Proc. 2nd International Workshop on Texture Analysis and Synthesis, in conjunction with ECCV 2002, June 2002, pp. 145--149

Recovering Facade Texture and Microstructure from Real-World Images

Xiaoguang Wang, Stefano Totaro, Franck Taillandier, Allen R. Hanson, and Seth Teller

Abstract— **We present a set of algorithms that recovers detailed building surface structures from large sets of urban images containing severe occlusions and lighting variations. An iterative weighted-average algorithm is introduced to recover high-quality consensus facade texture. 2D and 3D methods are combined to extract microstructures, facilitating urban model refinement and visualization.**

Keywords— **texture, texture fusion, microstructure, surface geometry**

I. Introduction

EXTRACTING and rendering detailed 3D urban environments is a difficult problem. The main bottleneck lies in the need for human intervention in current systems, preventing them from being scalable to large datasets [13]. A large body of research exists for automating some of the processes, including acquisition of large real-world datasets [1], [3] and reconstruction of coarse 3D geometric models (mainly at the level of buildings) [4], [5], [7], [9]. Detailed analysis of facade texture and *microstructure* (surface structures such as windows that possess few supporting pixels due to insufficient image resolution) has been very limited.

Texture and microstructure in real imagery are important because they provide high visual realism as well as cultural and functional information of the urban site. Interactive extraction methods are not preferable, given the large number of pixels and structures present in many situations (e.g. more than a thousand windows for four or five buildings). The major difficulty for automatic extraction lies in the severe quality degradation in real-world images caused by various factors, including (1) varying resolution due to perspective effects, (2) noise introduction during acquisition, (3) non-uniform illumination caused by lighting variations and complex environments, (4) occlusions caused by *modeled* objects (such as other buildings) and *unmodeled* ones (such as trees, utility poles, and cars). A system must be capable of dealing with all these coexisting factors in order to recover a high-quality texture representation.

Multi-view methods have been proposed for texture fusion/recovery, such as interpolation methods [6], reflectance models [11], and inpainting techniques [2]. These methods do not handle occlusions automatically. Wang and Hanson [14] introduced a system that determines modeled occlusions, but not unmodeled ones. Coorg and Teller [5] developed a median-based technique that repairs unmodeled occlusions; however, the method may cause blurred or disrupted boundaries of structures.

We develop a set of algorithms for automating the extraction process. Sec. II describes an iterative, weighted-average approach to high-quality facade texture recovery. Sec. III introduces 2D and 3D methods for microstructure extraction. Sec. IV concludes the paper with discussions.

II. Texture Recovery

We describe a new method to obtain a realistic facade texture map while removing occlusions and effects of illumination variations. Input to this method is a set of images annotated with intrinsic camera parameters and reasonably accurate (but not exact) camera pose, as well as a coarse geometric model, mainly the facade planes of buildings in the site (these pieces of information are available using the algorithms introduced in Sec. I).

As a preprocessing step, the input images are rectified into *facade images*, i.e. images of a facade under orthographic projection. This happens only to a subset of the input images in which the facade is visible. Fig. 1(a2, b2, c2, d) shows some sample facade images in our experiments. Note the significant lighting variations across these images and the strong occlusions caused by modeled/unmodeled objects.

Texture fusion is the basic technique for removing the degradation effects. To facilitate fusion, the facade images are normalized by linear gray-level stretching; the resulting *luminance-normalized facade images* (*LNF images*) have the same average luminance and thus are comparable to one another.

The core of our method is a weighted-average algorithm that generates a *consensus texture facade image* (*CTF image*) for each facade from its LNF images:

$$Y_{\mathrm{CTF}}[i,j] = \sum_{\tau} Y^{\tau}_{\mathrm{LNF}}[i,j] * w^{\tau}[i,j],$$

$$\sum_{\tau} w^{\tau}[i,j] = 1,$$

Xiaoguang Wang is currently with Cognex Corporation, Natick, MA 01760, USA; this paper reflects his work when he was officially affiliated with the University of Massachusetts Amherst (xwang@cs.umass.edu).

Stefano Totaro is with Dipartimento di Elettronica ed Informatica, University of Padua, Italy (tost@dei.unipd.it).

Franck Taillandier is with Institut Géographique National, Saint-Mandé Cédex, France (franck.taillandier@ign.fr).

Allen R. Hanson is with Department of Computer Science, University of Massachusetts, Amherst, MA 01003, USA (hanson@cs.umass.edu).

Seth Teller is with MIT Laboratory for Computer Science, MA 02139, USA (seth@graphics.lcs.mit.edu).

This work was funded in part by DARPA DACA76-97-K-0005, ARPA/ARL DAAL02-91-K-0047, ARPA/ATEC DACA76-92-C-0041, and ARO/ARL DAAG55-97-1-0026.

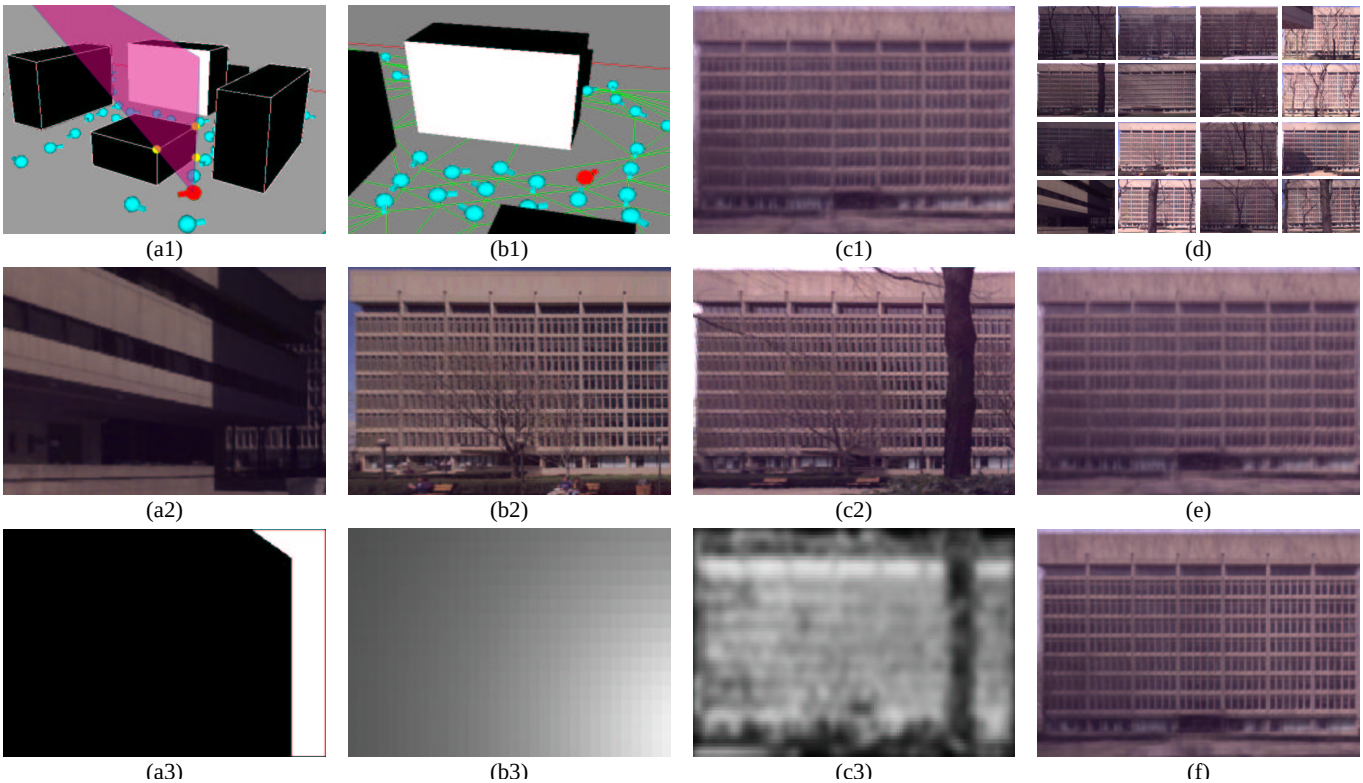


Fig. 1. Texture recovery. (a) environment mask [a1: camera position, a2: LNF image, a3: mask]; (b) obliqueness mask [b1: camera position, b2: LNF image, b3: mask]; (c) correlation mask [c1: a version of CTF image, c2: LNF image, c3: mask]; (d) sample original facade images of this wall; (e) initial CTF image without deblurring; (f) CTF image after iterative deblurring.

in which Y^{τ}_{LNF} is LNF image τ, Y_{CTF} is the fused CTF image, and w^{τ} is the weight factor determined by three *masks* described below. A mask is an image whose pixel value indicates the relative importance of the corresponding pixel in the LNF image. The three masks measure three different physical attributes.

Environment Mask is a binary mask that specifies whether a pixel is occluded by a modeled object (Fig. 1(a)). It is computed using the camera geometry and the coarse 3D model: $M^{\tau}_{\mathrm{E}}[i,j]$ is set to 0 if pixel $[i,j]$ is occluded; otherwise, it is set to 1.

Obliqueness Mask is a grey-scale mask that represents the obliqueness of a facade as seen from the camera (Fig. 1(b)) and is also computed from the geometry:

$$M^{\tau}_{\mathrm{O}}[i,j] = \cos\theta^{\tau}(i,j),$$

in which $\theta^{\tau}(i,j)$ is the camera viewing angle at $[i,j]$ on the facade measured from the normal of the facade.

Correlation Mask is a grey-scale mask intended to account for the effects of unmodeled occlusions and local illumination variations. To compute this mask, an initial CTF image is needed, and the mask is calculated using a standard linear correlation between the LNF image and the initial CTF image (Fig. 1(c)):

$$M^{\tau}_{\mathrm{C}}[i,j] = \frac{\mathrm{Cov}_{i,j}[Y^{\tau}_{\mathrm{LNF}}, Y_{\mathrm{CTF}}]}{\mathrm{Var}_{i,j}[Y^{\tau}_{\mathrm{LNF}}]\mathrm{Var}_{i,j}[Y_{\mathrm{CTF}}]},$$

where $\mathrm{Cov}_{i,j}$ and $\mathrm{Var}_{i,j}$ are based in an image window, centered at $[i,j]$, of a predetermined size (8×8 in our experiments). The initial CTF image can be obtained using traditional methods [5], [14] or using the weighted-average algorithm without the correlation mask. In practice, the weighted-average algorithm runs iteratively (see below), and in each iteration a new CTF image is used to calculate M^{τ}_{C}.

The weight w^{τ} at pixel $[i,j]$ of LNF image τ is computed using the following formulas:

$$W^{\tau}[i,j] = M^{\tau}_{\mathrm{E}}[i,j]M^{\tau}_{\mathrm{O}}[i,j]M^{\tau}_{\mathrm{C}}[i,j],$$

$$w^{\tau}[i,j] = \frac{W^{\tau}[i,j]}{\sum_{\tau} W^{\tau}[i,j]}.$$

The CTF image thus obtained may look blurred (Fig. 1(e)) because the LNF images may not be perfectly registered due to errors in camera parameters. A deblurring process is used that rewarps the source LNF images to align with the CTF image [12]:

$$[\,u,v,1\,]^T \cong P[\,u',v',1\,]^T,$$

which warps pixel $[u',v']$ to $[u,v]$ using P. Our goal is to find a warp P that minimizes E_{CTF}:

$$E_{\mathrm{CTF}} = \sum_{u,v}[e(u,v)]^2,$$

$$[e(u,v)]^2 = W^{\tau}[u',v'](Y_{\mathrm{CTF}}[u,v] - Y^{\tau}_{\mathrm{LNF}}[u',v'])^2.$$

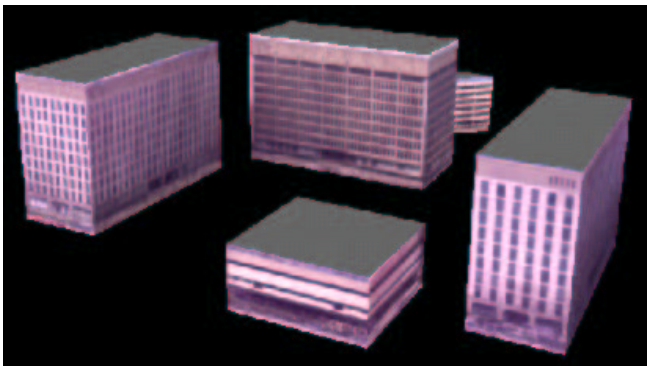


Fig. 2. CTF textured model.

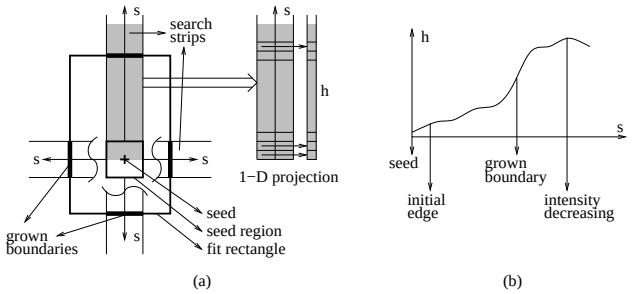


Fig. 3. Oriented region growing (ORG). (a) one iteration of region growing; (b) determining the grown boundary location.

Note that the overall weight mask W^{τ} is used, reflecting the degree of confidence we have in Y^{τ}_{LNF}. The Levenberg-Marquardt algorithm [10] is employed to solve the constrained minimization problem. It is an iterative process; in each iteration, a new P is calculated, the LNF images are rewarped, and the weighted-average algorithm is rerun to obtain a new CTF image.

The deblurring process is also executed iteratively. Recall that the correlation mask M_C is dependent on an initial CTF image. After deblurring, the new CTF image is used to compute a more accurate M_C, which then again updates the CTF image and triggers another round of deblurring. The convergence of the recursion is ensured by stopping when the difference between two successive CTF images is sufficiently small.

Experiments were carried out to test against a dataset acquired at Technology Square, MIT, an office park of four buildings. About 4,000 images were captured at 81 distinct locations in this site. LNF images were extracted for each facade. Fig. 1(f) shows the CTF result of the iterative weighted-average algorithm on a facade, for which 28 LNF images were extracted from the database and used to generate the CTF image. Most of the degradation effects in 1(d) were satisfactorily removed, and the luminance is reasonably consistent across the entire CTF image. Fig. 2 shows a perspective view of the resulting textured model of this site.

III. Microstructure Recovery

Accurate 3D recovery of detailed facade surface structures from images is difficult, mainly due to the small ratio of the depth of structures (on the order of centimeters) to the camera-to-wall distance (on the order of tens of meters). We developed a hybrid method that combines information of both 2D shape/position and 3D depth for microstructure recovery.

The 2D shapes/positions of microstructures are extracted from the CTF images. Although CTF images provide a good texture representation, they still suffer from degradation such as global illumination variation (e.g. lower parts of walls are usually darker than upper parts), making a global thresholding algorithm ineffective for microstructure detection. We develop a 2D extraction algorithm by combining an *oriented region growing* (*ORG*) module and a *periodic pattern fixing* (*PPF*) module. The algorithm detects a generic class of objects that exhibit a regular size, pattern, and orientation. The microstructures are symbolically represented as a set of disjoint 2D rectangles, each having two vertical and two horizontal edges. Many windows in urban areas fit well into this representation.

ORG is a "bottom-up" extraction module, starting from a seed rectangle in the CTF image and growing iteratively into a best-fit structure (Fig. 3). It is capable of handling global illumination variations, only requiring that the intensity of the structure be locally evident. Details of ORG can be found elsewhere [14].

PPF is a "top-down" process for fixing the pattern of symbolic microstructures extracted by ORG. A high-level constraint is employed by this module, enforcing that structures of similar size have a periodic pattern in horizontal and vertical directions on the facade (Fig. 4). Microstructures are firstly classified into groups, each of which represents a certain size range. The horizontal and vertical periods of a microstructure group are then found using a standard clustering algorithm based on their neighboring distances. Missing structures are filled in using interpolation and/or extrapolation.

In reality, the periodic pattern constraint may not be strictly satisfied on all buildings. To ensure that missing candidate are only filled in for structures that exist, a "bottom-up verification" test is used to verify their existence in the LNF images before interpolation/extrapolation. On each LNF image, a vertical and horizontal edge detection algorithm is performed at locations of missing candidates (if they are visible). A missing candidate is accepted only if there are sufficiently many LNF images that support the candidate.

Fig. 5 shows the results of the ORG/PPF modules on one facade. A total of ten facades, representing the major buildings in Technology Square, were used to test the extraction algorithm. Among the 1146 manually counted windows on the ten facades, 1119 of them have been extracted correctly. Only 27 are missing, accounting for 2.4% of the actual windows. The 1119 extracted windows accounts for 98.9% of the extraction results (totally 1133); there are only 14 false positives (or 1.2%). An examination

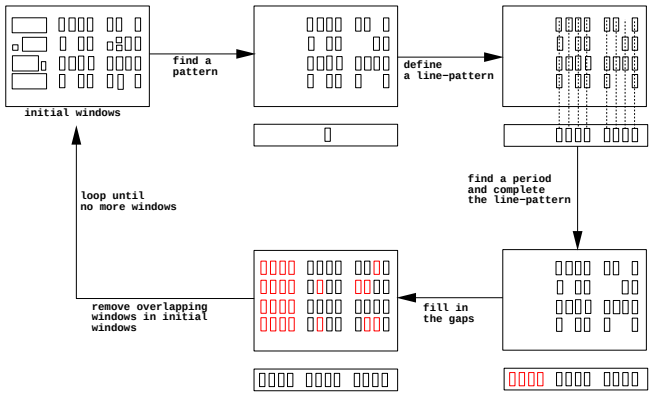


Fig. 4. Periodic pattern fixing (PPF).

of the images shows that the missing windows are mainly caused by low contrast of the windows and their blurred edges. False positives are due mostly to the complex intensity patterns caused by rectangular structures on the walls that look like windows but are not.

The goal of 3D depth estimation is to recover the relative depth of microstructures on the facade surfaces. We use a revised version of Fua and Leclerc's 3D mesh generation algorithm [8] for depth recovery. This approach has a number of advantages over traditional stereo analysis based on image pairs: it takes information from any number of images; geometric constraints can be added (particularly advantageous for largely planar surfaces); and additional information, such as that of occlusions, can be incorporated. The algorithm starts with a planar surface and deforms it by iteratively minimizing $E(S)$:

$$E(S) = \lambda_{\mathrm{D}} E_{\mathrm{D}}(S) + \lambda_{\mathrm{S}} E_{\mathrm{S}}(S) + \lambda_{\mathrm{G}} E_{\mathrm{G}}(S),$$

$$\lambda_{\mathrm{D}} + \lambda_{\mathrm{S}} + \lambda_{\mathrm{G}} = 1,$$

in which E_{D} is a planar surface constraint, E_{S} a correlation-based stereo constraint, and E_{G} a geometric constraint. Details for these components and the minimization scheme can be found elsewhere [8].

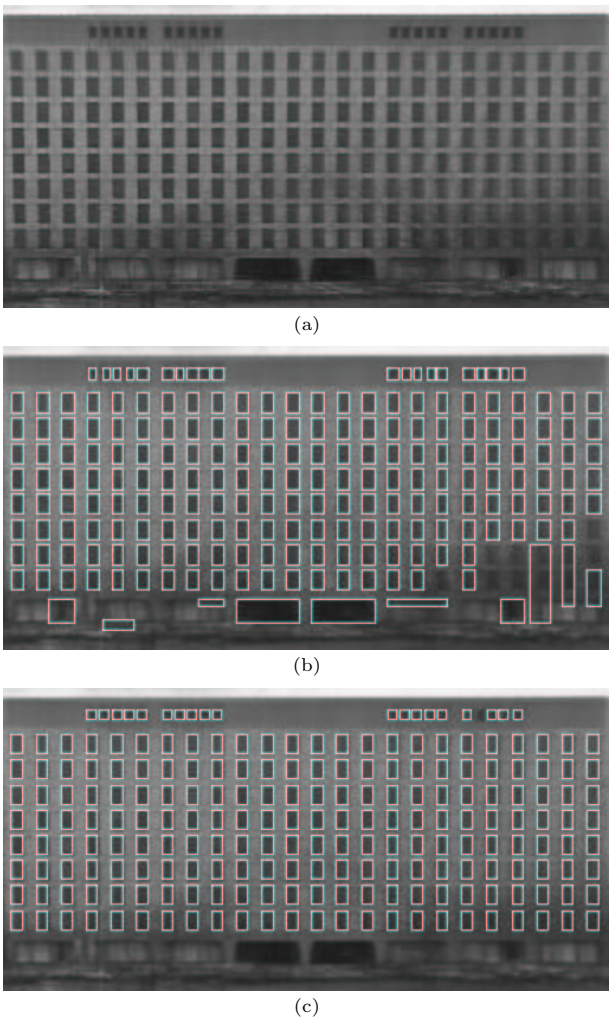
In order to take advantage of knowledge obtained in Sec. II, the E_{S} constraint is modified by excluding occlusions from stereo computation. We define an *occlusion-removed facade image* (*ORF image*) by

$$Y^{\tau}_{\mathrm{ORF}}[i,j] = Y^{\tau}_{\mathrm{LNF}}[i,j] M^{\tau}_{\mathrm{E}} M^{\tau}_{\mathrm{C}'},$$

where M^{τ}_{E} is the environment mask that represents the modeled occlusion, and $M^{\tau}_{\mathrm{C}'}$ is a binary version of the correlation mask M^{τ}_{C} that represents the unmodeled occlusion. We use Y^{τ}_{ORF} rather than Y^{τ}_{LNF} to calculate E_{S}, thus focusing on the visible parts of each facade.

The mesh algorithm was applied to all facades in our dataset (we set $\lambda_{\mathrm{D}} = 0.1$, $\lambda_{\mathrm{S}} = 0.9$, and $\lambda_{\mathrm{G}} = 0$). Fig. 6 shows the depth map of one of the facades. The results are noisy, but the general pattern of windows is evident.

2D shape and 3D depth are combined for a final representation of the microstructures. We made an assumption



(a) (b) (c)

Fig. 5. Symbolic window extraction. (a) CTF image; (b) results of ORG; (c) results of PPF.

that a facade surface can be approximated by two depth layers: the wall layer and the window layer. Detailed non-flat portions on a wall are beyond the scope of our current discussion. With this assumption, the average depth inside the 2D rectangles is used to represent the depth of the 3D microstructures. Fig. 7 shows an example of the recovered textured 2D/3D structures.

IV. Discussion

We described a suite of algorithms for detailed urban environment analysis, including an iterative, weighted-average algorithm for recovering a consensus texture map, nearly free from occlusions (modeled and unmodeled) and local illumination variations, and a hybrid 2D/3D method to extract surface microstructures.

It is worth noting that the proposed algorithms are effective for solving a generic set of urban environment extraction and refinement problems, in which the wall surfaces are largely planar and the microstructures are mainly rectangular. Many buildings in urban environments satisfy these constraints. (The PPF algorithm, which requires a periodic pattern of the microstructures as an additional

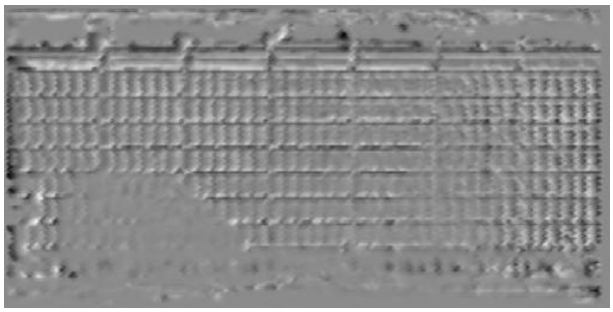


Fig. 6. Facade depth estimation.

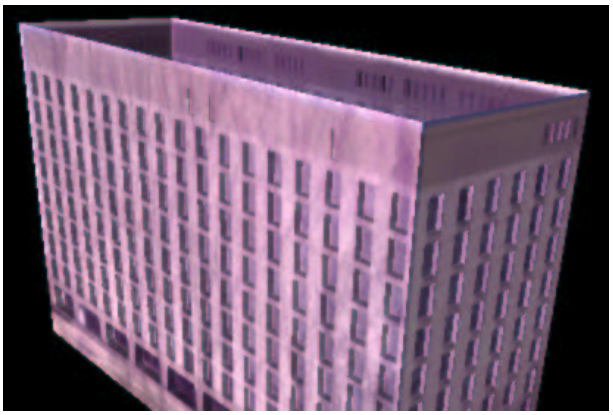


Fig. 7. Microstructure visualization.

constraint, is an optional module for large-size buildings and is not a necessary component for small structures.) In addition, practicality is one of the design emphases of the algorithms. The deblurring process allows the algorithm to tolerate camera pose error that often arises in real applications. The 2D microstructure module extracts structures of any size greater than 3×3 pixels, needing no interactive parameter adjustment. Our experiments show that only about a dozen original facade images, with quality shown in Fig. 1(d), are needed for texture recovery with a satisfactory result; this is a reasonable number of images in practice.

There are several directions in which the algorithms can be extended to solve more general problems. First, the extracted 2D microstructures can provide partial geometric constraints in $E_{\mathrm{G}}(S)$ for depth estimation. How to improve the depth estimation by incorporating the partial constraints is a topic for future study.

Second, the ORG algorithm is designed to extract a generic class of objects. Although a large variety of surface microstructures fit into this class, it has two major limitations: the shape of each microstructure is approximated by a rectangle, and the luminance of the microstructure must be relatively uniform. For more special problems, special object detection modules should be used as a successor of ORG/PPF.

Third, the global illumination variation problem has not been solved in the CTF algorithm. For rendering purposes, a better texture representation may be demanded. This problem could be solved using the heuristics given by the periodic pattern of microstructures. As the microstructures share a common shape and common period, they should also share the illumination in normal cases. An illumination adjustment algorithm could thus be designed to take advantage of this.

Acknowledgments

The authors would like to thank Eric Amram and Neel Master for their technical contributions to the experiments in the paper.

References

[1] M. Antone and S. Teller, "Automatic Recovery of Relative Camera Rotations for Urban Scenes," *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 282-289, June 2000.

[2] M. Bertalmio, G. Sapiro, V. Caselles, and C. Ballester, "Image Inpainting," *SIGGRAPH*, 2000.

[3] M. Bosse, D. de Couto, and S. Teller, "Eyes of Argus: Georeferenced Imagery in Urban Environments," *GPS World*, pp. 20-30, April 2000.

[4] R. Collins, C. Jaynes, Y. Cheng, X. Wang, F. Stolle, A. Hanson, and E. Riseman, "The Ascender System for Automated Site Modeling from Multiple Aerial Images," *Computer Vision and Image Understanding*, vol. 72, no. 2, pp. 143-162, 1998.

[5] S. Coorg and S. Teller, "Extracting Textured Vertical Facades From Controlled Close-Range Imagery," *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 625-632, June 1999.

[6] P. Debevec, C. Taylor, and J. Malik, "Modeling and Rendering Architecture from Photographs: a Hybrid Geometry and Image-based Approach," *SIGGRAPH*, pp. 11-20, August 1996.

[7] O. Firschein and T. Strat (Ed.), *RADIUS: Image Understanding for Imagery Intelligence*, Morgan Kaufmann Publishers, San Francisco, CA, 1996.

[8] P. Fua and Y. Leclerc, "Using 3-Dimensional Meshes to Combine Image-Based and Geometry-Based Constraints," *European Conf. on Computer Vision*, pp. B:281-291, 1994.

[9] H. Mayer, "Automatic Object Extraction from Aerial Imagery – A Survey Focusing on Buildings," *Computer Vision and Image Understanding*, vol. 74, pp. 138-149, 1999.

[10] W. Press, S. Teukolsky, W. Vetterling, and B, Flannery, *Numerical Recipes in C, The Art of Scientific Computing*, Second Edition, Cambridge University Press, 1992.

[11] Y. Sato, M. Wheeler, and K. Ikeuchi, "Object shape and Reflectance Modeling from Observation," *SIGGRAPH*, pp. 379-387, August 1997.

[12] R. Szeliski, "Video Mosaics for Virtual Environments," *IEEE Computer Graphics and Applications*, vol. 16, no. 2, pp. 22-30, March 1996.

[13] S. Teller, "Automated Urban Model Acquisition: Project Rationale and Status," *Image Understanding Workshop*, pp. 455-462, Monterey, CA, 1998.

[14] X. Wang and A. Hanson, "Surface Texture and Microstructure Extraction from Multiple Aerial Images," *Computer Vision and Image Understanding*, vol. 83, no. 1, pp. 1-37, July 2001.