MULTI-IMAGE FUSION FOR OCCLUSION-FREE FAÇADE TEXTURING

Jan Böhm

Institut für Photogrammetrie, Universität Stuttgart, Germany
Jan.Boehm@ifp.uni-stuttgart.de

Commission V, WG V/2

KEY WORDS: Texture, Fusion, Rectification, Terrestrial Imagery

ABSTRACT

Façade texturing is a key point for realistic rendering of virtual city models, since it suggests to the viewer a level of detail that is much higher than actually provided by the geometry. Façade textures are usually derived from terrestrial imagery acquired from a position on the ground. One problem frequently encountered is the disturbance of texture images by partial occlusion of the façade with other objects. These occluding objects can be moving objects such as pedestrians and cars or static objects such as trees and street signs. This paper presents a method for the detection and removal of these disturbances and allows for the generation of occlusion-free texture images by multi-image fusion.

1 INTRODUCTION

Today many approaches for the creation of three-dimensional virtual city models are reported in literature. As widespread as the approaches for their generation are the applications of such models (Haala et al., 2002). When it comes to visualizing virtual city models, a key point in most applications, façade texturing is essential for realistic rendering. Due to the modern media and entertainment industry and their use of highly sophisticated computer equipment and specialist in the field, today's audience has high expectations to the quality of computer-generated visualizations. This raises the demand for high-quality texturing in virtual city models. One problem, and a major cause for the lack in the quality of façade textures, is the disturbance of images by partial occlusion of the façade with other objects, such as pedestrians, cars, trees, street signs and so on. Especially in inner-city areas it is impossible to avoid these occlusions, as the choices for the camera stations are limited. Therefore strategies for the detection and removal of these disturbances are essential.

The creation of façade textures is usually a labor-intensive manual task involving the acquisition of terrestrial images of the façade, the rectification of the images, the mapping onto the geometry of the model and various improvements to the imagery. Automated approaches, which have been reported in literature, solve the tasks by projecting primitives (triangles or texels) form the images to the object geometry (El-Hakim et al., 1998). This requires absolute orientation of the images, derived from bundle adjustment. If more than one mapping is available for a primitive, redundancy can be used to remove occlusions. Our approach differs in that it does not explicitly map image points to 3D geometry, but that we rather attempt to warp the images in 2D using perspective transformation. Thereby pixel-level registered image sequences are generated providing the redundancy to eliminate occluding objects by means of background estimation.

The occluding objects can roughly be categorized into two classes: moving objects and static objects. To detect and remove moving objects from terrestrial images of a façade, it is sufficient to acquire a sequence of images from a single camera station. This image sequence can be processed with algorithms from the field of video-sequence analysis known as background estimators, normally used for change detection. However these algorithms usually require long sequences (>100 images) to converge to a satisfactory result. To acquire such long sequences with high-resolution digital cameras would significantly increase the time and effort to acquire texture images. Therefore we explore alternative approaches suitable to short sequences (<10 images).

For the case of static objects, a single camera station is insufficient. Images from several different stations have to be acquired and fused. We solve the fusion of these images without entering the three-dimensional domain, avoiding over-proportional computational costs and complexity. Instead we attempt to solve solely in the two-dimensional image domain by mapping the image to the planar surface of the façade. This step of mapping, also referred to as rectification, is always involved when using terrestrial imagery for façade texturing. We have studied both manual and automated approaches for rectification in the past. When an occluding object lies in front of the façade plane at a certain distance, its mapping to the plane varies across the image sequence, due to the oblique angles of the different camera stations. Thereby the case of a static occluding object is transformed to the case of a moving object in the rectified sequence and hence the same approaches for detection and removal as in the case of moving objects can be applied.

Section 2 introduces to the task of façade texturing and explores current approaches to the task. Section 3 and 4 detail our approach for image fusion and occlusion removal based on clustering. Examples of real façade imagery with occlusions and the results of our processing are presented.

2 FAÇADE TEXTURING

A large number of systems exist for the 3D modeling of urban environments. Based on either aerial imagery or laser scan data they usually create polyhedral descriptions of individual buildings (see figure 2). Texturing these polyhedral structures greatly enhances their visual realism and suggest a much higher level of detail to the viewer as can be seen in figure 1. When aerial imagery is used, roof structures can be textured automatically. For the texturing of the vertical facades terrestrial imagery is needed.

Façade texturing has received some attention in both the photogrammetric and the computer vision community. Approaches for manual, semi-automatic and fully automated texture extraction have been presented in the past. Some of the systems combine the reconstruction process with the texture extraction.

The simplest approach is to use a single image and warp this image onto the planar façade using the perspective transformation T_P. Four points in both the planar coordinate system of the façade and the corresponding points in the image have to be determined. Assuming that all façades are bound by a rectangular curve the approach can be further simplified by splitting the perspective transformation into an intermediate perspective and into

a scaling part $T_P = T_S \cdot T_{P'}$. Since the target shape is a rectangle, the image can be transformed onto a unit square. This will lead to an orthographic projection, however the width and height of the image are incorrect. The correct scaling will be applied to the cropped image at rendering stage, when the image is mapped onto the corresponding façade. With this simplification only the four corner points of the façade have to be identified in the image and no control information is required. The advantage of such an approach is that it does not require the determination of the exterior orientation and neither does it require the calibration of the camera. Furthermore no control points in object space have to be determined. The disadvantage of using only a single image is that any object occluding the façade will be mapped as well and thereby disturbing the texture image. To avoid these occlusions manual stitching of images is required. Despite these disadvantages the approach has shown to be quite successful and several hundred buildings have been textured using several thousand terrestrial images (Kada et al., 2003).

Coorg and Teller (1999) have introduced an automated approach for reconstructing vertical façades from a set of images using a space sweep approach. The method relies on controlled imagery, i.e. in photogrammetric terms the exterior orientation of each image is assumed to be known. Textures are computed from a set of images by a weighted median estimation process.

Wang et al. (2002) have contributed to this work with a further improved method of computing texture. They account for three sources of influence on the façade image: the occlusion by modeled objects, the obliqueness of a certain portion of the façade with respect to the camera station and the occlusion by unmodeled objects. The texture image is computed in an iterative manner using a weighted-average approach. In addition to the texture they present a method to compute the 3D façade relief as well.

Bornik et al. (2001) presented a photogrammetric approach for deriving texture maps. The geometry of buildings is reconstructed by means of photogrammetry using multiple views. Camera calibration data and exterior orientation from the photogrammetric step are then used to compute texture images. The texture image is synthesized from several views again using a median filter.

For our work we aim at keeping the simplicity of an approach purely based on perspective transformation. Yet we want to eliminate the need for manual stitching to suppress occlusions and thus enhance the approach to use multiple views. The main idea as mentioned above is to create pixel-level registered image sequences from these views. Regarding multiple images as a sequence allows us to apply background estimation techniques to eliminate occlusions.

The next section gives a quick overview of background estimation and introduces simple examples where several images from a single photo station are used, allowing for the elimination of moving objects.

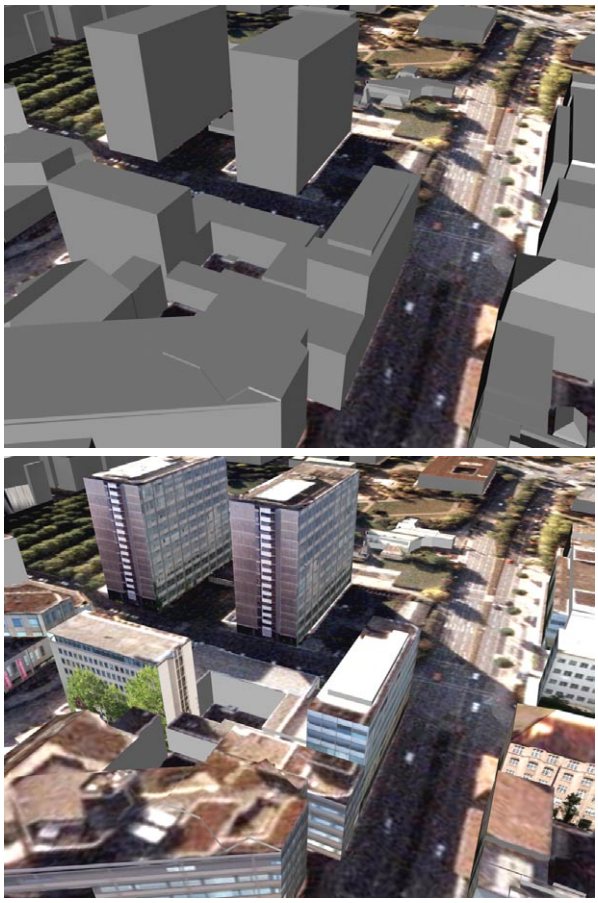


Figure 1: Façade textures can compensate for lack of geometric detail. The improvement in visual appearance can be seen from the above two images, where the second image is based on the exact same geometry and only texture was added.

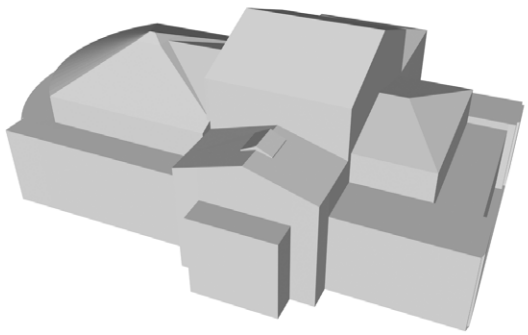


Figure 2: The model of a single building. All façades are approximated by planar surfaces and bounded by rectangular curves.

3 BACKGROUND ESTIMATION

The process of background estimation or background maintenance is frequently encountered in video surveillance systems, where a fixed camera is directed into a hallway, onto a building or to a road crossing, and so on. The sequence of images obtained from the camera is compared to an existing background image. Using simple background subtraction changes in the scene can be observed. These changes can be caused by people, cars or other objects, which are to be identified, tracked or otherwise detected.

The difficulty in such a system lies in obtaining the correct background image. This is especially true when no image of the background free of occluding object is available. A second reason for

the difficulties of the task are the changes that can appear in the background image, such as changes in illumination, shadows and other changes in the scenery. The topic has received great attention in the past as well as in the present. Toyama et al. (1999) list ten different algorithms to solve most of the common problems of background estimation. Among those are simple adjacent frame differencing, pixel-wise mean values and fixed thresholding, pixel-wise mean and covariance values and thresholding and several advanced and combined methods.

In the beginning of our experiments we have used a simple test sequence which was used by Prati et al. (2003) for their shadow detection algorithms and which they have provided to the research community. We tested two algorithms using a varying number of images from the test sequence. The first algorithm simply computes the average of the frames. The second algorithm is a state-of-the-art background estimator from a commercial image-processing library. Both algorithms were tested on a set of 4 and a set of 32 images from the sequence. Selected frames and the results are shown in figure 3.

The results reveal a problem of classic background estimation: the algorithms adapt slowly when no proper initialization is available. When the sequence contains only four frames clear ghosting effects are visible for both the simple averaging and the commercial implementation. When the length of the sequence is extended the results are better, but still both algorithms produce artifacts in the background image. Usually the number of frames needed for proper initialization is above 100.

Since we wish to keep the number of images and thereby the extra effort of manual image acquisition to a minimum, this behavior is not acceptable. Therefore we propose a different mechanism of computing the background image from a small set of images.

While most background estimation processes only consider one image at a time, a method suitable for processing a continuous stream of images, our method processes the full set of images at a time iterating over each pixel. We can think of the image sequence as a stack of per-pixel registered images, where each pixel exactly corresponds to the pixel of a different image at the same location. Thus we can create a pixel stack for each pixel location.

Using the basic assumption of background estimation that the background dominates over the altering foreground objects, we have to identify the subset of pixels within each pixel stack, which resembles the background. In other word we have to identify the set of pixels, which form the best consensus within each pixel stack. Projecting the pixels into a certain color space, we can think of the problem as a clustering task, where we have to find the largest cluster of pixels and disregard outliers. Figure 4 shows the pixels taken from the same location in four images of the test sequence of figure 3. The pixels are displayed in the two dimensions (red/green) of the three-dimensional RGB color space. The diagram shows three pixels in the upper right corner, which form a cluster and an outlier in the lower left corner.

Any unsupervised clustering technique could be employed in order to solve this task. Our approach is inspired by the RANSAC method introduced by Fischler and Bolles (1981). However, since the number of samples is small we do not have to randomly select candidates, but we can rather test all available samples. For every sample we test how many of the other samples are compatible. Compatibility is tested using a distance function within the selected color space. Either Mahalanobis distance or Euclidean distance can be used to compute the distance. If the distance is below a fixed threshold, the samples are determined to be compatible. The tested sample receives a vote for each compatibility.

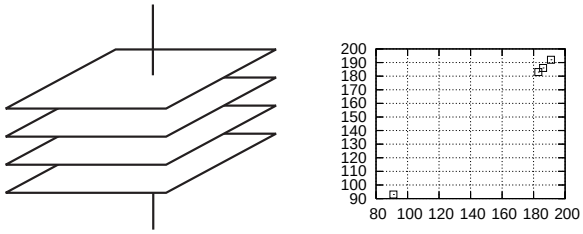


Figure 4: A sequence of images is treated as a stack of images. The graphic on the left shows a stack of four images. A single pixel is observed along a ray passing through all images at exactly the same position.
The diagram to the right shows the pixel values in red/green color space. Three pixels agree while one is clearly an outlier.

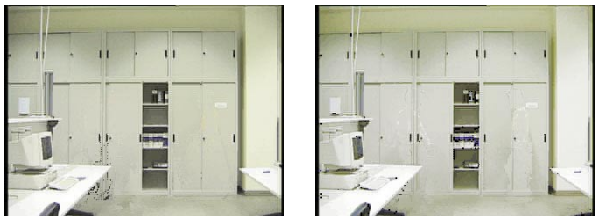


Figure 5: The result of the proposed algorithm on the test sequence. The left image shows the result obtained from computation in RGB color space, while the right image shows the result from HSV color space. In both cases the persons walking before the background were completely suppressed from only four images. The computation in HSV color space gives slightly better results.

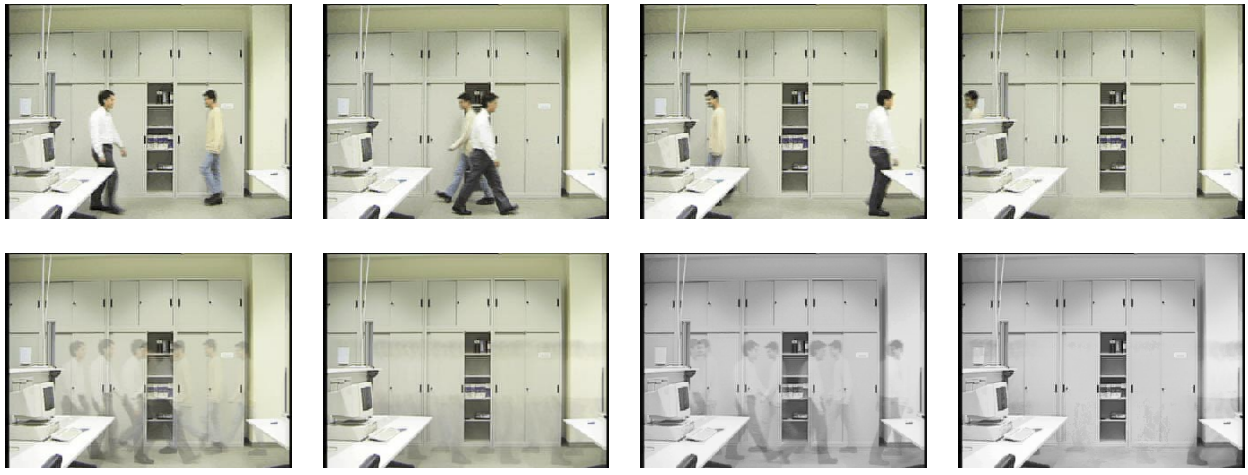
The sample with the largest number of votes is selected as the background pixel.

For the implementation we had to chose the appropriate distance function and select a proper color space. While some researches favor RGB Color space, others have reported good results from CIE color space (Coorg and Teller, 1999). A comparison of the results of the proposed algorithm from computation in RGB and HSV color space is given in figure 5. Simple Euclidean distance performed sufficiently in our test. For this test four input images were used. In both cases the persons walking before the background were completely suppressed.

Figure 6 shows a sequence of four images from a fixed viewpoint. The object depicted is the façade of a house imaged from across a busy street. Traffic and pedestrians partially occlude the lower portion of the façade. In figure 7 the results of the background estimation process are shown. The algorithm is able to remove all occlusions of moving objects from the images. Figure 7 (b) is an image that was computed by combining those samples, which received the least number of votes during clustering. The image thereby combines all outlier pixels and thus contains a combination of all occlusions. It serves as a comparison to image 7 (a) to assess the quality of the removal. In image 7 (c) all pixels are marked in white for which no unique cluster could be determined, i.e. all samples received only one vote or two clusters received the same number of votes. In these cases the pixel of the first frame was chosen to represent the background.

4 MULTI-VIEW FUSION

In section 3 we have shown how moving objects occluding a façade can be removed using several images taken from a fixed viewpoint. A static object occluding a façade will be imaged at



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

Figure 3: Background estimation results of different algorithms and varying sequence length. The top row shows 4 images from a sub-sequence of 32 frames. Images (a) and (b) show the result of averaging over 4 and 32 of the sequence. Images (c) and (d) show the results using a commercial implementation of a state-of-the-art background estimator, again using 4 and 32 images.

Figure 6: Four images from a fixed viewpoint of a façade imaged from across a busy street. Traffic and pedestrians partially occlude the lower portion of the façade.

(a)

(b)

(c)

Figure 7: The result of the proposed algorithm on the input sequence shown in figure 6. Image (a) contains the pixels, which received the most votes. Image (b) shows the pixels, which received the least votes and therefore contains all the occlusions combined. In image (c) pixels for which no unique decision could be made are marked white.

Figure 9: The result of the image fusion with the input images shown in figure 8. The bottom picture shows the final rectification onto the façade's rectangle. The statue has completely disappeared and most of the pedestrians were suppressed. The bottom picture shows the final rectification onto the façade's rectangle. Note the improvement to each of the rectified images in figure 8.

exactly the same location in the images when all images are taken from a single station. To reveal the occluded part of the façade additional viewpoints have to be included. In the top row of figure 8 a sequence of three images taken from three different stations is shown. The images depict the façade of a historic building, which is occluded by moving pedestrians in its lower portion. Additionally it is occluded by a stationary statue that extends across the full height of the façade. Obviously overlaying these images will not yield a per-pixel registered image stack. But since the background we are interested in is a façade, which is assumed to be planar, a perspective transformation can warp the images into a common projection.

One possibility to derive a proper transformation is to determine the final rectification of each input image individually. To achieve this it is necessary to mark the four corner points of the rectangular portion of the façade. The second row of figure 8 shows the result of such a rectification. We can observe how the statue seems to move across the image. This example makes it obvious that we have successfully transformed the problem of a static object to the problem of a moving object, which we have solved earlier.

However marking the four corner points of the façade is a non-optimal choice, since the image measurements are imprecise. Furthermore it cannot be guaranteed that all corner points are visible in every image. An alternative to compute the transformation is to warp the images to a different plane than to the façade. Actually any plane could be used. A proper choice is to warp the images to the same view as an arbitrarily selected key frame. In effect this transforms the images to the plane of the image sensor of the selected key frame.

To compute this transformation four corresponding points have to be measured for each input image. The bottom row of figure 8 shows the three images warped to the view of the last image of the sequence. These images form a per-pixel registered image stack and can therefore be processed with the method we have introduced in section 3. The result of the method is shown in figure 9. The statue has completely disappeared and the façade is free of most occlusions. Slight inaccuracies visible as blurring are caused by the non-planarity of the façade.

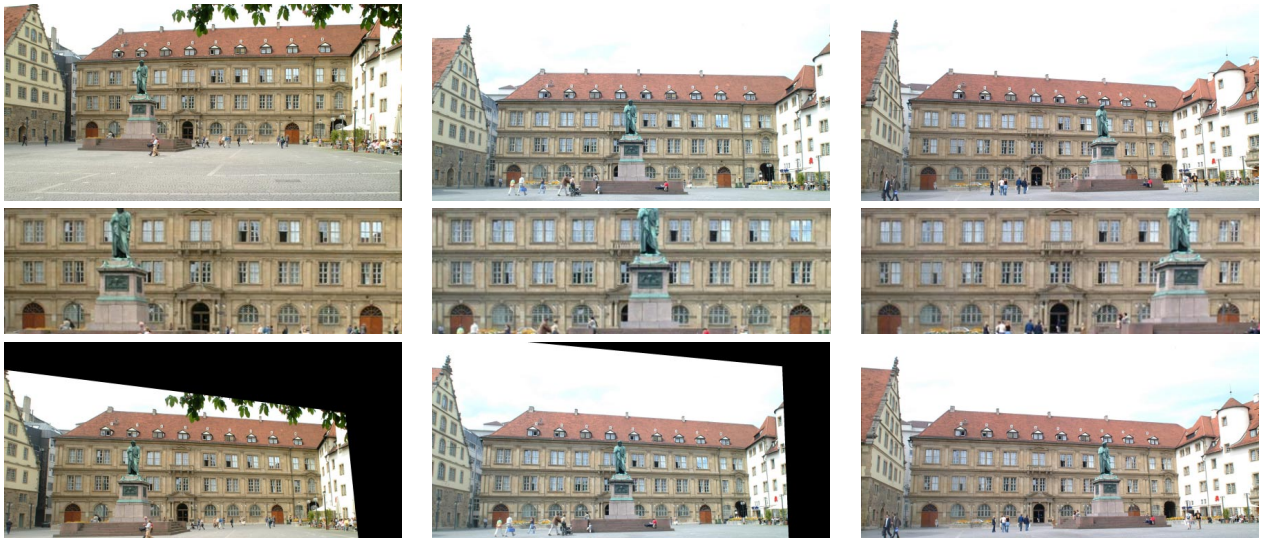


Figure 8: A complex example of a façade occluded by pedestrians and a statue of Schiller. The top row shows three images taken from three different stations. The middle row shows the three images rectified individually. The bottom row shows the three images transformed to the plane of the third image.

5 SUMMARY

We have demonstrated a simple approach for multi-image fusion for the generation of occlusion-free textures for façade texturing. Both the cases of moving objects and stationary objects occluding a façade can be handled in a unified manner. The approach does not use complex 3D computations. Specifically it does not require the determination of the exterior orientation of the sensor or the use of control points in object space. The approach is suitable for use with uncalibrated cameras since it only requires the computation of the perspective projection of images. Arguments can be made whether the lens distortions of the camera ought to be calibrated. Our experience has shown that the distortions introduced by the façade's relief during rectification are by far larger than the distortions caused by quality lenses.

Our approach requires the measurement of four points per image to be included in the fusion process. Comparing this to the simple single image approach for façade texturing, we see that the manual work load will increase linearly with the number of images. Due to this moderate need of user interaction it holds the potential for large-scale use.

Of course full automation is always desirable. Using a procedure, which is able to automatically detect buildings in terrestrial imagery, will lead to full automation. We have presented such a procedure in a previous publication (Böhm et al., 2002). We have described a method for automated appearance-based detection of buildings in terrestrial images. From an image with a given approximated exterior orientation and a three-dimensional CAD Model of the building, we were able to detect the exact location of the building in the image. The method used the combination of an image device and some hardware to approximately measure orientation. In future work we aim to integrate this procedure with the results presented here to achieve full automation of multi-image fusion.

References

Böhm, J., Haala, N. and Kapusy, P., 2002. Automated appearance-based building detection in terrestrial images. In: ISPRS Commission V Symposium, International Archives on Photogrammetry and Remote Sensing, Vol. 34number 5, pp. 491–495.

Bornik, A., Karner, K. F., Bauer, J., Leberl, F. and Mayer, H., 2001. High-quality texture reconstruction from multiple views. Journal of Visualization and Computer Animation 12(5), pp. 263–276.

Coorg, S. and Teller, S., 1999. Extracting textured vertical facades from controlled close-range imagery. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 625–632.

El-Hakim, S., Brenner, C. and Roth, G., 1998. An approach to creating virtual environments using range and texture. In: IAPRS, Vol. 32number 5, Hakodate, Japan, pp. 331–338.

Fischler, M. A. and Bolles, R. C., 1981. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. Communications of the ACM 24(6), pp. 381–393.

Haala, N., Böhm, J. and Kada, M., 2002. Processing of 3d building models for location aware applications. In: ISPRS Commission III Symposium, International Archives on Photogrammetry and Remote Sensing, Vol. 34number 3, pp. 138–143.

Kada, M., Roettger, S., Weiss, K., Ertl, T. and Fritsch, D., 2003. Real-time visualisation of urban landscapes using open-source software. In: Proceedings of ACRS 2003 ISRS, Busan, Korea.

Prati, A., Mikic, I., Trivedi, M. M. and Cucchiara, R., 2003. Detecting moving shadows: Algorithms and evaluation. IEEE PAMI 25(7), pp. 918–923.

Toyama, K., Krumm, J., Brumitt, B. and Meyers, B., 1999. Wallflower: Principles and practice of background maintenance. In: ICCV99, pp. 255–261.

Wang, X., Totaro, S., Taillandier, F., Hanson, A. and Teller, S., 2002. Recovering facade texture and microstructure from real-world images. In: Proc. 2nd International Workshop on Texture Analysis and Synthesis, pp. 145–149.