Unsupervised Facade Segmentation Using Repetitive Patterns

Andreas Wendel, Michael Donoser, and Horst Bischof

Institute for Computer Graphics and Vision, Graz University of Technology
{wendel,donoser,bischof}@icg.tugraz.at

Abstract. We introduce a novel approach for separating and segmenting individual facades from streetside images. Our algorithm incorporates prior knowledge about arbitrarily shaped repetitive regions which are detected using intensity profile descriptors and a voting–based matcher. In the experiments we compare our approach to extended state–of–the–art matching approaches using more than 600 challenging streetside images, including different building styles and various occlusions. Our algorithm outperforms these approaches and allows to correctly separate 94% of the facades. Pixel–wise comparison to our ground–truth yields a segmentation accuracy of 85%. According to these results our work is an important contribution to fully automatic building reconstruction.

1 Introduction

Large–scale image acquisition for geospatial mapping platforms such as Microsoft Bing Maps or Google Maps requires appropriate data processing methods. While early systems just showed raw photographies, more recent visualization techniques allow to superimpose data obtained from satellite imagery, aerial photography, and road maps. Modern systems use multiple–view images to analyze the geometric properties of objects, resulting in 3D visualizations.

State–of–the–art methods [1] for 3D reconstruction are able to automatically extract simple building models from aerial images. However, photographs taken from an airplane offer limited view of building facades and therefore the reconstruction lacks details. This is depicted to the left in Figure 1 using a 3D model obtained from Microsoft Bing Maps. In contrast, an image of the same location taken from street–level can be found to the right. The advantages are obvious: *Streetside images* can be obtained with higher spatial resolution and from a more natural point of view than aerial images.

Recently, there has been increasing interest in using streetside imagery for automatically deriving 3D building models [2,3]. Müller et al. [2] introduced an approach on image–based procedural modeling. Given single facade images they first determine the structure of a facade. Once they know about the hierarchical subdivisions of the facade it is possible to replace architectural elements by parameterizable models. This representation has several advantages: First, the visual quality of the image is improved. Whether we scale or change the view within the model, there is no limit in spatial resolution. Second, the approach

M. Goesele et al. (Eds.): DAGM 2010, LNCS 6376, pp. 51–60, 2010.

Fig. 1. Comparison between aerial and streetside imaging: While the 3D model obtained from Microsoft Bing Maps (**left**) lacks details, it is easy to recognize the New York Stock Exchange in the streetside image (**right**)

assigns semantical meanings to parts of the facade. This is important for future applications using city models, for instance if the entrance has to be located. Third, the huge amount of image data required to visualize an entire city can be reduced to a predictable number of parameters. This allows geospatial mapping systems to transmit high–quality data in a reasonable time across the web.

The availability of procedural modeling algorithms motivates the implementation of fully automatic streetside modeling pipelines. However, the gap between real–world data and assumptions in algorithms is big: State–of–the–art approaches for 3D modeling [2,3] require orthorectified images of a *single* facade as input. Since streetside data is in general acquired by cameras on top of a moving car, images are not rectified and may show multiple facades. The goal of our work is to close this gap by detecting and extracting single facade segments from streetside images. By our definition, two facades should be separated if a significant change in color or building structure can be detected. A single facade segment is therefore a coherent area in an image, containing *repetitive patterns* which match in color and texture.

The successful realization of this task does not only have an impact on automatic procedural modeling workflows but also supports other computer vision algorithms that cope with urban environments. For example, window detection is strongly simplified if applied to single facades because the appearance of windows is often similar. Our work contributes to this goal in two areas:

(1) Repetitive patterns. We analyze repetitive patterns by using contextual information rather than directly comparing features or raw image data. Besides, we compare our algorithm against various state-of-the-art feature matching approaches which we adapt for our purpose.
(2) Facade separation and segmentation. Building upon repetitive patterns discovered in streetside images, we show how to separate and segment facades. The approach is evaluated using 620 high–resolution streetside photographs, acquired by cameras on top of a moving car. The images offer total coverage of a city as seen from roads, but also include difficulties such as various building styles and occlusions.

2 Facade Separation and Segmentation Algorithm

Our algorithm for facade segmentation consists of three major steps: First, we detect repetitive patterns in streetside images by extending a method designed for wide–baseline matching (Section 2.1). The resulting pairs of interest points are then used in a bottom–up manner to separate facades (Section 2.2). Finally, we combine the knowledge about repetitive areas with state–of–the–art segmentation methods to obtain individual facade segments (Section 2.3).

2.1 Finding Repetitive Patterns

Matching of local features has been widely investigated but hardly ever applied to a *single* image. Most algorithms were originally developed for object recognition or wide–baseline matching where the task is to find the single best match to a descriptor in a second image. We choose the Scale–Invariant Feature Transform (SIFT) [4] to represent the category of local feature–based matching approaches. SIFT has been designed for finding correspondences among feature sets from different images. For single image operation a range of valid matches needs to be defined. However, finding a proper threshold turned out to be difficult as the descriptor either matched with structures across facade boundaries or it did not find enough matches within a facade.

Shechtman and Irani [5] presented an approach to match complex visual data using local self–similarities. They correlate a patch centered at the point of interest with a larger surrounding region and use the maximal correlation values within log–polar bins as descriptor. The benefit of this approach is the independence of representation, meaning that just the spatial layout or shape is important. However, this poses a problem for our needs: The most common repetitive patterns are windows and their shape is often similar in different facades. While the texture within the window often stays the same, the texture outside changes and should definitely influence the result.

Tell and Carlsson [6,7] developed a robust approach for wide–baseline matching. The basic idea is to extract intensity profiles between pairs of interest points and match them to each other. If these profiles lie on a locally planar surface such as a facade, any scale–invariant descriptor is also invariant to affine transformations. Fourier coefficient descriptors of the first image are then matched to descriptors of the second image, and votes are casted for the respective start- and endpoints of the matching intensity profiles. Maxima in the voting table are then considered as the best matches for the interest points of both images.

Within this work, we adapt the approach of [6] for finding repetitive patterns within a single image. We detect interest points in the image, namely Harris corners [8], and extract color intensity profiles on a straight line between them. This results in a graph which connects all interest points (nodes) to each other. To limit the complexity, we take only the nearest 30 neighbors into account.

Every RGB color channel contributes 20 values to the descriptor, sampled using bilinear interpolation in regular intervals along the line. Finally, the 60–dimensional descriptor is normalized. We achieve scale invariance by extracting

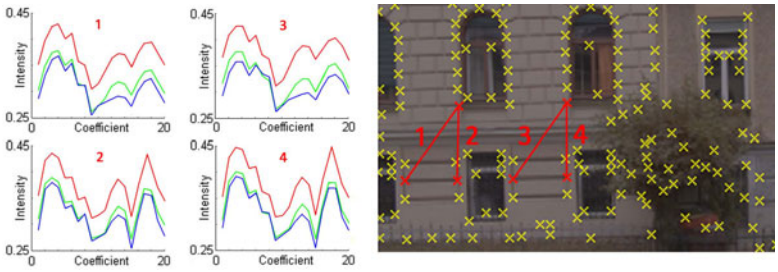


Fig. 2. We describe the image content between Harris corners by extracting intensity profiles with 20 values for every RGB color channel

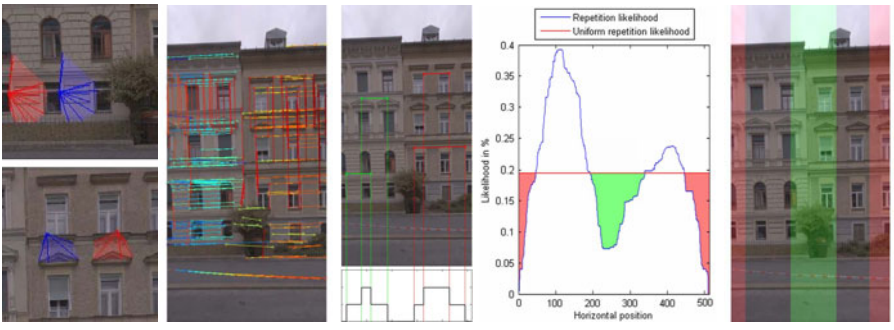
a fixed amount of coefficients from interpolated, one–dimensional data. While illumination changes of small areas are also corrected by the interpolation, larger areas can be handled with the tolerance of the matching approach. Figure 2 visualizes some extracted intensity profiles and their origin in the image.

We use a kd–tree [9] for efficient matching of descriptors within a single image, tolerating ±5% deviation off the descriptor prototype for finding repetitive patterns. We do not consider matches with more than 10 descriptors involved because these features are not discriminative enough. The robustness of the approach is based on an additional voting step. All matching profiles vote for the similarity of the respective pair of start- and endpoints. Using this method, interest points which are in similar regions get more votes than two random points. We store a list of contributing profiles for every possible pair and increase the vote count only if a descriptor has not contributed to that correspondence so far. This is in a similar manner described in [7] and ensures that no bias is introduced in the voting matrix. For locating repetitive patterns in the voting matrix, we threshold the number of votes a correspondence received. A correspondence of interest points has to be supported by at least 3 of 30 intensity profiles (10%). One of the advantages of this voting process is the ability to match arbitrary areas of the image, as visualized in Figure 3(a).

For the purpose of our work, we can further restrain the previously found matches. Repetitive patterns on facades are unlikely to occur across the entire image, but also very close matches are not valuable. We therefore restrict the horizontal or vertical distance of the matches to avoid outliers. The final result of our algorithm for finding repetitive patterns in a single image can be seen in Figure 3(b). Note that only a small amount of interest point correspondences cross the boundary between the two facade segments, while we can find a large number within a segment.

2.2 Facade Separation

Facade separation is a task which has not received much attention in the past. Müller et al. [2] introduced an algorithm which is able to summarize redundant parts of a facade and thus subdivide images into floors and tiles. However, a



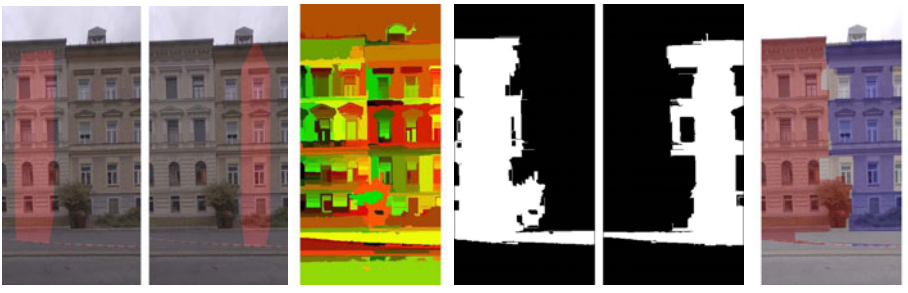
(a) (b) (c) (d) (e)

Fig. 3. From streetside data to separation (best viewed in color): **(a)** Matching of arbitrary areas **(b)** Detected repetitive patterns (color–coded lines) **(c)** Projection results in a match count along the horizontal axis **(d)** Thresholding the repetition likelihood with the uniform repetition likelihood **(e)** Resulting repetitive areas, separation areas (green), and unknown areas (red)

major limitation is the dependency on *single* facade images, and automatic processing fails for scenarios with blurry texture, low contrast, chaotic ground floors, and occlusions caused by vegetation. Other works on facade separation [3,10] are based on the evaluation of directional gradients, which did not proof to be robust for our datasets because it only works for highly regular facades.

The first stage of our facade segmentation algorithm provides interest point correspondences. We can now use these results to detect clusters of repetitive patterns. In this step we employ the Gravity Assumption, meaning that the majority of images in streetside datasets show facades where repetitions occur in horizontal direction and separations between facades in vertical direction. This allows us to project the lines between all pairs of matching interest points onto the horizontal axis and obtain the match count for every position on the horizontal axis. An illustration can be found in Figure 3(c). We normalize the match count to obtain the percentage of all matches at a given place on the horizontal axis, which we call the repetition likelihood.

Simply detecting minima on the repetition likelihood is not suitable for finding separation areas, as the global minimum would fail for panoramic images with multiple splits and local minima occur regularly between rows of windows. If all parts of the facade would contribute the same amount of repetitive patterns to the likelihood, we would get a uniform repetition likelihood. This value is an intuitive threshold, because areas where the likelihood is higher are more repetitive than average (*repetitive areas*) and areas where it is lower are less repetitive (*separation areas*). This fact is visualized in Figure 3(d). While repetitive areas are used for segmentation later on, separation areas mark the position where one facade ends and another starts. We also have to cope with the problem of narrow fields of view, i.e. few or no repetitions are visible in images that actually show a separation. We solve this by defining an *unknown area* on both sides of



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

Fig. 4. From separation to segmentation (best viewed in color): **(a)** Convex hull of repetitive points as segmentation prior **(b)** Superpixel segmentation [11] **(c)** Combination of prior and appearance **(d)** Typical result

the image, starting at the image boundary and ending at the point where the first repetitive area is detected. Separation examples are given in Figure 3(e). The algorithm does not depend heavily on the quality of repetitive patterns but works well with several approaches, as the evaluation in Section 3.1 shows.

2.3 Facade Segmentation

For the task of facade segmentation it is important to consider both the continuity of segments and the underlying structure of facades. Popular unsupervised methods such as Normalized Cuts [12] or efficient graph–based segmentation [11] failed to provide satisfactory results on our data because they lack this prior knowledge. An approach which considers prior knowledge has been presented by Korah and Rasmussen [13], who assume that the pixels located around a window belong to the building wall. Their method also fails for our datasets as we do not have knowledge about the window grid and a significant part of facades does not have homogeneous texture.

In contrast, we want to incorporate repetitive areas as prior knowledge. Based upon the results of the separation stage, we process repetitive areas individually. An interest point is called repetitive point if it has some correspondence to another point. We assemble a set of all repetitive points within a repetitive area and compute a convex hull [14] for this set of points, as can be seen in Figure 4(a). The graph–based segmentation approach of [11] can be adjusted to deliver superpixels (oversegmentations), visualized in pseudo–colors in Figure 4(b). We further combine prior knowledge and superpixels to the binary masks in Figure 4(c). For all repetitive areas, we include those superpixels into the respective facade segment which overlap with the convex mask. After morphological post–processing, the final output of our algorithm are individually segmented facades as depicted in Figure 4(d).

3 Experiments and Results

Experimental evaluation is based on 620 images in total, which can be grouped in two single–frame datasets and one dataset with panoramic images. First, the US–style dataset contains 220 single–frame facade images. The main difficulties of the dataset are very similar facades and low image quality in the top third of the image. The second dataset consists of 380 frames with European–style buildings. Compared to the first dataset, lower buildings result in more visible parts of the facades. However, cars and trees occlude the facade and large parts of road and sky can be seen. Furthermore, facades are heavily structured and it is hard to obtain good segments for most of them. Finally, we manually selected 20 series (each of about 30 consecutive frames) to create the Panorama dataset. Panoramic stitching is important because it increases the field of view in the direction necessary for finding repetitive patterns. However, common algorithms such as Autostitch [15] regularly fail due to occlusions by vegetation and significant depth changes. While we cannot rely on fully automatic image stitching for large sequences, we still want to show the applicability of our algorithm.

We use precision and recall [16] to evaluate our algorithm and combine them by a harmonic mean to obtain the measure of effectiveness, also called F_1–measure. We obtained ground–truth by manually labeling individual facades. We estimate the point matching quality by clustering the matches and assigning them to ground–truth, resulting in a set of inliers and outliers for every segment. We only use the match precision (PR_{match}), as it is not possible to estimate the amount of false negatives required to compute the recall. Facade separation quality $F_{1,sep}$ is estimated by checking if the detected repetitive area lies within the ground–truth segment. More or less splits lower the effectiveness except when they occur in an unknown area. The facade segmentation quality $F_{1,seg}$ is estimated using a pixel–wise comparison between the automatically obtained segment and the ground–truth segment.

Our approach depends on the parameters of the Harris corner detector ($\sigma_D = 0.7$, $\sigma = 3.0$ for the European–style dataset and $\sigma = 2.0$ else) and the settings of the superpixel segmentation ($\sigma = 1.0, k = 100, min = 100$). All other parameters are defined with respect to the image scale. Facade segmentation using intensity profiles takes about 10s per frame in a MATLAB implementation and could be further improved by a proper implementation.

3.1 Comparison of Methods

The first experiment analyzes the influence of different profile descriptors. We compare our choice of using intensity profiles in RGB color space, as described in Section 2.1, to intensity profiles in different color spaces (Lab, grayscale), Fourier profiles as used by Tell and Carlsson [6], gradients as used in SIFT [4], and RGB histograms which neglect all spatial information.

The second experiment analyzes the influence of different methods for point matching on the quality of facade separation and segmentation. We compare our profile matching approach to the extended versions of SIFT descriptors [4],

self–similarities [5], and raw patches of pixels. All methods are integrated in the workflow of the facade segmentation algorithm and differ only by the approach to repetitive pattern detection.

Table 1. Evaluation results as an average of all datasets. Our approach of intensity profiles in RGB color space performs best

620 images, 3 datasets	PR_{match}	$F_{1,sep}$	$F_{1,seg}$
Intensity profiles (RGB)	**72.8%**	**94.0%**	**85.4%**
Intensity profiles (Lab)	68.8%	89.6%	83.8%
Intensity profiles (gray)	67.2%	90.9%	83.6%
Fourier profiles (gray)	67.1%	89.8%	84.6%
Gradient profiles (gray)	51.0%	82.6%	80.3%
Histogram profiles (RGB)	51.8%	83.6%	81.3%
SIFT [4]	72.6%	86.6%	78.6%
Self–similarities [5]	63.9%	78.4%	70.2%
Raw patches of pixels	72.3%	92.4%	83.6%

Table 1 presents an overview of the average results on all datasets. According to these results our method performs better than any other intensity profile descriptor or any extended state–of–the–art method. Although the distance to its competitors is small there are two main reasons for using our method: First, it achieves the best matching precision combined with a number of correct repetitive patterns which is three times higher than for other methods. This increased support makes the repetition likelihood defined in Section 2.2 more robust. Second, our method is most tolerant regarding the appearance of repetitive patterns. Corresponding areas can be arbitrarily shaped and do not depend on scale or rotation.

3.2 Illustration of Results

After comparing the performance using objective measures we want to illustrate the results. Figure 5 shows the separation and segmentation of four consecutive images for the US- and European–style datasets. For better visualization, videos are provided online[1]. A typical result for multiple–facade separation and segmentation (Panorama dataset) can be found in Figure 6.

Separation problems occur if the field of view is too small. In such a case, a different column of windows is enough to indicate a different facade. Segmentation problems are mainly caused by repetitive patterns which are not part of the facade, such as power lines in the sky. The resulting segment is therefore too large and includes parts of the sky. Missing repetitive patterns at image boundaries lead to wrong segmentations as well.

[1] http://www.icg.tugraz.at/Members/wendel/

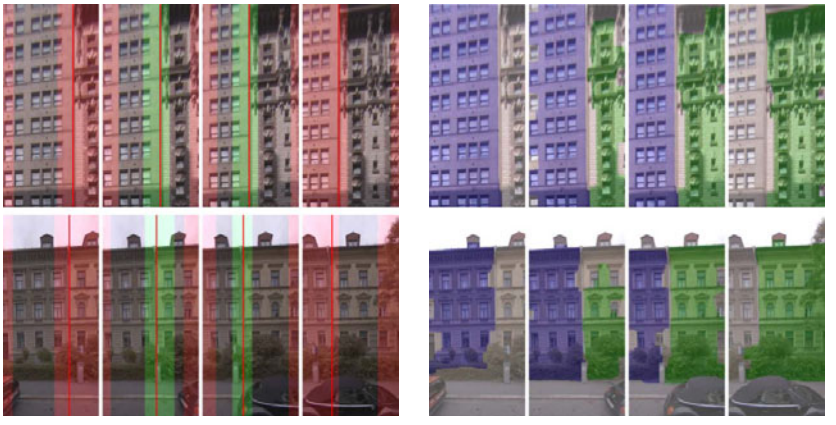


Fig. 5. Typical results for facade separation (**left**) and segmentation (**right**) for US–style facades (**top**) and European–style facades (**bottom**). The separation ground–truth is marked by red lines. It overlaps either with separation areas (green), or unknown areas (red) if not enough repetitive matches can be found.

Fig. 6. Typical results for multiple–facade separation (**top**) and segmentation (**bottom**) in the Panorama dataset

4 Conclusion

In this work we proposed an algorithm which closes the gap between real–world data and state–of–the–art procedural modeling approaches [2]. Our contributions are two–fold: First, we developed a novel algorithm for finding repetitive patterns in a single image. We compare contextual information using pairwise intensity profile descriptors and an intermediate step of vote casting. As a result, corresponding areas can be arbitrarily shaped and the matches are invariant to small illumination changes and affine transformations. Second, we presented a novel approach for

facade separation and segmentation. Our algorithm achieves excellent results of 94.0% separation and 85.4% segmentation effectiveness, making our work an important contribution to fully automatic building reconstruction. Future research should focus on detecting occlusions such as vegetation and cars, as avoiding separations in these areas would improve the performance. Furthermore, our approach will be applied to texture segmentation and symmetry detection.

Acknowledgments. This work has been supported by the Austrian Research Promotion Agency (FFG) project FIT-IT CityFit (815971/14472-GLE/ROD) and project FIT-IT Pegasus (825841/10397).

References

1. Zebedin, L., Klaus, A., Gruber-Geymayer, B., Karner, K.: Towards 3d map generation from digital aerial images. Journal of Photogrammetry and Remote Sensing 60(6), 413–427 (2006)
2. Mueller, P., Zeng, G., Wonka, P., Gool, L.V.: Image-based procedural modeling of facades. ACM Transactions on Graphics 26(3) (2007)
3. Xiao, J., Fang, T., Zhao, P., Lhuillier, M., Quan, L.: Image-based street-side city modeling. ACM Transactions on Graphics 28(5) (2009)
4. Lowe, D.G.: Distinctive image features from Scale-Invariant keypoints. International Journal of Computer Vision (IJCV) 60(2), 91–110 (2004)
5. Shechtman, E., Irani, M.: Matching local Self-Similarities across images and videos. In: Proceedings of CVPR (2007)
6. Tell, D., Carlsson, S.: Wide baseline point matching using affine invariants computed from intensity profiles. In: Vernon, D. (ed.) ECCV 2000. LNCS, vol. 1842, pp. 814–828. Springer, Heidelberg (2000)
7. Tell, D., Carlsson, S.: Combining appearance and topology for wide baseline matching. In: Heyden, A., Sparr, G., Nielsen, M., Johansen, P. (eds.) ECCV 2002. LNCS, vol. 2350, pp. 68–81. Springer, Heidelberg (2002)
8. Harris, C., Stephens, M.: A combined corner and edge detector. In: Proceedings of the Alvey Vision Conference, vol. 15, p. 50 (1988)
9. Friedman, J.H., Bentley, J.L., Finkel, R.A.: An algorithm for finding best matches in logarithmic expected time. ACM Transactions on Mathematical Software 3(3), 209–226 (1977)
10. Hernandez, J., Marcotegui, B.: Morphological segmentation of building facade images. In: Proceedings of ICIP, p. 4030 (2009)
11. Felzenszwalb, P.F., Huttenlocher, D.P.: Efficient Graph-Based image segmentation. International Journal of Computer Vision (IJCV) 59(2) (2004)
12. Shi, J., Malik, J.: Normalized cuts and image segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI) 22(8), 888–905 (2000)
13. Korah, T., Rasmussen, C.: Analysis of building textures for reconstructing partially occluded facades. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part I. LNCS, vol. 5302, pp. 359–372. Springer, Heidelberg (2008)
14. Barber, C.B., Dobkin, D.P., Huhdanpaa, H.: The quickhull algorithm for convex hulls. ACM Transactions on Mathematical Software 22(4), 469–483 (1996)
15. Brown, M., Lowe, D.G.: Automatic panoramic image stitching using invariant features. International Journal of Computer Vision (IJCV) 74(1), 59–73 (2007)
16. Rijsbergen, C.J.V.: Information retrieval. Butterworth-Heinemann Newton, MA (1979)