AUTOMATIC EXTRACTION OF BUILDING FEATURES FROM IMAGE DATA: HOW FAR ARE WE?

KUI YUE, RAMESH KRISHNAMURTI
School of Architecture, Carnegie Mellon University, Pittsburgh, Pennsylvania, USA
kyue@andrew.cmu.edu, ramesh@cmu.edu

Abstract. This paper examines how far we are towards automatic extraction of building features, by comparing two pipelines from image data to building features: *an ideal pipeline*, based on the requirements of an on-going project, and *a realistic pipeline*, based on current computer vision technologies.

Keywords. Automatic extraction: building features; image data.

1. Introduction

This paper explores automatic extraction of building features from image data and poses the question on how far we are, realistically, from attaining this goal. This query stems from our current investigation into how the interior layout of a building can be determined if given a set of building features and a shape grammar that describes the style of the target building. The input feature is a 3D building shell model, and includes the footprint for each story, as well as exterior features, such as windows, chimneys and surrounding buildings. Our particular focus is on conventional building types, that is, buildings comprised of rectilinear spaces and components, or approximated as such.

Our research was conducted through a set of test cases, namely, Queen Anne houses in Pittsburgh (Flemming, 1987); row-houses in Baltimore (Hayward, 1981) and Portland; and high-rise apartments in Baltimore. We proposed a general approach (Yue et al., 2008) based on the fact that, when applied exhaustively, a shape grammar can generate, as a tree, the entire layout space for a style. The approach begins with an estimated initial layout. Spatial and topological constraints are extracted, which are used to

prune the layout tree. The layouts that remain correspond to the desired results.

A practical difficulty of this approach is in obtaining building feature input; the geometry of an arbitrary target building is usually unknown, and time-consuming to generate. Automatic generation of building feature input is, of course, desirable. Prior experience (Yue and Krishnamurti, 2007) inspires us to investigate automating the generation of building features from image data by taking advantage of state-of-art computer vision technologies.

We focus on two types of image data employed in computer vision research: photo and range images. Their basic characteristics are briefly reviewed. Requirements on building feature input are analyzed to identify an *ideal pipeline*. State-of-art techniques are reviewed and, accordingly, possible pipelines proposed, with closest to the ideal selected as the *realistic pipeline*. Lastly, gaps between this and the ideal pipeline are discussed.

2. Photo images and range images

2.1 PHOTO IMAGES

Quite simply, a photo image records the world through color or brightness information, and photo imaging systems have become cheaper and more ubiquitous. Points are basic units in describing the geometry of an object. Measuring points using photo images is the precise goal of traditional photogrammetry (Mikhail et al., 2001). Modern computer technology relieves photogrammetry from reliance upon specific physical devices.

The basic approach to measuring point coordinates is triangulation (Figure 1a). By taking photographs from at least two different locations, ‘lines of sight’ are developed from each camera to points on the object. The lines of sight are mathematically intersected to produce the 3D coordinates.

This approach necessarily implies a sub-procedure, namely, that of establishing correspondences between pixels in different views. Automatic pixel matching is difficult and costly; many solutions have been proposed, including the famous RANSAC algorithm (Fischler and Bolles, 1981).

A major drawback is that photogrammetric measurements are inherently dimensionless. For instance, there is no way to distinguishing between pictures of full-sized cars and their match-box models. Mathematically, an extracted model is correct to scale; for exact dimensions, we need at least one known measurement. As discussed later, compared to range images, photo images are relatively noiseless, although rectification and correction of radial distortion are often required.

2.2 RANGE IMAGES

Range images store the depth at which the ray associated with each pixel first intersects the scene as observed by a range sensor. A Cartesian transformation converts range pixels to points in space, resulting in a *3D point cloud*. Range images are 'easier' in that the image data points explicitly represent scene surface geometry. The incident mesh is virtually ready for varying uses, for example, monitoring the progress of construct sites (Shih and Wang, 2004). However, to extract the geometry as basic shapes, such as lines, planes, cylinders, etc., most low-level problems that exist for photo images remain the same, such as filtering, segmentation, and edge detection (Paul, 1988).

Range images are mainly captured using 3D laser scanners, which typically have limits on the view in terms of horizontal and vertical angles. To capture a given scene, multiple scans are required. These scans have to be further aligned, also known as *registration*, and optionally merged together, as each scan is represented in a local coordinate system relative to the laser scanner. Many algorithms had been proposed to automate the registration of a large number of individual scans; for instance, Huber and Hebert (2001) introduce fully automated registration based on spin-images.

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

Figure 1. (a) Measuring 3D coordinates by triangulation (From http://www.geodetic.com: Dec 2008) (b) range/intensity crosstalk (adapted from Tanget al. 2007), (c) mixed pixels (adapted from Tanget al. 2007), (d) formation of mixed pixels.

Many laser scanners determine the range by measuring the shift in phase between an amplitude-modulated continuous-wave emitted beam and its reflection. This principle leads to two detrimental effects, namely range/intensity crosstalk (Figure 1b) and mixed pixels (Figure 1c) (Herbert

and Krotkov, 1992; Tang et al., 2007). Range/intensity crosstalk is due to the fact that a range measurement is not independent of the reflective properties of the observed surface. The influence is so significant that useless range data can be produced. Mixed pixels happen when a laser beam partially hits the front surface and then hits another surface behind (Figure 1d). The fact that the range is measured by integrating over the entire projected spot leads to the result that the measured range can be anywhere along the line of sight. The implication is that occluding edges of scene objects are often unreliable.

Although there are algorithms proposed to eliminate mixed pixels in special cases (Tuley et al., 2005; Tang et al., 2007), simple general cost-effective remedies do not seem to exist at the moment (Herbert and Krotkov, 1992). As a result, these two effects greatly reduce the accuracy, down to centimeter from the advertised millimeter level.

3. An ideal pipeline

The desired building feature input has to be given as typed objects. That is, the type and geometry of a particular object are known. Notably, we are able to distinguish between the geometry representing a window and that of a door. This requires that the object recognition algorithm is capable of both extracting object geometry, as well as annotating its type.

Figure 2 shows an ideal pipeline based on the above. The pipeline first builds a co-located model of range and photo images; that is, we know which pixel in the photo image corresponds to a point in 3D space. In the next step, geometries are extracted and photo images are automatically annotated. Each basic shape in the extracted geometries is typed by using the annotation of the corresponding pixels in the photo images. In this way, an annotated 3D model is created. The desired features of the model are outputted into XML, to serve as input to the layout determination program.

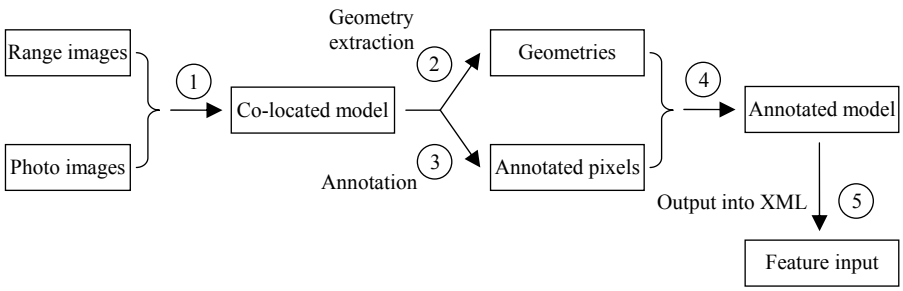


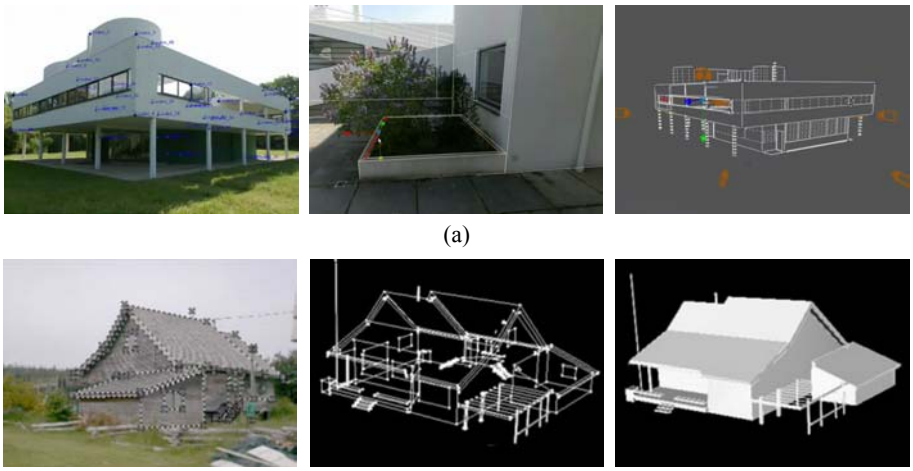
Figure 2. An ideal pipeline

4. State-of-art

The ideal pipeline is related to both modeling-from-reality and appearance-based object recognition in computer vision research. The former aims at photorealistic reconstruction of scenes; the latter identifies the existence of an object in a given photo image, as well as its location.

4.1 MODELING-FROM-REALITY

Modeling-from-reality is one of the more challenging and well-studied problems in computer vision. Mainstream techniques have been developed using photo images or range images or a combination of the two. Techniques for photorealistic reconstruction typically use both photo and range images.



(a)

(b)

Figure 3. (a) An example adapted from an ImageModeler demonstration (http://usa.autodesk.com/adsk/servlet/index?siteID=123112&id=11983371: Dec 2008) (b) An example adapted from a PhotoModeler demonstration (http://www.photomodeler.com/applications/architecture_and_preservation/examples.htm: Dec 2008)

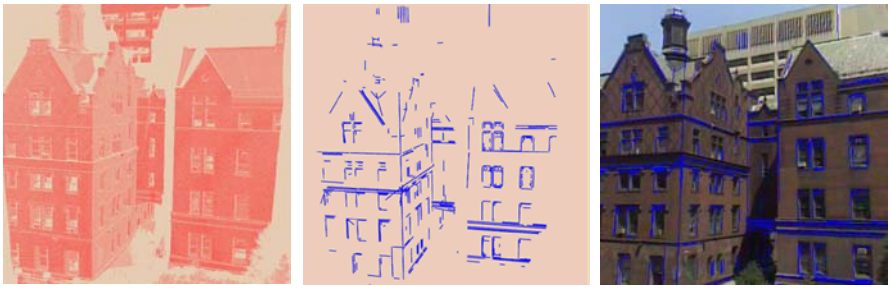
In the literature, *image-based modeling* refers to modeling from multiple photo images. The determination of the geometry of objects from multiple views is not the solely in the domain of computer vision; *photogrammetry* (Mikhail et al., 2001), which dates back to the mid-19th century, also attempts to precisely recover quantitative geometric information from multiple photo images. There is commercial software based on computer vision and photogrammetric technology. PhotoModeler by Eco System (http://www.photomodeler.com/index.htm: Dec 2008) and ImageModeler

(http://usa.autodesk.com/adsk/servlet/index?id=11390028&siteID=123112: Dec 2008) by Autodesk are two among the better known products. ImageModeler (Figure 3a) relies on marker points specified by the user to calibrate camera position and parameters. Once calibrated, modeling is a manual procedure using polygonal primitives. Likewise, modeling in PhotoModeler (Figure 3b) is mainly manual, although it can automate many sub-procedures, such as, automated marking and matching.

There are models that directly use meshes incident with registered range images for various objects, for example, statues (Levoy et al., 2000), heritage sites (Ikeuchi et al., 2003) and underground mines (Huber and Vandapel, 2006). The basic modeling procedure comprises capturing of range images, their alignment, merging range images into a mesh object, and optionally texture-mapping. As geometry information is given as meshes, this kind of technique does not meet our requirements.

Without a priori knowledge of the type of the objects in a scene, it is generally hard to extract object surfaces from range images. This is because a range image treats an entire scene as an entity; thus, it is difficult to automatically determine which subset of points belongs to an object. Various techniques for geometry extraction have been developed. These fall into two categories: those that segment a point cloud based on such criteria as proximity of points or similarity of locally estimated surface normals, and those that directly estimate surface parameters by clustering and locating maxima in a parameter space. The former obtain the geometry as meshes, the latter, though, is more robust, and is only used for shapes that are described by a few parameters such as planes or cylinders. For our purpose, these latter methods provide more appropriate geometry. Examples include Faber and Fisher (2002) who use knowledge-based architectural models as constraints to build geometric models with the quality of CAD models; Vosselman et al. (2004) who explore techniques for recognizing objects as planes, cylinders or spheres in industrial plant and urban landscape contexts.

Stamos and Allen (2002) develop a systematic approach to the problem of photorealistic 3D model acquisition from a combination of range and photo images. Their approach utilizes parallel and orthogonal constraints, which abound in urban environments. As a result, this approach works well on urban scenes consisting of conventional buildings. The system takes a set of 3D range images from different viewpoints and a set of 2D photo images of the scene, creating first a 3D solid model, which describes the geometry of the scene, then recovering the positions of the 2D cameras with respect to the extracted geometric model, and finally, photorealistically rendering the scene by texture-mapping the associated photographs on the model. Figure 4 shows the results.



(a) (b) (c)

Figure 4. a) range image of the scene, b) extracted 3D lines, c) 3D lines projected on the photo image (all adapted from Stamos and Allen, 2000).

4.2 APPERANCE-BASED OBJECT RECOGNITION

Appearance-based object recognition can be used to annotate objects. With a co-located model, we can determine the object type of the extracted geometry by detecting the object type the corresponding pixels belong to.

Appearance-based object recognition is still active research. The earliest work utilized global descriptions such as color or texture histograms. The main drawback to this was sensitivity to real world variability, such as viewpoint and light changes, clutter and occlusion. Global methods have been gradually supplanted by part-based methods in the last decade. Part-based object models combine appearance descriptors of local features with a representation of their spatial relations. While part-based models offer a satisfying way to representing many real-world objects, learning and inference problems for spatial relations remain complex and computationally intensive, especially in a weakly supervised setting where the location of the object in a training image has not been marked by hand. The bag-of-features model (Csurka et al., 2004) has the advantage of simplicity and computational efficiency, though it fails to represent the geometric structure of the object class. Various approaches have been developed to overcome this; examples are SIFT descriptors (Sivic et al., 2005), novel kernels (Zhang et al., 2007), etc.

5. An realistic pipeline

Using commercial software such as ImageModeler and PhotoModeler offers a practical option. Photo images at different angles are input to such software, with a 3D annotated model manually created, and the desired

building features then output to XML. This approach is, however, time-consuming, as there is limited automation involved.

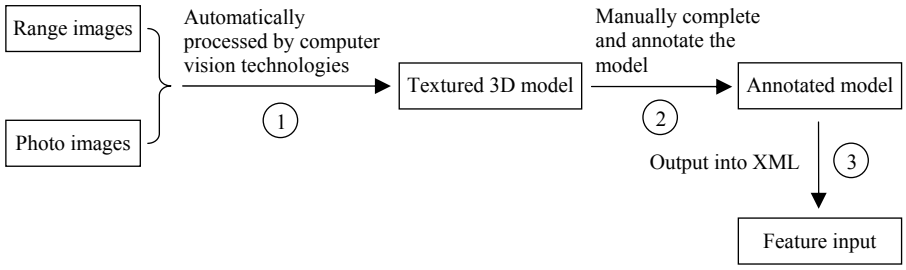


Figure 5. A realistic pipeline

On the other hand, the technique, developed by Stamos and Allen (2002) in reconstructing photorealistic texted 3D model, is fully automated. However, for our purposes, there are potential problems. For instance, there is no way of guaranteeing that the desired geometry information can be extracted; moreover, it is unlikely that the photo-image pixels are correctly annotated. The extracted geometry is typically *loosely* connected —it would be hard to automatically convert such geometry information to annotated objects even if annotations were available. Consequently, there are manual operations involved. However, on the motto that there is presently nothing better, a pipeline based on this technique is still preferable to the manual approach; at least, there is some automation involved. We choose one such pipeline as the *realistic* pipeline (Figure 5). Note that the textured 3D model also serves as a co-located model.

6. Gap between the ideal and realistic pipeline

Significant progress has been made in computer vision research; nonetheless, there is still a noticeable gap between the ideal and reality. Current approaches are still mainly purely geometry-based, with little concern to the specific type of the underlying object, without knowledge of which, makes extraction of complete geometry extremely difficult. To know the object type, it is necessary to improve current object recognition algorithms so that object types are accurately identified. This seems to be a kind of chicken-and egg situation. However, this cycle can be broken, with some promise, by using a combination of range and photo images, the former primarily for geometry and the latter for annotation.

7. Conclusion

In response to the query posed in this paper, it is fair to state that we are still remarkably far from being able to automatically extract building features from a set of image data. For those processes or projects requiring as-is building features, the best we can achieve is a semi-automatic procedure along the lines of the realistic pipeline we propose. Even then such a semi-automatic procedure may still not be 'realistic' in the following sense that no system is yet commercially or publically available, although similar systems do exist in academic research. In other words, it would require a significant amount of coding before we can take advantage of even a semi-automatic procedure. For small-scale projects, however, it is realistically feasible, and hence probably a good solution, to manually generate building features with the aid of commercial software such as ImageModeler or PhotoModeler.

Acknowledgements

The authors wish to thank Daniel Huber and Pingbo Tang for their helpful discussions and comments to this research, which was supported in part by from US Army Corps of Engineers, Engineer Research and Development Center – Champaign. Any opinions, findings, conclusions or recommendations presented in this paper are those of the authors and do not necessarily reflect the views of CERL.

References

Csurka, G., Dance, C. R., Fan, L., Willamowski, J. and Bray, C.: 2004, Visual categorization with bags of keypoints, *Workshop on Statistical Learning in Computer Vision, ECCV.*

Faber, P. and Fisher, B.: 2002, How can we exploit typical architectural structures to improve model recovery? *3D Data Processing Visualization and Transmission,* Padova, Italy.

Fischler, M. A. and Bolles, R. C.: 1981, Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography, *Commun. ACM,* **24**, 381-395.

Flemming, U.: 1987, More than the sum of parts: the grammar of Queen Anne houses, *Environment and Planning B: Planning and Design,* **14**, 323-350.

Hayward, M. E.: 1981, Urban Vernacular Architecture in Nineteenth-Century Baltimore. *Winterthur Porfolio,* **16**, 33-63.

Herbert, M. and Krotkov, E.: 1992, 3D measurements from imaging laser radars: how good are they? *Image Vision Comput.,* **10**, 170-178.

Huber, D. and Hebert, M.: 2001, Fully automatic registration of multiple 3D data sets. *IEEE Computer Society Workshop on Computer Vision Beyond the Visible Spectrum (CVBVS 2001).*

Huber, D. and Vandapel, N.: 2006, Automatic three-dimensional underground mine mapping, *The International Journal of Robotics Research,* **25**, 7-17.

Ikeuchi, K., Nakazawa, A., Hasegawa, K. and Ohishi, T.: 2003, The great buddha project: modeling cultural heritage for VR systems through observation, *The Second IEEE and ACM International Symposium on Mixed and Augmented Reality,* Tokyo.

Levoy, M., Pulli, K., Curless, B., Rusinkiewicz, S., Koller, D., Pereira, L., Ginzton, M., Anderson, S., Davis, J., Ginsberg, J., Shade, J. and Fulk, D.: 2000, The digital Michelangelo project: 3D scanning of large statues *in* Akeley, K. (ed), *Siggraph 2000, Computer Graphics Proceedings*, ACM Press / ACM SIGGRAPH / Addison Wesley Longman.

Mikhail, E. M., Bethel, J. and McGlone, J. C.: 2001, *Introduction to modern photogrammetry*, John Wiley and Sons, Inc.

Paul, J. B.: 1988, Active, optical range imaging sensors, *Mach. Vision Appl.,* **1**, 127-152.

Shih, N.-J. and Wang, P.-H.: 2004, Point-Cloud-Based Comparison between Construction Schedule and As-Built Progress: Long-Range Three-Dimensional Laser Scanner's Approach. *Journal of Architectural Engineering,* **10**, 98-102.

Sivic, J., Russell, B. C., Efros, A. A., Zisserman, A. and Freeman, W. T.: 2005, Discovering objects and their location in images*, Tenth IEEE International Conference on Computer Vision, ICCV 2005.*

Stamos, I. and Allen, P. K.: 2000, 3-D model construction using range and image data, *IEEE Conference on Computer Vision and Pattern Recognition.*

Stamos, I. and Allen, P. K.: 2002, Geometry and texture recovery of scenes of large scale, *Comput. Vis. Image Underst.,* **88**, 94-118.

Tang, P., Huber, D. and Akinci, B.: 2007, A Comparative Analysis of Depth Discontinuity and Mixed Pixel Detection Algorithms, *The 6th International Conference on 3-D Digital Imaging and Modeling,* Montréal, , IEEE.

Tuley, J., Vandapel, N. and Hebert, M.: 2005, Analysis and Removal of Artifacts in 3-D LADAR Data, *Proceedings of the 2005 IEEE International Conference on Robotics and Automation, ICRA 2005.*

Vosselman, G., Gorte, B. G. H., Sithole, G. and Rabbani, T.: 2004, Recognizing structure in laser scanner point clouds. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences,* Freiburg, Germany.

Yue, K., Hickerson, C. and Krishnamurti, R.: 2008, Determining the interior layout of buildings describable by shape grammars. *CAADRIA08,* Chiang Mai,.

Yue, K. and Krishnamurti, R.: 2007, Extracting building geometry from range images of construction sites, *CAADRIA07,* Nanjing.

Zhang, J., Marsza, M., ek, Lazebnik, S. and Schmid, C.: 2007, Local Features and Kernels for Classification of Texture and Object Categories: A Comprehensive Study, *Int. J. Comput. Vision,* **73**, 213-238.