

# User Assisted Modeling of Buildings

J. Li, R. Nevatia and S.Nornoha

Institute for Robotics and Intelligent Systems  
University of Southern California  
Los Angeles, California 90089  
<http://iris.usc.edu>

## Abstract

An approach that allows a user to assist an automatic system in modeling buildings is described. The approach is designed to be efficient in user time and effort while preserving the quality of the models created. Currently our system is able to handle the rectangular buildings with flat roof or symmetric gabled roof. Models can be created by only one or two clicks in many cases. Efficient editing of automatically derived models is also possible.

## 1 Introduction and Overview

Significant progress has been made in recent years in the goal of extracting models of buildings from aerial images by completely automatic systems [Strat et al., 1992] [Jaynes et al., 1997] [Nornoha and Nevatia, 1997] [Nevatia and Huertas, 1998]. Still, the results are not as good as models that can be generated by a human user. The completely manual systems, however, require an acceptable amount of effort from a human modeler, both in terms of time and cost. We describe an approach that attempts to provide user assists to an automatic system in a way that the user effort is diminished significantly while the quality of the results is still preserved.

Several approaches to user assisted modeling are possible. The conventional approach is to provide a set of generic models which are then fit to the image data by changing model and viewing parameters as in the basic CME system [Strat et al., 1992]. In this approach, the system provides geometric computations but substantial time and effort are required from the user. Newer approaches have attempted to combine user input with varying amounts of auto-

matic processing [Hsieh, 1996] [Heuel and Nevatia, 1996]. As many urban areas contain buildings that are identical or very similar to others, tools for replicating them can also increase the user productivity as in [Hsieh, 1996].

In our approach, basic modeling tasks in our system are still performed by an automatic system but this system receives simple, but critical, assists from a user. The assisted system's capabilities are limited by those of the underlying system. In our case, the shapes of the buildings are restricted to be rectilinear; the roofs may be either flat or symmetric gables.

The underlying automatic system is the USC multi-view building detecting system described in [Nornoha and Nevatia, 1997] [Nevatia and Huertas, 1998]. The basic steps of this system consist of forming parallelogram hypotheses (to represent rectangular parts of roofs) and inferring 3-D shapes from them. 3-D analysis also provides verification of the hypotheses by determining if sufficient evidence for their presence exists. A user can assist this system in the process of hypothesis formation as well as in making corrections to the resulting 3-D models.

A user interaction typically consists of the user pointing to a point or line feature; the pointing need not be precise as precise features are automatically selected by the system. We will call one such interaction a "click". The system requires two (or more) views of a scene with associated camera geometry; however, all user interactions take place in one view only. Other views can be displayed but the user is not asked to view the images stereoscopically. We believe that confining interactions to one view can significantly reduce the effort required by the user.

The following system description is divided in two components. In the first situation a building has not been detected and needs to be added. In the second,

---

\* This research was supported in part by the Defense Advanced Research Projects Agency under contract DACA76-97-K-0001 and monitored by the Topographic Engineering Center of the U.S. Army.

a building has been detected partially and requires editing.

## 2 Adding a building

User assistance in adding a new building to the model consists of approximately indicating some numbers of corners of its roofs. The automatic system constructs and displays a 3-D model after each click (though in some cases, no models may result). Each subsequent click refines the models. For each rectangular component, up to three clicks may be required for flat roofed buildings and up to four clicks for gabled roofs. The resulting models can then also be edited by the methods described in section 3.

After each click, indicating the approximate position of a corner, the system finds nearby corners constructed from extracted line segments. The sides of a single corner can suffice to trigger some parallelogram hypotheses. Multiple hypotheses are possible at each step; selection among them is made by searching for matching evidence in all the available views. The matching process also creates 3-D models from the initial 2-D roof hypotheses.

The operational and computational processes for the flat and the gabled roofs are quite similar, however, for simplicity, we describe each of them separately below.

### 2.1 Flat roof buildings

A maximum of three corner clicks is required to specify a rectangular flat roof component.

Figure 1 depicts the situation after the first user click. The system locates all junctions near the click and reports failure if none is found. For each junction found, the system attempts to construct a parallelogram. The parallelogram is formed by first examining the stored information and looking for a U-structure that uses the junction legs. If no U-structure is available, the junction legs are used to derive the parallelogram (roof hypothesis). The elements of the parallelogram are matched to elements on the other views and scores are computed as the system would during automatic operation. The system then selects one configuration and presents it to the user.

Figure 2 illustrates the situation after the second click. The second click is used to generate a new hypotheses in the same manner as with the first click. The hypotheses are formed that include the point from the first click however, are weighted higher.

After the third click, three points are used to form three possible parallelograms to represent roof hy-

potheses, as shown in Figure 3. The system calculates the 3D orientation of these planes, matches the elements with elements on other views. For all possible matches, select those hypotheses with least inclination for a flat roofed building. Also the angles between the sides must be close to 90 degrees in 3D. The system computes scores as before, and selects the hypothesis with the best score.

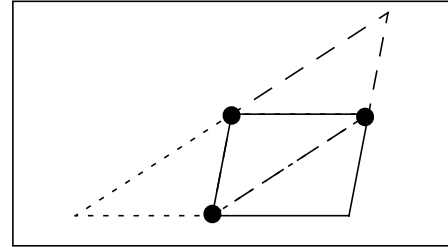


Figure 3. Three parallelograms can be formed from three points.

We show some examples of user interaction using portions of images from the McKenna MOUT site at Fort Benning. Figure 4 shows two examples where a single click was sufficient to recover each of the two buildings, with no further editing required.

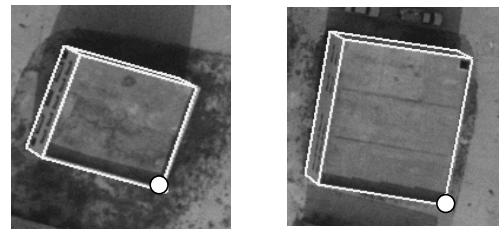


Figure 4. Addition of these buildings required one click each.

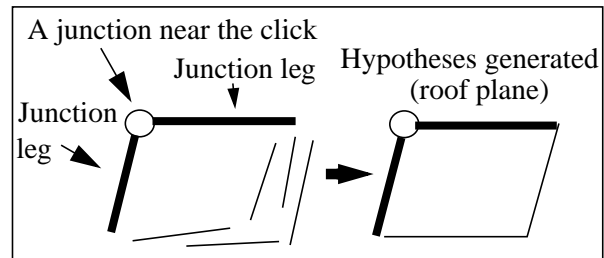


Figure 1. First input analysis.

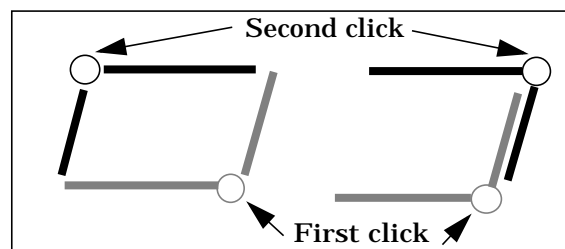


Figure 2. Two possible configurations for a second click.

Figure 5 shows an example where three clicks are needed. The first click results in a partial hypothesis. The second click gives a better hypothesis but is still not completely accurate. The third click results in an accurate model that requires no further editing. For each of these examples, the time required to construct a model is less than 4 seconds.

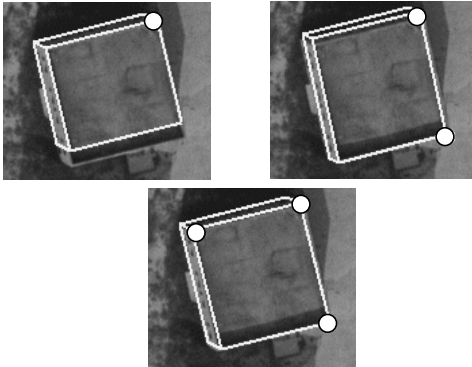


Figure 5. Addition of these buildings where three clicks are needed.

## 2.2 Gabled roof buildings

At least two corner clicks but not more than four corner clicks are required to generate a rectangular symmetric gabled roof component.

First, the user needs to click on or near two corners on roof spine of the desired building. The system locates all the junctions near these two clicks. If none is found for either corner, the system reports failure and prompts the user to give more clicks. Otherwise, for each junction found, the system tries to find all slanted side edges which make right angle with the spine edge. Among all these slanted side edges, the system evaluates their goodness and picks a best one. Thus we can obtain a derived junction for one side of the roof. The situation after first two clicks is shown in Figure 6.

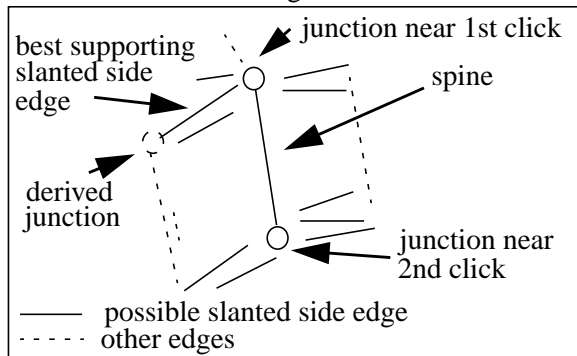


Figure 6. The situation after first two clicks.

If the above derived junction can not be obtained or is inaccurate, the user may click on or near a corner of the roof side. With these three junctions obtained

either by clicks or from computations, the system then tries to locate the symmetric corner of the current obtained side corner. Because the side height is not known yet, the system assumes a medium height for this corner and then projects it from the image coordinate to the world coordinate. Using symmetry property, the system is able to acquire the 3D coordinate of the reflected corner. Then the system projects it back to the image coordinate, locates all possible corners around, and chooses a best one as the other derived junction. Figure 7 illustrates the use of symmetry property in obtaining the other derived junction.

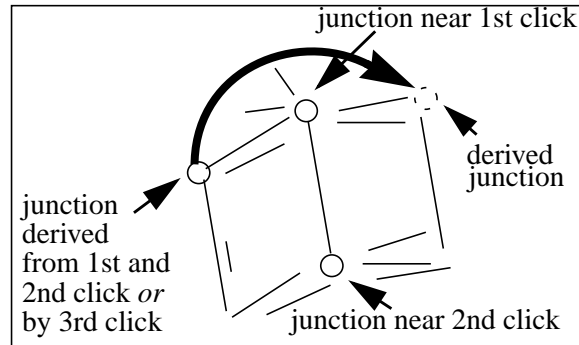


Figure 7. Use symmetry property to try to compute the symmetric junction.

If the other derived junction is inaccurate, the user needs to provide a fourth click. With all these four junctions obtained either by clicks or from computations, the shape of the gabled roof is determined and the corresponding 2D roof hypothesis can be generated, as shown in Figure 8.

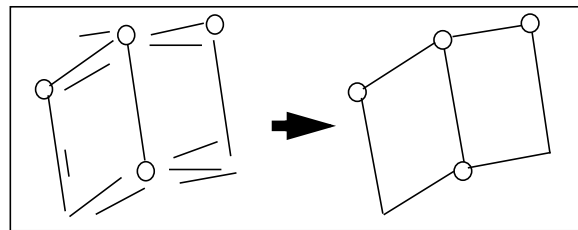


Figure 8. 2D Gabled roof hypothesis formed.

In order to form the 3D building hypothesis, we need to compute the side height and spine height of the anticipated gabled building. This is done by matching the 2D hypothesis in one view to another view and collecting the supporting matching evidence. A pair of side height and spine height which score best are used to convert 2D hypothesis into 3D hypothesis. To improve the speed of height computation, two-level (coarse level and fine level) matching analysis is adopted.

We show some examples of user interaction using portions of images from the McKenna MOUT site.

Figure 9 shows two examples where two clicks are sufficient to detect symmetric gabled buildings, with no further editing required.

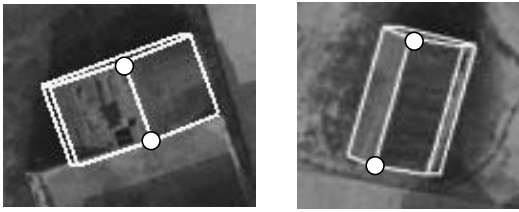


Figure 9. Addition of gabled buildings with only 2 clicks.

Figure 10 shows an example where a third click is needed to refine the hypothesis from the previous two clicks. The third click generates the accurate model that requires no further editing.

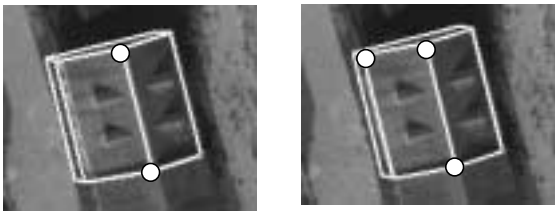


Figure 10. Addition of a gabled building where the third click is needed to refine the hypothesis from 2 clicks.

Figure 11 gives an example where four clicks are necessary. The system finds no hypothesis with first two clicks. The third click results in an inaccurate hypothesis. Thus the last click is essential to form the accurate hypothesis. For each of above gabled buildings, the time to construct a complete model is less than 6 seconds.

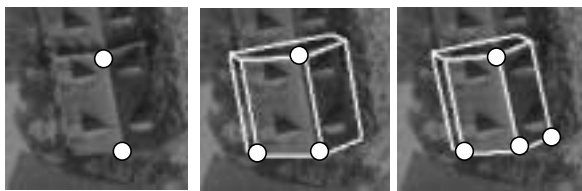


Figure 11. Addition of a gabled building where four clicks are needed.

### 3 Editing a Building

This process can be used to edit a current building hypothesis, either derived automatically or by interactions in earlier stages as described above. Three available corrections are to adjust a corner, to adjust a side or to adjust height. If more complex interactions are needed, the building should be deleted and reconstructed. Again, interactions take place in only

one view and only approximate locations are needed. The user interactions are as follows:

- To adjust a corner, click on a new location for it.
- To adjust a roof side, click anywhere along the actual roof side.
- To adjust height, click anywhere along the base of the building.

For each of the steps above, the system recomputes all aspects associated with the formation of 3D hypothesis during automated operation. The actions of the automatic system for editing a building are similar to those for adding a buildings. If a corner is indicated by the user, the system finds the nearest corner in the existing hypotheses and replaces it by a corner near the indicated position. A new hypotheses is generated and its height recomputed. If a side is indicated, the closest side of the existing hypothesis is found and moved to include the indicated position. Again, a new hypothesis and building model is constructed.

The height correction is a little more complex. After the user clicks anywhere along the base of the building, the system first calculates the correct ground point, which is the intersection of the new base line with one of the projected walls. The wrong height indicates that the matching linear evidence in another view used by the system in calculating the height automatically can not be relied on. As illustrated in Figure 12, the roof corner is correct in 2D, but incorrect in 3D due to its wrong z-coordinate. The system adopts the binary search algorithm to search for the height with required accuracy. During each iteration, the system takes the middle height in current height range and uses it to calculate the 3D coordinate of the roof corner and projects it back to 2D. The error between the projected 2D value and the correct 2D value is used to reduce the height range.

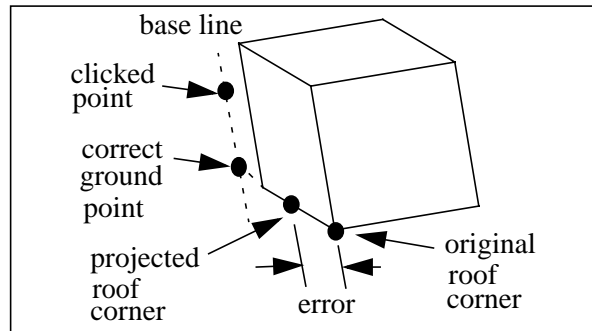


Figure 12. Illustration of height correction.

These processes are illustrated by examples below. Figure 13 shows a detected building that is only partially correct. The appropriate correction consists of indicating a point along the actual building boundary to cause the system to adjust the incorrect side. As before, the system automatically recalculates the height and location of the new model.

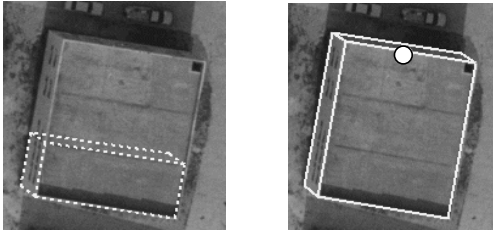


Figure 13. Adjusting wrong side of a building

The last example, shown in Figure 14 illustrates a similar procedure to adjust the height of the building. The user needs only to select a point along the base of the building.

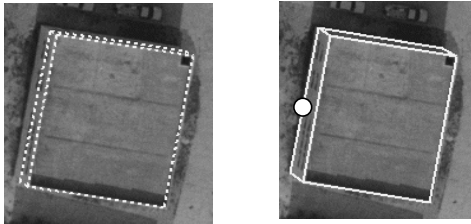


Figure 14. Adjusting wrong height of a building.

## 4 Results and Conclusions

This system has been applied to a number of images from the Ft. Hood and the Ft. Benning sites. Results on the McKenna MOUT site in Benning are given in our companion paper [Nevatia and Huer-tas, 1998]. We give results on some examples chosen from Ft. Hood here, as shown in Figures 15 and 16. Table 1 summarizes statistical information on the number of clicks and time needed to construct these models. .

Table 1 Distribution of Interactions for Three Areas from Ft. Hood Site

| Roof type                                 | Clicks needed | Components Formed |
|---|---------------|-------------------|
| Flat Roof Buildings                       | 1             | 1                 |
|   | 2             | 1                 |
|   | 3             | 14                |
| Gabled Roof Buildings                     | 2             | 5                 |
|   | 3             | 8                 |
|   | 4             | 16                |
| TOTAL                                     |               | 45                |
| Buildings requiring height correction = 2 |               |                   |
| Total elapsed wall time = 290 seconds     |               |                   |

The above results show that the described system can be a very effective tool for constructing models for the classes of buildings within its capability. We intend to extend this system to include more complex, rectilinear shapes.



Figure 15 Results from Area 1 of the Fort Hood site.



(a) Area 2 in Ft. Hood Site



(b) Area 3 in Ft. Hood Site

Figure 16. Results from Two Areas in the Ft. Hood Site.

## References

- [Heuel and Nevatia, 1996] S. Heuel and R. Nevatia, "Including Interaction in an Automated Modeling System", *Proceedings of Image Understanding Workshop*, Palm Springs, CA, February 1996, pp. 429-434.
- [Hsieh, 1996] Y. Hsieh, "SiteCity: A Semi-Automated Site Modeling System", *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, San Francisco, California, June 18-20 1996, pages 499-506
- [Jaynes et al., 1997] C. Jaynes, M. Marengoni, A. Hanson, E. Riseman and H. Schultz, "Knowledge Directed Reconstruction from Multiple Aerial Images", *Proceedings of the DARPA Image Understanding Workshop*, New Orleans, LA, May 1997, pp 971-976.
- [McKeown et al., 1997] D.M. McKeown, S.D. Cochran, S.J. Gifford, W.A. Harvey, J.C. McGlone, M.F. Polis, S.J. Ford, and J.A. Shufelt, "Research in the Automated Analysis of Remotely Sensed Imagery: 1995:1996", *Proceedings DARPA Image Understanding Workshop*, New Orleans, LA, May 1997, pp. 779-812
- [Nevatia and Huertas, 1998] R. Nevatia and A. Huertas, "Knowledge-Based Building Detection and Description: 1997-1998", *Proceedings of the Image Understanding Workshop*, Monterey, CA, November 1998 (these proceedings).
- [Noronha and Nevatia, 1997] S. Noronha and R. Nevatia, "Detection and Description of Buildings from Multiple Aerial Images", *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, San Juan, PR, June 1997, pp. 588-594
- [Strat et al., 1992] T. Strat, L. Quam, J. Mundy, R. Welty, W. Bremner, M. Horwedel, D. Hackett, and A. Hoogs, "The RADIUS Common Development Environment", *Proceedings of the 1992 DARPA Image Understanding Workshop*, San Diego, California, pp. 215-226