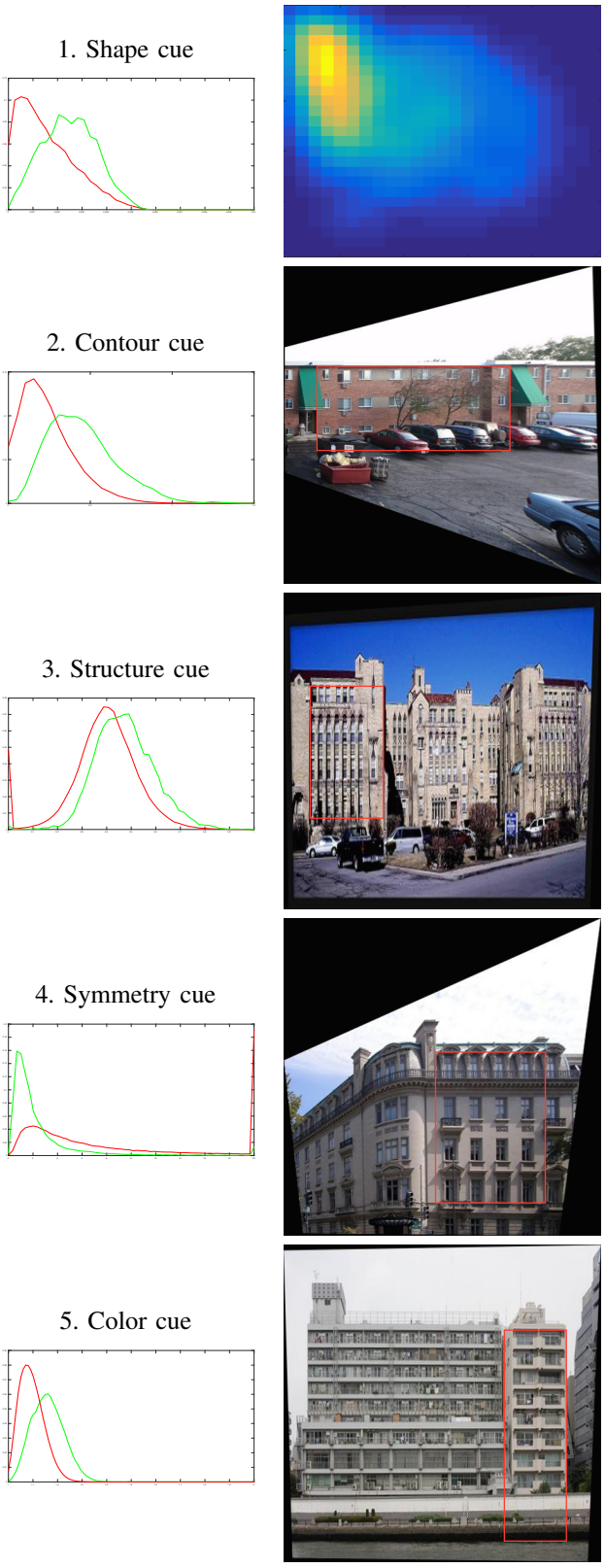

Fig. 2. Left column: probabilities for each cue values to be obtained on a facade (in green) or on a non facade (in red). Right column: best rectangle obtained among all candidates for each cue, in example images of our database (for the shape cue, the whole heatmap of the histogram H_s is shown).

of these patches we compute the HOG descriptor with 8 bins. Then we evaluate the distance between each of the 8 patches on the left with their symmetrical patch on the right :

$$s_{sym}(r) = \frac{1}{8}\sum_{i=1}^{4}\sum_{j=1}^{2} d_{\chi^2}\left(HOG^{sym}(s(i,j)), HOG(i,j)\right) \quad (4)$$

where $HOG(i,j)$ is the HOG descriptor with 8 bins of the patch (i,j). HOG^{sym} is the flipped version of vector HOG. s is the axial symmetry of vertical axis and d_{χ^2} the χ^2 distance.

5) Color cue: Color itself is a poorly informative feature to describe facades as facades can have many different colors. However the color homogeneity of a facade compared to its local context is a much interesting feature. The difference of color distribution between the inside of the rectangle and the surrounding region can distinguish facades as in Fig. 2:

$$s_{color}(r) = d_{\chi^2}\left(H_c^{b(r,\beta)}, H_c^r\right) \quad (5)$$

where $H_c^{b(r,\beta)}$ and H_c^r are respectively the color histogram of the inside of r and the color histogram of the band of thickness β surrounding r. We use LAB color space quantized into $256 = 4 \times 8 \times 8$ bins.

Computation of all the cues is in constant time for one rectangle thanks to the use of integral images. This trick is detailed in [20] for the computation of sums in regions as well as local histograms. The parameters α and β have been learnt on our learning database so as to maximize the separability between positive and negative examples (Fig. 2). The optimal values for these parameters are 5% and 30% of the dimensions of the rectangles, respectively.

D. Cues combination

Intersections between facade and non-facade probability distributions of cues values (Fig. 2, left column) mean that one cue alone cannot separate between facades and non-facades. To combine all these features into a single metric we use a multi-layer perceptron. It is composed of two hidden layers of respectively 8 and 6 neurons. This neural network has been trained on positive and negative examples, taken from the rectangle sets generated by the sampling procedure presented in section III-B, applied to all images of the GT database. To decide if a rectangle is a positive or a negative example, we used the commonly used metric "Intersection over Union" (IoU score s_{IoU}) [21]. An IoU threshold of 0.5 is often used in the literature to decide whether or not two image regions coincide. Morevover, an illustration in [21] shows that an IoU score of 0.5 already indicates a relatively high overlap. For these reasons, we took as positive examples rectangles that overlap the GT with $s_{IoU} \geq 0.5$, whereas negative examples are candidates with $s_{IoU} \leq 0.2$. This set of examples will be referred as our training set.

The final output of the perceptron can be seen as a probability score of being a facade. All candidates are sorted using this metric and the first N ranked candidates are finally kept. Figure 3 shows the recall rate obtained with the 920-images of our GT database as a function of N. A GT rectangle is counted retrieved when at least one of the selected candidates overlaps it with a $s_{IoU} \geq 0.5$. In practice, we use the 500 first candidates which corresponds to 84 % of recall. For such a number of candidates, the mean computation time on I7-3520M CPU is 3.2s.

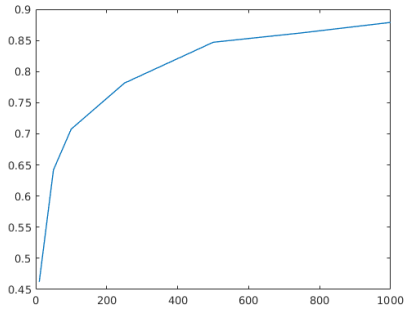

Fig. 3. Recall rate as a function of the number of candidates chosen.

IV. Application 1: Facade detection

The first application of our facade proposal method is facade detection. Inspired by recent works in object detection based on object proposal and CNN [22], our method intends to first propose a small number of facade candidates and then use a CNN classifier. Most object proposals methods rely on objectness assumptions that are poorly adapted to facades. As a consequence for these methods facades may be not detected or detected far in the list of proposals. That justifies the use of our facade proposal method as a first step for facade detection.

Another difference between facade detection and general object detection is that facade cannot be described only by the inside of their bounding box. Indeed, a part of a facade may be visually a facade too (the entire facade cropped so that one floor is missing for example). The only way to avoid this multiple parts problem is to consider the surrounding visual context. A true facade shall look like a facade inside but its context shall differ. We propose to build our facade descriptor by concatenating the CNN descriptor inside the rectangle with the CNN descriptor of the augmented rectangle by 25% in every directions. These two CNN descriptors are the last 4096-dimensional layer before the classification layers of the f-VGG network trained on ImageNet [23]. Their 8192 concatenated vector is the input of a neural network classifier. The classifier is composed of 2 more hidden linear layers of size 8192 and 4096 (figure 4). They have been trained on an augmented version of our training set. This new training set is divided into synthetic and real examples. The 20000 positive synthetic examples are plain facade images pasted in urbain context images. The negative synthetic examples are 10000 images of these urban context images (unrectified streets, cars, pedestrians,...) and 10000 of parts of the plain facade images. The real positive and negative examples are the same as for the learning of the cues combination metric. In total the augmented database gather 80000 examples.

To guarantee that the detected facades do not overlap each other we use the same greedy approach as in [10] regarding the output score of our network.

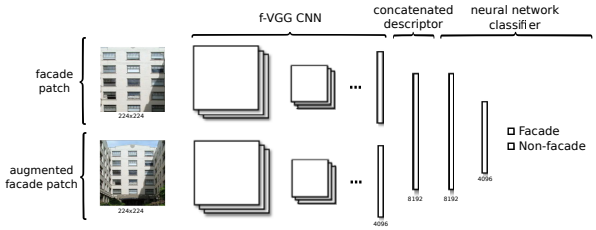

Fig. 4. Outline of our CNN architecture for facade detection.

V. Application 2: Facade Matching

Visual localization in urban environment is a second application for our facade proposal method. In such environment methods using local features such as SIFT can fail due to extreme perspective effects and numerous repetitions on buildings. We propose to rely on facades as visual landmarks. Facades are more complex visual objects than local features and so may be more reliable for matching with a known model of the scene.

We consider we know a model of the scene. This model is composed of the different facades of the urban scene. Each facade is stored by its high-resolution frontal view. The problem here is to find these facades in an unknown view of the scene.

We first generate facade candidates in this new image using our facade proposal method. Then for each of these candidates we compute a CNN descriptor. We use the same 4096-dimensional layer as in the previous application. We match a facade of the model with the closest facade candidate regarding the Euclidean distance of their CNN descriptors.

VI. Experimental Results

A. Facade detection

Our algorithm for facade detection (section IV) was assessed on a set of 118 Google Street View images of Nancy. As shown in Fig. 5, our method is able to handle various types of scenes and architectures (aligned facades, isolated buildings, houses) observed at different scales and orientations, and sometimes in very cluttered environments.

To quantitatively assess the relevance of the method, facades in the dataset were manually tagged by two people, and the intersection of their tags was taken as ground-truth. To compute the intersection of the tags as well as the results presented below, two regions were considered the same when their IoU score was greater than 0.5. Fig. 6 - Left (yellow curve) shows the ratio of GT facades obtaining an IoU score greater than the x-values (when a facade has been tagged twice, the highest of the two IoU scores is taken). Summary statistics are provided in Tab. I - Left (last column): mean IoU score of the GT tags, percentage of GT facades considered as found (i.e. $s_{IoU} > 0.5$), mean number of false positive (i.e. detected facades not in GT) and mean computation time on I7-3520M CPU. A mean IoU score of 0.45 is a pretty good result considering the discussion in section III-D. Moreover, inter-human variability in tagging facades is high in itself: Tab. I - Right shows the mean IoU of T_i compared to T_j, where T_1 and T_2 are the tag sets we used to build the GT, and T_3 is the tag set obtained by a third person.

These results can be compared to those obtained with the algorithms described in [17] and [16]. We used the MATLAB code of [16] available on the author's website `http://www.eecs.berkeley.edu/~ppnathan/research.html` and our own MATLAB implementation of [17]. Fig. 6 shows the statistic values, as well as typical failure results obtained with each of the compared methods. The method of Lam et al. shows high computation times (986 sec. per frame – spf) and low IoU scores (mean of 0.11). As shown in Fig.6 - [Lam], the texture similarity score, based on Gabor filters, often fails to discriminate between two adjacent facades. Moreover, superpixel grouping based on low-rank analysis can lead to merge facade regions with uniform parts of the background. In Fig.6 - [Lam], the pavement is not completely uniform, as it contains some linear structures, but these structures are parallel to the horizontal lines of the facades, which allows grouping while preserving the rank. The method of Liu et al. is much faster (0.39 spf) and gets a higher mean IoU (0.24), but this score is still poor compared with what is achieved with our method. As shown in Fig. 6 - [Liu], this method often leads to over-segmented regions. One difficulty in implementing their algorithm was to tune the sampling of the point grid used for greedy region expansion. We tried different sampling values and chose the one leading to the highest mean IoU.

Our algorithm performs in 32.3 spf, of which only 3 sec are used for the facade proposal procedure. The rest of the time is used to apply the detection network to the 500 retained candidates, which could easily be parallelized on GPU. Fig. 6 - Our shows an example of failure obtained with our method. The detected facades are separated at incorrect positions due to the presence of high vertical gradients near the true separation line. Moreover, the right part of the first (from left to right) facade is merged with the second facade, while its left part is undetected. The facadeness cues defined in section III do therefore not allow to distinguish between the first and the second facade. The non-detection is mainly due to the fact that our algorithm performs on rectified images, which are black in regions falling outside the image boundaries in the unrectified image (e.g. the top-left corner of the first facade). Black values are used as a mask in the proposal procedure, so that the facadeness cues are only computed outside the mask. However, they are necessarily included in the rectangles and augmented rectangles used to build the CNN descriptors, so that the detection capabilities can be degraded for rectangles that contain a large proportion of mask values.

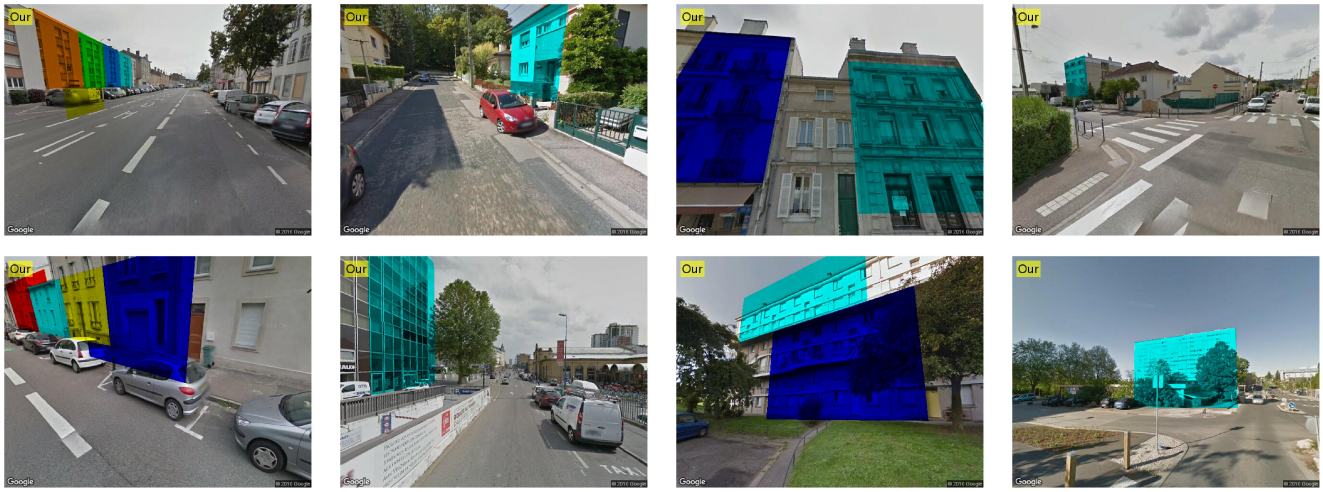

Fig. 5. Example results of our facade detection algorithm.

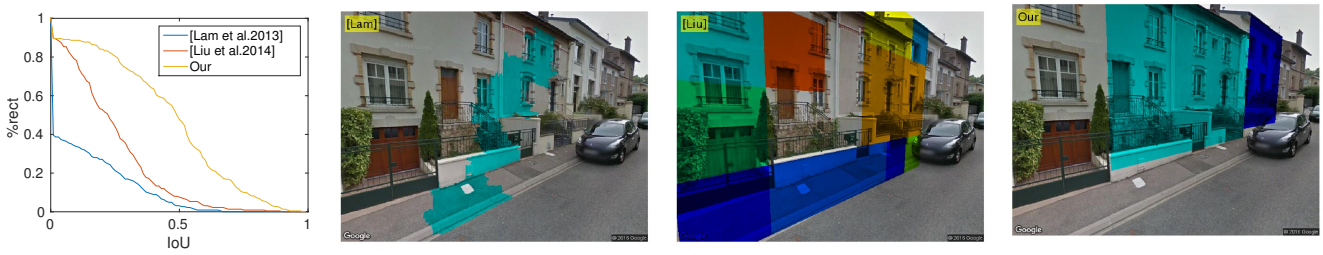

Fig. 6. Comparison between [16], [17] and our method.

TABLE I

STATISTICS OF FACADE DETECTION.

	[Lam]	[Liu]	Our
Mean IoU	0.11	0.24	0.45
Found (%)	3.1	7.6	47.7
False pos.	0.6	8.1	2.3
Time (spf)	986	0.39	32.3

	T_1	T_2	T_3
T_1	100.0	63.8	69.8
T_2	70.0	100.0	65.5
T_3	68.8	58.3	100.0

B. Window Matching

The test set of the second application is composed of 20 different scenes of Nancy from Google Street View. The buildings of each scene are observed from different viewpoints leading to an average of 20 images per scene and a total of 477 images. Each facade building has been manually tagged. When a facade appears in several images of the scene all of its instances refer to a single high-resolution frontal view of that facade refered as the model view.

For each of the 477 images, a matching is performed with all referred model views. We compare our method to different approaches based on SIFT. In the first approach SIFT features and descriptors are extracted in the whole image. The descriptors are then matched to the SIFT descriptors extracted from the model view, and a homography is estimated by RANSAC from these matches. In the second approach, the SIFT descriptors are extracted in the ortho-rectified images. Matching with the model view is done the same way but RANSAC only estimates a similarity.

Eventually in the last approach the SIFT descriptors extracted in the model are matched to the SIFT extracted inside a rectangle generated by our facade proposal method in the ortho-rectified images. For each of these rectangles we estimate a similarity with RANSAC algorithm. We only keep the best one regarding the inlier's rate. In order to assess the three matching procedures, we back-project the boundaries of the rectangles to the image through the inverse transformation (homography or similarity). When this area overlaps the ground-truth with a score $s_{IoU} \geq 0.5$, the matching is considered as correct.

TABLE II

STATISTICS OF FACADE MATCHING

	SIFT	SIFT+rect	SIFT+rect+bb	Our
Found (%)	18.43	29.54	31.76	36.66

Table II shows that our method outperforms a standard SIFT approach for matching facades in urban environments. Indeed in such context where perspective effects are usually strong the invariance of SIFT descriptor is really put to the test. Thus it is hard to find any correct matches based on SIFT descriptors. The statistic results show that orthorectification may overcome this point. Limiting the area of matching in the image also improves the matching rate by about 2%. Indeed, it prevents the SIFT features of the model view from being matched to SIFT features on similar but

separate facades. However, this is not sufficient as facades are basically uniform area with repeated windows, which generates a lot of similar descriptors. This may confuse the matching procedure. On the other hand, rejecting those descriptors whose ratio of the nearest neighbor distance to the second nearest neighbor distance is greater than a threshold results in a very low number of matches (Fig. 10).

The CNN descriptor overcomes this problem because it describes an entire image area and not only local features. Thus uniform texture and repetitive patterns are somehow part of the descriptor (see images 3 and 4 of Fig. 7). Eventually the descriptor is very stable to clutter as it is shown in images 1 and 2 of figure 7. However, the invariance to scale and translation of the CNN descriptor can cause some issues like in figure 9. In that case, the closest CNN descriptor to the model view regarding the Euclidean distance refers to a rectangle insufficiently overlapping the ground truth. However it is not the best rectangle that can be matched among the facade candidates. The green curve in figure 8 shows that in case of failure this situation is not so rare. If there is 36.6 % chance of finding a correct match between the model and the closest rectangle regarding the Euclidean distance with its CNN descriptor, there is 52 % chance of a correct match between the model view and one of the 10 closest rectangles. And the curve increases very fast with more than 60 % chance of correct matches among the 50 closest. By contrast, the red curve in Fig.8 (rate of matches found with the SIFT+rect+bb method as a function of the rank) is almost linear, with a low slope, which means that considering higher ranks would not make the chance of finding correct matches increase a lot.

Finally, an example of matching between a reference image taken in daylight and an image taken at night time is provided in Fig. 10. Results obtained with SIFT-based matching are very poor due to the repetitive patterns in the facade. On the contrary the global CNN descriptor succeeds in providing appropriate matching and has proven to be robust againts abrupt changes in the lighting conditions. This opens the way towards challenging applications where the conditions of the applications are very different from the ones encountered during model acquisition.

VII. CONCLUSIONS

We presented a fast facade proposal method that can be applied to both facade detection and facade matching. We demonstrated the relevance of using CNN descriptors for both these problems. Logical future developments would be places recognition from facade retrieval and pose computation. Though the invariance of CNN descriptors to small translations makes our facade-matching framework not currently suited to accurate pose computation, it proposes a good initialization for a further gradient-based image registration method.

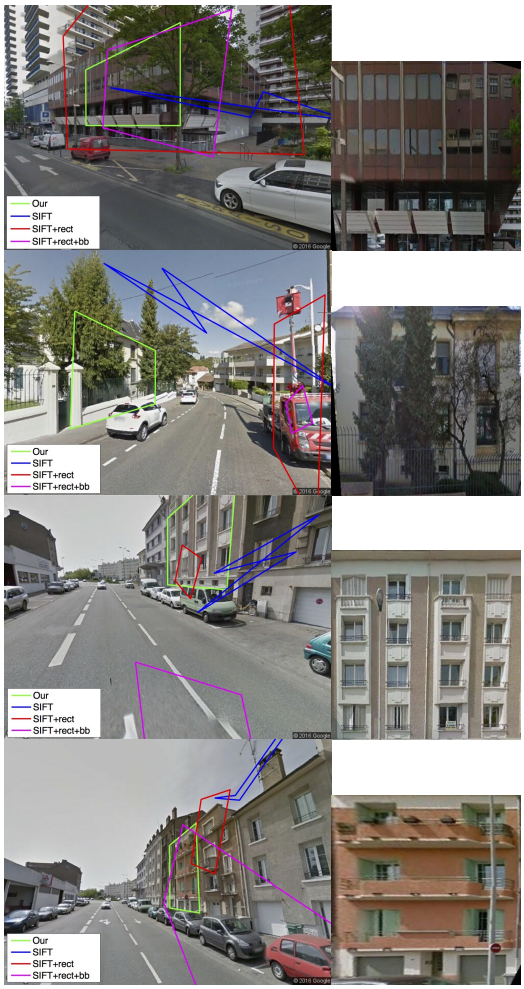

Fig. 7. Exemples of correct match for our method and incorrect match for the others. The overlaying outlines are the projected boundaries of the model in the image. The model to be matched is the image on the right.

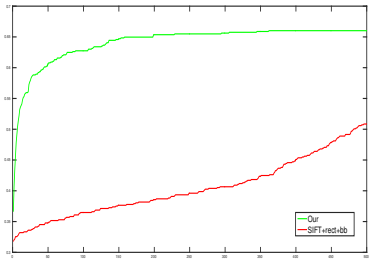

Fig. 8. Rate of matches found before the rank i over the 500 candidates.

REFERENCES

[1] J. Košecká and W. Zhang, "Extraction, matching, and pose recovery based on dominant rectangular structures," *Comput. Vis. Image Underst.*, vol. 100, no. 3, pp. 274–293, Dec. 2005.

[2] D. M. Chen, G. Baatz, K. Koser, S. S. Tsai, R. Vedantham, T. Pylvanainen, K. Roimela, X. Chen, J. Bach, M. Pollefeys, B. Girod, and R. Grzeszczuk, "City-scale landmark identification on mobile devices," in *Proceedings of the 2011 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2011*, 2011, pp. 737–744.

[3] H. Li, D. Song, Y. Lu, and J. Liu, "A two-view based multilayer feature graph for robot navigation," in *IEEE International Conference on Robotics and Automation, ICRA 2012, 14-18 May, 2012, St. Paul, Minnesota, USA*, 2012, pp. 3580–3587.

[4] C. Arth, C. Pirchheim, J. Ventura, D. Schmalstieg, and V. Lepetit, "Instant outdoor localization and slam initialization from 2.5d maps," 2015.

Fig. 9. Top : Exemple of incorrect match for our method and correct match for the others. Bottom : 12 closest facade candidates sorted regarding the Euclidian distance of their CNN descriptor with the model. The 12th is a correct match.

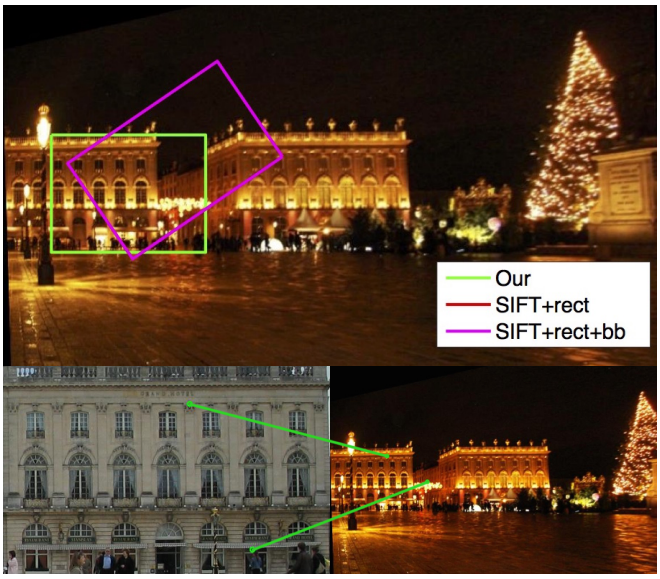

Fig. 10. A matching example where SIFT based method fails and CNN descriptor succeeds.

[5] F. Liu and S. Seipel, "Detection of facade regions in street view images from split-and-merge of perspective patches," *Journal of Image and Graphics*, vol. 2, no. 1, pp. 8–14, 2014.

[6] A. Fond, M.-O. Berger, and G. Simon, "Prior-based facade rectification for AR in urban environment," in *ISMAR workshop on Urban Augmented Reality*, Fukuoka, Japan, 2015.

[7] D. Hoiem, A. A. Efros, and M. Hebert, "Automatic photo pop-up," in *ACM SIGGRAPH 2005 Papers*, 2005, pp. 577–584.

[8] C. Farabet, C. Couprie, L. Najman, and Y. LeCun, "Learning hierarchical features for scene labeling," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 8, pp. 1915–1929, 2013.

[9] A. Vedaldi, V. Gulshan, M. Varma, and A. Zisserman, "Multiple kernels for object detection," in *Proceedings of the International Conference on Computer Vision (ICCV)*, 2009.

[10] B. Alexe, T. Deselaers, and V. Ferrari, "Measuring the objectness of image windows," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 11, pp. 2189–2202, Nov. 2012.

[11] H. Harzallah, F. Jurie, and C. Schmid, "Combining efficient object localization and image classification," in *ICCV 2009 - 12th International Conference on Computer Vision*, Kyoto, Japan, 2009, pp. 237–244.

[12] I. Endres and D. Hoiem, "Category independent object proposals," in *Proceedings of the 11th European Conference on Computer Vision: Part V, ECCV 2010*, Berlin, Heidelberg, 2010, pp. 575–588.

[13] M.-M. Cheng, Z. Zhang, W.-Y. Lin, and P. H. S. Torr, "BING: Binarized normed gradients for objectness estimation at 300fps," in *IEEE CVPR*, 2014.

[14] B. Micusík, H. Wildenauer, and J. Kosecka, "Detection and matching of rectilinear structures," in *2008 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2008), 24-26 June 2008, Anchorage, Alaska, USA*, 2008.

[15] H. Mobahi, Z. Zhou, A. Yang, and Y. Ma, "Holistic 3d reconstruction of urban structures from low-rank textures," in *IEEE International Conference on Computer Vision Workshops*, Nov. 2011.

[16] C.-P. Lam, A. Y. Yang, E. Elhamifar, and S. S. Sastry, "Multiscale tilt feature detection with application to geometric image segmentation," in *Proceedings of the 2013 IEEE International Conference on Computer Vision Workshops, ICCVW 2013*, Washington, DC, USA, 2013, pp. 570–577.

[17] J. Liu and Y. Liu, "Local regularity-driven city-scale facade detection from aerial images," in *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2014*, Washington, DC, USA, 2014, pp. 3778–3785.

[18] G. Simon, A. Fond, and M.-O. Berger, "A Simple and Effective Method to Detect Orthogonal Vanishing Points in Uncalibrated Images of Man-Made Environments," in *Eurographics 2016*, Lisbon, Portugal, May 2016.

[19] P. Dollár and C. L. Zitnick, "Fast edge detection using structured forests," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 8, pp. 1558–1570, 2015.

[20] P. Viola and M. Jones, "Robust real-time object detection," in *International Journal of Computer Vision*, 2001.

[21] C. L. Zitnick and P. Dollár, "Edge boxes: Locating object proposals from edges," in *Proceedings of the 13th European Conference on Computer Vision, ECCV 2014*, Zurich, Switzerland, Sept. 2014.

[22] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition*, ser. CVPR '14, 2014, pp. 580–587.

[23] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman, "Return of the devil in the details: Delving deep into convolutional nets," in *British Machine Vision Conference*, 2014.