Interactive 3D Building Modeling Using a Hierarchical Representation

Sung Chun Lee* and Ram. Nevatia*

*Institute for Robotics and Intelligent Systems,
University of Southern California
Los Angeles, California 90089, USA
{sungchun|nevatia}@usc.edu

Abstract

Modeling and visualization of city scenes is important for many applications including entertainment and urban mission planning. Models covering wide areas can be efficiently constructed from aerial images. However, only roof details are visible from aerial views and ground views are needed to provide details of the building facades for high quality fly-through visualization or simulation applications. Different data sources provide different levels of necessary detail knowledge. We need a method that integrates the various levels of data. We propose a hierarchical representation of 3D building models for urban areas that integrates different data sources including aerial and ground view images. Each data source gives us different details and each level of the model has its own application as well. Through the hierarchical representation of 3D building models, large area site modeling can be done efficiently and cost-effectively. This proposal suggests efficient approaches for acquiring each level model and demonstrates some results of each level including the integration results.

1 Introduction

Accurate 3D building models for a city are useful for a variety of applications such as 2D and 3D GIS, fly-through rendering, and simulation for mission planning. Each application requires different aspects of the 3D building model. 2D GIS needs the roof boundary of buildings and 3D GIS requires the height and roof boundary of buildings. Fly-through rendering demands textures of building facades in addition to the 3D building model. Furthermore, a walk-through or high-quality rendering application requires detailed 3D structural description of a building facade.

These different levels of information for 3D building models are computed or obtained from different data sources. For example, information such as the roof boundary or height can be obtained by using aerial images or LIDAR (LIght Detection And Ranging) data. Facade information such as its texture, descriptive information, or detailed 3D structure, can be computed from multiple ground view images.

A variety of computer vision problems arise in the process of obtaining 3D building models. The first is the 3D building reconstruction to acquire the initials. The creation of 3D building models from aerial view analysis has been researched extensively [1, 2, 3]. Second, we have to perform calibration, or pose estimation, of uncalibrated ground view images to obtain facade related information. Even though general camera calibration is still an open problem in computer vision, the calibration process for ground view images of an architectural scene can be solved by using the regularity knowledge of building structure such as an orthogonality of vertical and horizontal lines and parallelism of roof and ground lines.

In this paper, we attempt to integrate the processes of obtaining the different level knowledge for building models from various data sources by deriving a hierarchical representation of a 3D building model. To implement hierarchical representations for 3D building models, we exploit the concept of *Level Of Detail* (LOD) from virtual reality literature to represent the different level of knowledge. The LOD is used for storing a number of representations of an object, where the complexity of each representation is varied so that an approximate model can be selected according to the complexity of the application. In this paper, we define the representations of 3D building models as followings:

Level 1: Structural information of building
Level 2: Facade texture information
Level 3: Detailed geometry of building facade

The concept of the hierarchical representation for building models is illustrated in Figure 1.

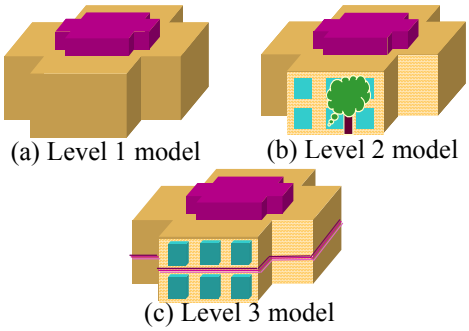


Figure 1. A hierarchical building representation.

1.1 Related Work

Recently, ground view based approaches for obtaining full geometric information of 3D building models, have been popular since data sources such as multiple ground images or video sequences are more readily obtained than aerial view images. In addition, the ground view images provide high resolution modeling results.

In [4], data sources are multiple ground view images and the polyhedral primitive 3D models. The drawback of this approach, however, is that it is hard to scale to large area site modeling because laborious user interactions are required for a large number of models.

New approaches of integrating more than one data source such as multiple ground view images and GPS data have been attempted to model 3D buildings [5, 6]. This approach can be applied to modeling a relatively large site due to the global positioning capability supported by a GPS device, but it is limited to model only simple shape buildings and does not capture the roof structure.

Werner and Zisserman [7] recently tried to reconstruct the detail structure of building facade by fitting primitive blocks. Performance of their method is highly dependent on the accuracy of line matching.

A Bayesian approach is used to reconstruct 3D building models in Cipolla at el. [8]. They estimate relative rotation between 3D models and camera by using three orthogonal vanishing points. This method requires vanishing points from orthogonal families of lines. Robertson and Cipolla [9] add a geo reference functionality to their previous work [8] by exploiting a 2D map. Given the extracted layers on building facade, Dick at el. [10] reconstruct the 3D facade primitives using prior 3D primitive blocks based on a Bayesian framework. Recently, Dick at el. [11] improve their previous method by adding more components such as alignment and symmetry among the facade primitives by exploiting architectural knowledge. However, these approaches depend on strong priors, which makes their algorithms have a limitation on scalability.

Stamos and Allen [12] use dense range data to reconstruct the detailed 3D building models by the swept multiple high precision range data. Range data allows making detailed models but it expensive and cumbersome to acquire. Fruh and Zakhor [13] use ground level range data, ground view images, and aerial view photographs (including digital roadmap). Their approach fuses different data to compensate for the accumulated errors of range data. However, the generated 3D building models are a mesh structure and do not make the structure explicit.

1.2 Overview

Using a hierarchical building representation, the reconstruction process of more complex level models is aided by simpler level models. The hierarchical building representation reduces the complexity of computer vision problems and makes the task of generating building models feasible.

For level 1 model (simpler model), we use the knowledge that a flat roof building is likely to have the roof parallel to the ground and walls perpendicular to the ground. The representation of level 1 model consists of the 2D roof boundary and its height. As shown in Figure 2, level 1 models are acquired using this knowledge, by a human or an automatic process from multiple aerial view images as will be explained in Section 2.

To obtain level 2 model, we exploit the knowledge from level 1 models, which is the 3D information of wireframe building models such as its 3D vertices, boundary lines, and faces. With 3D building models and automatically or interactively extracted vanishing points, we estimate the pose of the ground view camera using only two 3D to 2D point correspondences, as explained in Section 3.

We use the knowledge that the orientation of the detailed structures of building facade (level 3 models) is perpendicular to the building facade that contains them. Then, we use the calibrated ground view camera information from level 2 and 3D model information from level 1 to reduce user interactions for creating level 3 models, as described in Section 4.

2 Acquisition of Level 1 models

In this section, we describe a cost-effective user interactive method to reconstruct a 3D architectural wireframe models (level 1 models) from multiple aerial images for urban or suburban site modeling.

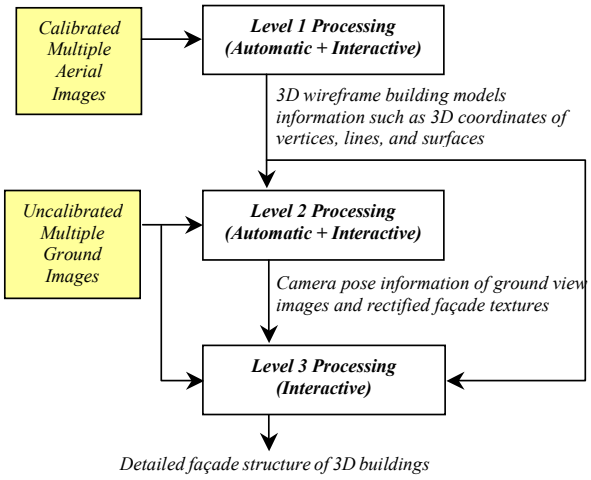
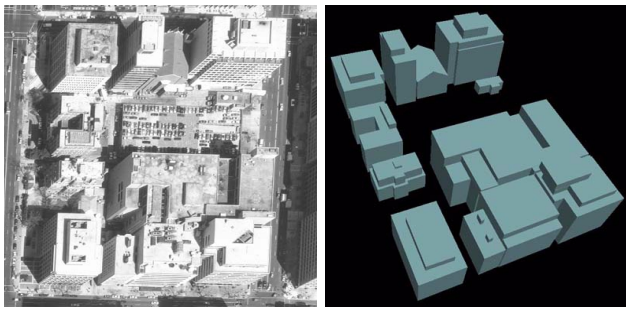


Figure 2. The overview of the proposed system.

The method consists of processes to initiate the construction of a model, and to add or subtract 3D blocks to it with minimum interaction and maximal use of automatically pre-computed 3D information. A model starts with a *seed*. Seeds are generated in two ways: by running the automatic system [1] up to its capability, or by the user interactive system described in [14]. User interaction starts with the seed building. The user can add or subtract blocks to or from the seed as needed. The added (or subtracted blocks) can be rectangular or triangular to allow polygonal shapes. Our goal is to reduce the number of user interactions. The proposed method satisfies this goal as the number of pointer (mouse) *clicks* needed is less than the number of corners on the roof outline of the buildings since adding or subtracting operations require only one or two clicks and no further height computation for sub-blocks. Figure 3 shows a result of extracting level 1 models by our method. For more details, please refer [14].



(a) An aerial view image (b) Level 1 models

Figure 3. Extracted level 1 models from aerial images.

3 Acquisition of Level 2 models

In this section, we explain how to obtain building facade texture information (Level 2 model) from ground view images by exploiting the obtained 3D building (level 1) models.

To obtain facade texture information, we need to estimate the global position and orientation of the camera used for taking the ground view images. In an urban area, it is not easy to find ground views that capture an entire building since many buildings are close to each other, and narrow streets limit the field of view. The small field of view problem prevents the estimation of external camera parameters by traditional methods, which use point correspondences, because not enough identifiable feature points such as corners of building are visible. Because of this constraint, a line based approach should be considered for pose estimation of ground view cameras.

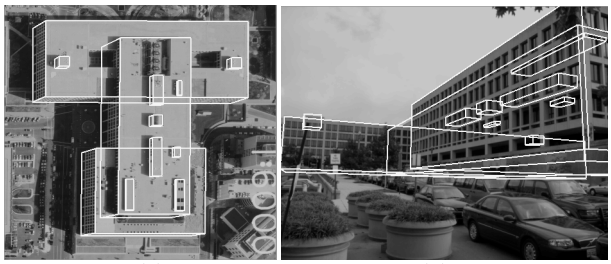
Automatically obtainable image features, such as vanishing points have been used to compute the external and some internal parameters of the camera [8, 15]. These methods require vanishing points from orthogonal families of lines, however building's facades are not always orthogonal. Our approach does not require orthogonal families of lines but instead uses the knowledge of the 3D angles between them (from level 1 models).

3.1 Pose estimation of ground view images

As we know the angle between the two facades from the level 1 models, we can infer a real orthogonal vanishing point under the assumption that the principal point of the camera is known (to be the center of the image). Three *inferred* orthogonal vanishing points are used to compute the camera rotation [8].

Given the external rotation matrix and one 3D to 2D point correspondence, the 3D position of the camera center must be on the 3D line, which intersects the 3D point in the model and has a directional vector from the center of projection to the corresponding 2D point. With two point correspondences, the exact position of the camera can be obtained by intersecting two lines. For more details, see [16, 17].

Figure 4 shows a result of pose estimation for the ground view image. Complex 3D building wireframe models (level 1 models) in Figure 4 (a) are obtained from aerial images. Figure 4 (b) shows the projected 3D building wireframe models on the ground view images.



(a) (b)

Figure 4. Result of pose estimation for an uncalibrated ground view image: (a) level 1 models from aerial view image, (b) A pose-estimated ground view image.

3.2 Adjusting level 1 models at level 2 processing

3D building models derived from aerial images may be inaccurate due to the relatively low resolution of such images. Since we have high resolution ground view image calibrated, we can adjust 3D building models. In this section, we explain how to adjust the inaccurate level 1 models using the calibrated ground view image. Adjusting operations consist of height (building's roof or ground elevation) and roof side adjustments.

To measure a correct height under a perspective projection camera, we use a cross ratio, which is a projective invariant [18].

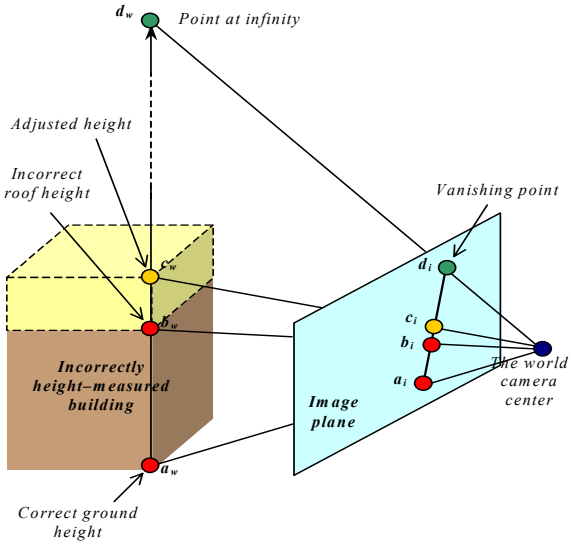


Figure 5. Adjusting height using the cross ratio.

As shown in Figure 5, the cross ratios of (a_w, b_w, c_w, d_w) and (a_i, b_i, c_i, d_i) are equal, where a_w is the ground height, b_w is incorrect roof height, c_w is the adjusted roof height, and d_w is the point at infinity, a_i, b_i, c_i, d_i are image projection of a_w, b_w, c_w, d_w respectively. Note that d_w is an imaginary point, but its image projection point, d_i, which is a vanishing point, is a real point on the image plane.

The cross ratio of (a_i, b_i, c_i, d_i) can be computed because all image points are known (a_i and b_i are from the 3D building model, c_i is from the user, and d_i was obtained during generation of level 2 models). Because d_w is the point at infinity, the cross ratio of (a_w, b_w, c_w, d_w) is only related with a_w, b_w, and c_w. The unknown point, c_w can be computed using the following equations:

$$Cross(a_i, b_i, c_i, d_i) = \frac{|a_i c_i||b_i d_i|}{|a_i d_i||b_i c_i|} = k$$

$$= Cross(a_w, b_w, c_w, d_w) = \frac{|a_w c_w||b_w d_w|}{|a_w d_w||b_w c_w|} = \frac{|a_w c_w|}{|b_w c_w|}$$

$$\Rightarrow |a_w c_w| = k|b_w c_w|$$

The side adjustment is a similar procedure and also uses the cross ratio and the respective vanishing point to compute the adjusted world point.

4 Acquisition of Level 3 models

So far, we have 3D building models, and we want to generate geometric 3D facade structures, such as building entry structures with some detailed structures of the building facade.

We define some primitive modeling operations such as indentation, protrusion, and columns for modeling geometric 3D facade structures. Our approach requires user interaction, but we reduce user effort because we already know the orientation of 3D building models in the calibrated ground view image. The primitive modeling operations do not require an orientational adjustment by user, rather they only require dimensional information such as width, height, and depth.

4.1 Modeling indentation structures

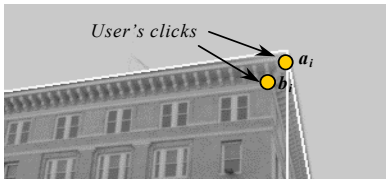
Due to roof eaves of the building, the vertical lines of roof boundary from 3D building model from aerial views may not match with the vertical lines from ground view images as shown in Figure 6. This recess gap from the roof boundary needs to be modeled to obtain correct facade textures from the ground view images. We define an *indentation* operation to deal with recess.

Our approach requires two clicks from the user, which indicate the height (a point, a_i) and depth (a point, b_i) of the indenting part from the ground view image as shown in Figure 7 (a).

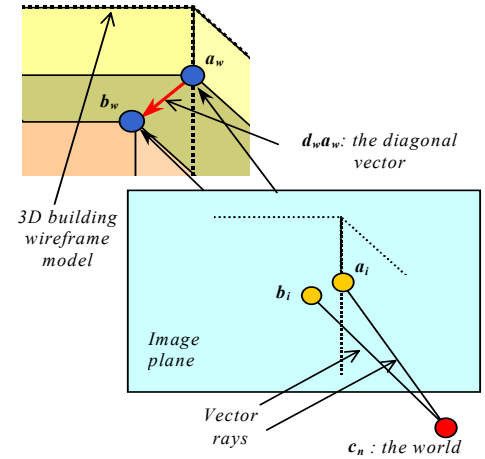
Since we know the position and rotation of the ground view camera, we can compute the 3D position (a_w and b_w) of the user's clicked points, a_i and b_i. The world

(a) An aerial view (b) A ground view

Figure 6. Mismatch of the 3D building wire frame model (a) from aerial views and the actual vertical wall (b) seen from ground views.



(a) User Interaction for modeling a recess indentation



(b) Illustration of computing 3D position of the clicked point

Figure 7. Modeling a recess indentation.

position (a_w) of the height reference point (a_i) is easily computed by intersecting a vector ray formed by a_i and c_n and a building facade surface as depicted in Figure 7 (b). Since the world position (b_w) of the recess reference point is not on the facade surface, but on the indented surface, we use a diagonal vector defined as $d_w a_w$, where d_w is an intersection of surface inner normal vectors of two facades which contain the point, a_w on their boundaries. Because b_w should be located on the direction of the diagonal vector, $d_w a_w$, the world position of the recess reference point can be computed by intersecting a vector ray, $b_i c_n$ and the diagonal vector.

There exist some indentations within a facade such as the main entry, windows, or decorative structures of a building. To model such indentations, we need three clicks which indicate 2D dimension (two clicks) and depth (one click) information.

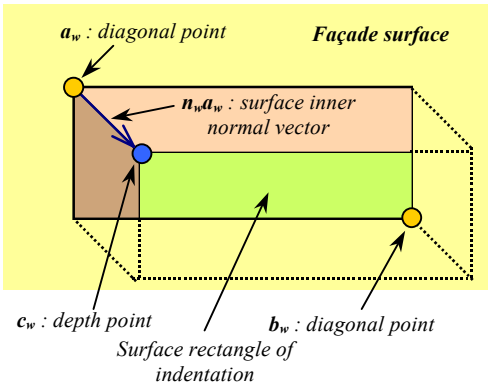


Figure 8. Modeling an inside indentation.

User clicks two diagonal points for a surface rectangle of indentation and one more click for depth position as depicted in Figure 8. Since the world diagonal points (a_w and b_w) are on the facade surface, they are computed by intersecting the facade surface and vector rays formed by the image diagonal points (a_i and b_i respectively) and the world position of the camera center (c_n), as for the recess reference points. However, since the world depth point is not located on the facade surface, it also requires the surface inner normal vector, $n_w a_w$ to compute the world position of the depth point (c_w) as shown in Figure 8. It can be located by intersecting the surface inner normal vector and a vector ray, $c_i c_n$, formed by the image depth point (c_i) and the world camera center.

4.2 Modeling protrusion structures

Like indentation, modeling protrusion requires three clicks which specify 2D dimension (two clicks) and depth (one click) information. In the protrusion case, the world depth point is on the building facade surface, but the world diagonal points are not located on the surface.

The world position of the depth point, c_w is obtained by intersecting a vector ray from the world camera center passing through a user clicked point (c_i) and the facade surface. The world position of one protruding point, a_w is acquired by intersecting a vector ray from the world camera center passing through a user clicked point (a_i) and the surface outer normal vector. Since we do not know the position of b_w as well as its counterpart, u_w, we conduct a one-dimensional search to compute the world position of another protruding point, b_w. The unknown point, b_w

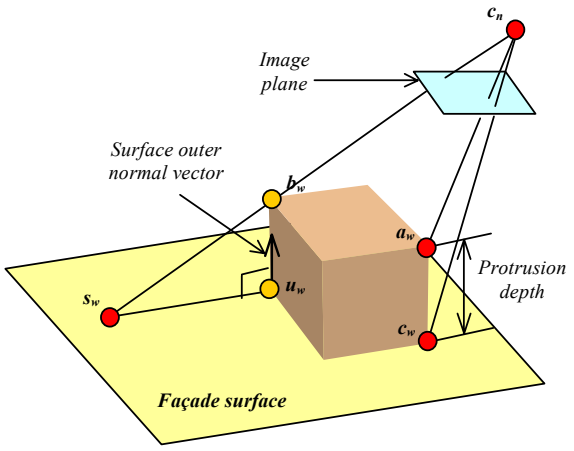
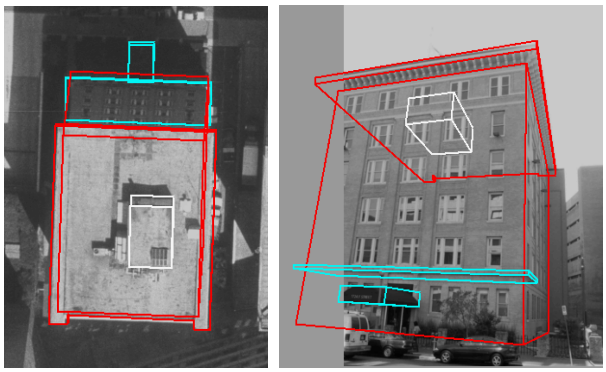


Figure 9. Finding a world protruding point.

should be located on the vector ray, $s_w c_n$, where s_w is the intersecting point of a camera ray vector passing b_i and the facade surface as illustrated in Figure 9. We search for a point through the line, $s_w c_n$ such that its perpendicular distance with the facade surface should be the same as the protrusion depth, which is the length of the known line segment, $a_w c_w$.

Figure 10 shows an intermediate result of modeling 3D protrusions and indentations.

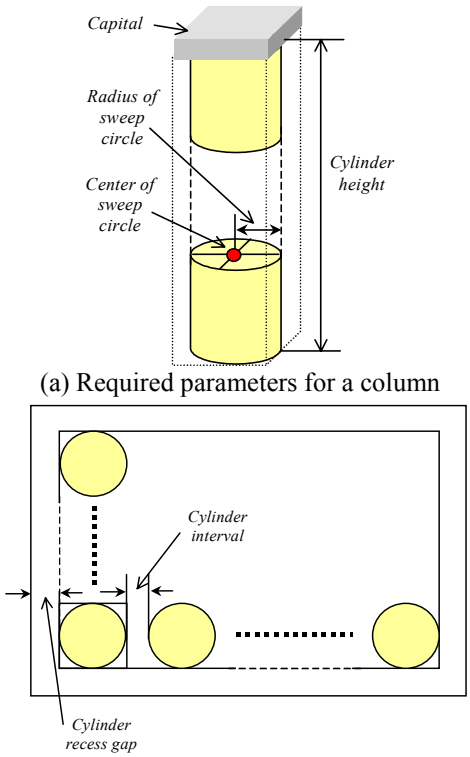


(a) An aerial view (b) A ground view

Figure 10. Intermediate result of modeling protrusions and indentations.

4.3 Modeling column structures

In our approach, we define a column structure to have a *cylinder* part and a *capital* part as illustrated in Figure 11 (a). Under the assumption of equal distance between columns, we need four parameters to model a group of columns: capital size (physical height of the capital), radius of sweep circle, cylinder recess gap, and cylinder interval as depicted in Figure 11.



(a) Required parameters for a column

(b) Required parameters for a group of columns

Figure 11. Required parameters for modeling column structures.

The number of cylinders in one roof side can be computed by the following equation:

$$l_c + 2r_c = \frac{l_{width} - 2g_r - 2r_c}{n-1}$$

$$n = \frac{l_{width} - 2g_r - 2r_c}{2c_r + I_c} + 1$$

where I_c is the cylinder interval, r_c is the radius of the sweep circle, g_r is the cylinder recess gap, n is the number of cylinders, and l_{width} is the length of the roof side that a group of cylinders are located beneath.

We need four clicks for creating one group of columns as depicted in Figure 12. The world positions of four user clicked points are computed by intersecting with building facade surface and each vector formed by the selected image point and the camera center. These four computed 3D points are used to define the required four parameters of radius of circle, cylinder recess gap, capital height, and cylinder interval as explained above. When the recess gap point is on the vertical line of the 3D model, we set this value to be zero.

Figure 13 shows an intermediate result of modeling a group of columns. 3D building (level 1) models are shown in Figure 13 (a) and intermediate level 3 models (protru-

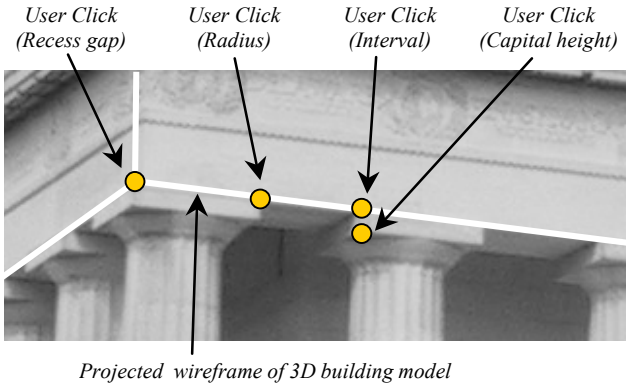


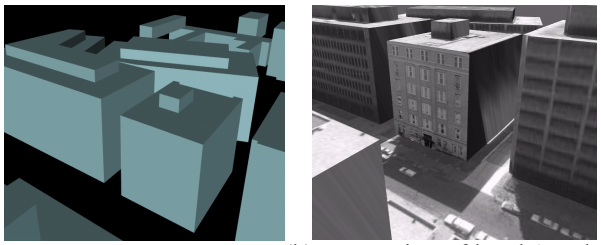
Figure 12. User Interaction for columns.

sions and indentations) are shown in Figure 13 (b) after their height and side are adjusted. Figure 13 (c) shows a group of column structures beneath the front architrave of the building.

5 Results and Discussion

We have tested our method with several buildings in the city of Washington D.C. We acquired the ground view images from the Internet and constructed 3D building models from aerial images by using an interactive building modeling system described in Section 2.1.

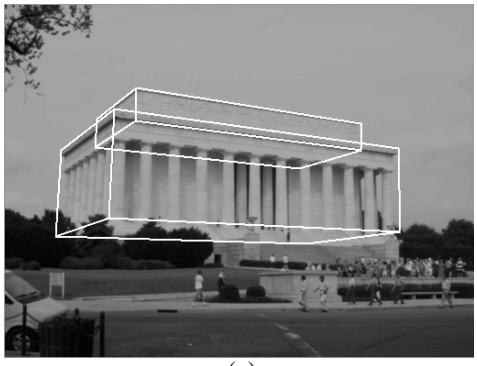
Figure 14 shows an integration result of level 1 and 2 models. We render a building cluster in Washington D.C using only aerial view images in Figure 14 (b) except one building that we integrated a building facade texture to.



(a) level 1 models (b) Integration of level 1 and 2

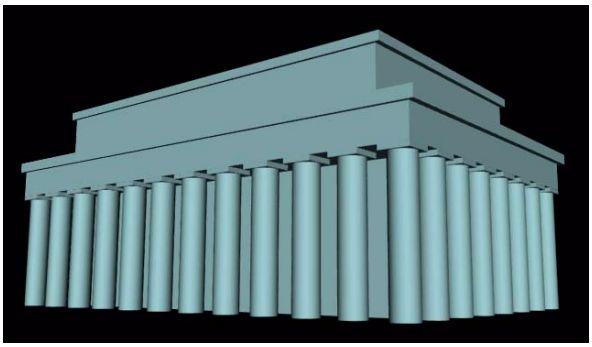
Figure 14. Integration of level 1 and 2 models.

The U.S. National Archive building is shown in Figure 15. The wireframe building models (level 1) are superposed on an aerial image as shown in Figure 15 (a). We estimated the pose of the ground view picture of this building using the method described in Section 2. Using this information, building models projected on a ground image are shown in Figure 15 (b) and VRML models (level 1) are rendered in Figure 15 (c). We modeled more detailed facade structure using the interactive method with the calibrated ground view image as described in Section 3. An intermediate result of level 3 models is shown in



(a)

(b)



(c)

Figure 13. Result of modeling a group of columns.

Figure 15 (d). The final 3D building model of the U.S. National Archive building is shown in Figure 15 (e) by adding a few groups of columns.

6 Conclusion

A hierarchical representation of the 3D building models is proposed. Information for different levels can be acquired from aerial and ground view images, or the integration of them depending on the complexity of the models. Information on different levels can be used for an appropriate purpose with low cost.

The proposed method requires much less user interaction than the Facade system [4]. In the Facade system,

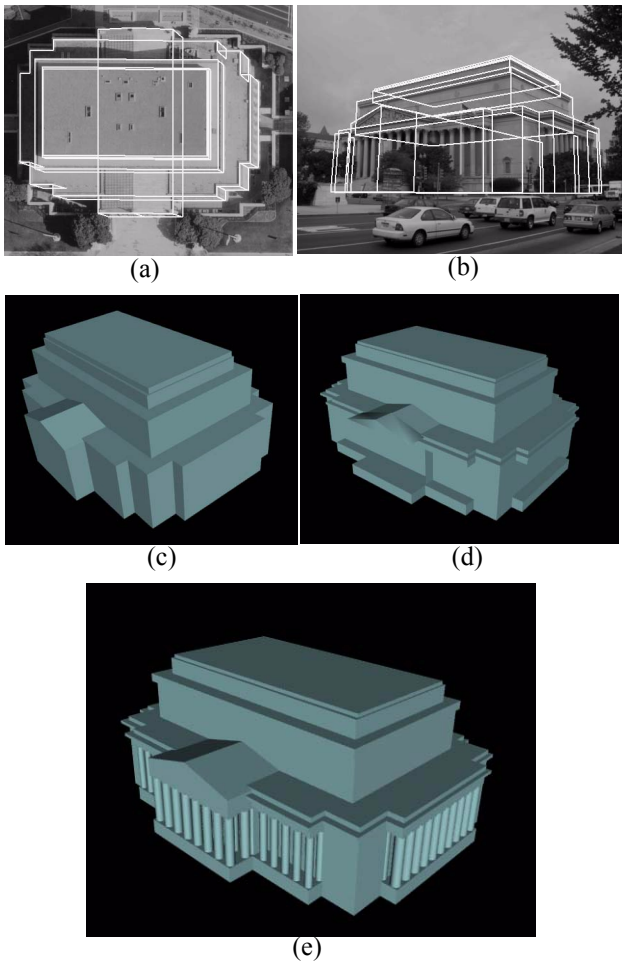


Figure 15. Final result of 3D reconstruction of U.S. Archive building.

user has to build the entire building with primitive blocks without knowing dimension information while our method constructs the metric 3D building models. Given 3D models, the Facade system still has to solve calibration and computing dimension problems, which requires laborious user clicks for model to image correspondences. For modeling a simple rectangular building, for example, the Facade system requires a process of selecting a primitive 3D model (3D box), drawing necessary image lines on the ground view image, and corresponding model to image lines. Our method requires at most 3 clicks for creating a 3D model and two model to image points correspondences for pose estimation process.

In addition, the proposed system is able to construct much more complex buildings than other image based automatic systems [5, 11]. One drawback of our approach is the need for aerial images which may not always be easily available.

References

[1] S. Noronha and R. Nevatia, Detection and modeling of buildings from multiple aerial images, *PAMI*, 23(5):501-518, 2001.

[2] Y. Hsieh, "SiteCity: A Semi-Automated Site Modeling System," *CVPR*, 499-506, 1996.

[3] A. Gruen and R. Nevatia (Editors), Special Issue on Automatic Building Extraction from Aerial Images, *CVIU*, November 1998.

[4] P. E. Debevec, C. J. Taylor and J. Malik, Modeling and rendering architecture from photographs: A hybrid geometry- and image-based approach, *SIGGRAPH*, 11-20, 1996.

[5] S. Coorg and S. Teller, Extracting textured vertical facades from controlled close-range imagery, *CVPR*, 625-632, 1999.

[6] F. Taillandier, Texture and Relief Estimation from Multiple Georeferenced Images, *MS Thesis*, DEA Algorithmique, Ecole Polytechnique, 2000.

[7] T. Werner and A. Zisserman, New Techniques for Automated Architecture Reconstruction from Photographs, *ECCV*, 2002.

[8] R. Cipolla, T. Drummond and D.P. Robertson, Camera calibration from vanishing points in images of architectural scenes, *BMVC*, 2:382-391, 1999.

[9] D. P. Robertson and R. Cipolla, Building Architectural Models from Many Views Using Map Constraints. *ECCV*, 2:155-169, 2002.

[10] A. Dick, P.Torr, S. Ruffle, and R. Cipolla. Combining Single View Recognition and Multiple View Stereo for Architectural Scenes, *ICCV*, 2001.

[11] A. R. Dick, Philip H. S. Torr, and R. Cipolla, A Bayesian Estimation of Building Shape Using MCMC. *ECCV*, 2:852-866, 2002.

[12] I. Stamos and P. K. Allen, Automatic Registration of 2-D with 3-D Imagery in Urban Environments, *ICCV*, 2:731-736, 2001.

[13] C. Fruh and A. Zakhor, 3D Model Generation for Cities Using Aerial Photographs and Ground Level Laser Scans, *CVPR*, 2001.

[14] S. C. Lee, A. Huertas, and R. Nevatia, Modeling 3-D complex buildings with user assistance, *WACV*, 170-177, 2000.

[15] B. Caprile and V. Torre, Using vanishing points for camera calibration, *IJCV*, 127-140, 1990.

[16] S. C. Lee, S. K. Jung, and R. Nevatia, Integrating Ground and Aerial Views for Urban Site Modeling, *ICPR,* 4:107-112, 2002.

[17] S. C. Lee, S. K. Jung, and R. Nevatia, Automatic Integration of Facade Textures into 3D Building Models with a Projective Geometry Based Line Clustering, *Computer Graphics Forum (EuroGraphics)*, 21(3):511-519, 2002.

[18] R. Hartley and A. Zisserman, Multiple View Geometry in Computer Vision, *Cambridge University Press*, 2000.