

The 3D MOSAIC Scene Understanding System

Martin Herman

Takeo Kanade

Shigeru Kuroe

Computer Science Department

Carnegie-Mellon University

Pittsburgh, PA 15213

Abstract

A scene understanding system derives and represents information about a given scene for the purpose of various given tasks. The 3D Mosaic system incrementally derives a 3D description of a complex urban scene from multiple images. The description, which we call a *scene model*, is intended to be useful for tasks such as matching, display generation, planning paths through the scene, and making other decisions dealing with the scene environment. This paper briefly describes the system and some experiments in acquiring and using the scene model. Further details may be found in [6, 7].

1. Introduction

We are currently working with aerial photographs of Washington, D.C. (Fig. 1). Our approach of incrementally acquiring the model arises from three observations: First, single images contain only partial information about a scene. Second, errors and inconsistencies often result when interpreting complex images. Third, features of visible surfaces (such as scene edges and texture) are often more apparent (and thus easier to extract) in one image than another because of differences in viewpoint and lighting conditions.

In our approach, information derived from a sequence of images obtained from multiple viewpoints is incrementally accumulated to construct the model. A partial 3D description of the scene is derived from each view. The initial model, constructed from the 3D information obtained from the first view, represents an initial approximation of the scene. As each successive view is processed, the model is incrementally updated and gradually becomes more accurate and complete.

At any point along its development, the model represents the current understanding of the scene and may be used for various tasks, including the following two that are important in the incremental acquisition process itself: (1) 3D information derived from a new view must be matched to the model so that updating can occur, (2) higher-level components should be able to use the model to determine the parts of the scene to be analyzed in more detail.

In our approach, 3D scene features that are relatively inexpensive to obtain, such as certain corners of buildings, are extracted from the images using stereo analysis. The scene model is a surface-based description which is hypothesized from these features by utilizing task-specific knowledge (i.e., block-shaped objects in an urban scene). Examples of models obtained are shown in Figs. 4, 5, and 7. Some applications for these types of models are:

1. Model-based photointerpretation. The model can aid in interpreting images of the scene obtained from arbitrary viewpoints [3, 10].
2. 3D change detection. Rather than detecting changes in image features, which is the conventional approach, scene models can be obtained and compared to detect changes in scene features.
3. Simulating the appearance of the scene. Although the model may be obtained from only a few views, arbitrary views may be generated from it. This is useful for familiarizing personnel with a given area and for flight planning by generating the scene appearance along hypothetical flight paths.
4. Robot navigation. Approximate descriptions of complex environments may be used to make decisions dealing with path planning or determining which parts of the environment to analyze in more detail.

2. Stereo Analysis

We currently extract 3D features from the images by means of stereo analysis, which is based on matching junctions extracted from the two images. There are several reasons why we use this type of stereo matching. First, by extracting 3D information dealing with scene vertices and edges emanating from them, we obtain portions of boundaries of scene buildings, particularly building corners. These boundaries are then used to construct 3D approximations of the buildings. (For a different approach, see [5].)

Second, because of our wide-angle stereo images, there are large disparity jumps and large portions of the scene are visible in

one image but not the other. Because most stereo systems do not distinguish these from other regions of the image, they try to find matches for them and therefore have trouble [1, 2, 4, 8]. In our approach, rather than attempting to find matches for scene faces occluded in one of the images, we match face boundaries visible in both images. We do this by explicitly taking into account the way junction appearances change from one image to the other, using the knowledge that in urban scenes, roofs of buildings tend to be parallel to the ground plane, while walls tend to be perpendicular to this plane. Edges in the scene perpendicular to the ground will appear in each image to be directed towards the vertical vanishing point. If an edge lies on a roof, its orientation in the other image as a function of position along the epipolar line can be predicted if the normal to the ground plane is known.

Therefore, when a (say) FORK junction is found in one image, the leg of the junction directed towards the vertical vanishing point is initially assumed to arise from a scene edge perpendicular to the ground, while the other two legs are initially assumed to arise from scene edges lying on a roof or on the ground. Its appearance in the other image can therefore be predicted.

Structural relationships between scene vertices are also used to aid in the matching, using the knowledge that if two junctions in an image arise from scene vertices at the same height above the ground, the positions of the corresponding junctions in the other image, as a function of position along the epipolar line, can be predicted. We make the assumption that junctions close to one another in the image often correspond to vertices lying on top of the same building and therefore have approximately the same height. In this way, the configurations within the neighborhoods around junctions in the two images are used in the matching.

A final reason for using junctions to perform the stereo matching is that feature-based matching results in more accurate 3D positions for occlusion boundaries than gray scale area matching.

Our stereo analysis involves first extracting linear segments from each image [9], and then forming junctions from the segments. Next, potential matches for each junction are searched by predicting its appearance in the other image. To arrive at a single match for each junction, the space of potential matches is searched with a beam search [10] using two main heuristics: First, relative junction positions in the two images should agree with predicted positions (as described above). Second, image intensities inside matching junctions should be similar since the junctions arise from

the same face corners. Finally, 3D coordinates are derived using triangulation. Fig. 2 shows a perspective view of the 3D vertices and edges, called a wire-frame description of the scene, that result after processing the stereo pair in Fig. 1.

3. The 3D Scene Model

The following factors have determined how the scene model is represented and manipulated.

1. Partially complete, planar-faced objects must be efficiently described by the model. It is therefore represented as a graph in terms of symbolic primitives such as faces, edges, vertices, and their topology and geometry. Information is added and deleted by means of these primitives.
2. The model must be easy to use in matching. This is facilitated by representing it compactly in terms of symbolic primitives rather than, say, surface points.
3. Because scene approximations are often more useful if they contain reasonable hypotheses for parts of the scene for which there is little data, we introduce mechanisms that permit hypotheses to be generated, added, and deleted. These mechanisms are aided by the compact, symbolic model representation. Primitive elements in the model are tagged as *confirmed* if they are derived directly from images, or *unconfirmed* if they are hypothesized.
4. Because incremental modifications to the model must be easy to perform, we introduce mechanisms to (a) add primitives to the model in a manner such that constraints on geometry imposed by these additions are propagated throughout the model, and (b) modify and delete primitives if discrepancies arise between newly derived and current information.

4. Constructing and Updating the 3D Scene Model

4.1. Single View

This section describes the construction of an initial scene model from 3D wire-frame data (Fig. 2) extracted from a single (stereo pair) view (Fig. 1). The processing proceeds as follows.

First, if two wire-frame edges are nearly parallel and very close (e.g., E1 and E2 in Fig. 2), they are merged into a single edge. Next, since each vertex is assumed to correspond to a corner of an object, each adjacent pair of legs ordered around the vertex is assumed to correspond to the corner of a planar face. Partial faces, called *web faces*, are generated for such pairs (Fig. 3a).

After all web faces have been created, those that represent the corners of a single face are merged. Web faces to be merged may be either touching (e.g., Fig. 3b and F1 and F2 in Fig. 2) or non-touching (e.g., Fig. 3c and F3 and F4 in Fig. 2). After all mergers, many faces may still be incomplete because they may not have closed boundaries. These are completed by first hypothesizing their shapes, either as parallelograms (for faces consisting of single corners as in Fig. 3d) or as other closed polygons (Fig. 3e), and then generating appropriate edges and vertices. After all faces have been completed, those that seem to be holes in other faces are converted to holes (Fig. 3f).

At this point, many objects may be incomplete because they may not be closed. Since we are dealing with urban scenes, faces that lie high enough above the ground are assumed to represent roofs of buildings. A hypothesized vertical wall is dropped towards the ground from each edge of such faces, unless the edge is already part of another face (Fig. 3g). Each wall is dropped either to the ground or to the first face it intersects on the way down. The resulting scene model is displayed in Fig. 4. In order to render more realistic displays, gray scale obtained from one of the original images (Fig. 1) is added to them (Fig. 5).

Note that this model is a higher level description than a depth map, but is obtained without first generating a depth map. The depth map, which is the result of many stereo analysis systems [1, 4, 8], is like an image in that it is an array of numbers that requires description. Our **3D Mosaic** model, on the other hand, is symbolic and compact, making it more useful for matching, incremental updating, and other tasks.

4.2. Multiple Views

This section describes the process by which wire-frame information derived from a second view (Fig. 6) is incorporated into the scene model of Fig. 4. This new set of wire frames was manually generated to simulate the information available from a point of view opposite to that of Fig. 1 (i.e., viewing the scene from the "back"). The primary difference between the procedure used here and the one described in the previous section is that the wire-frame data is matched with and merged into the current model. The matching step provides the coordinate transformation from the data to the model and corresponding confirmed edges and vertices in the two. We assume in this experiment that the coordinate transformation is known; the data and model may therefore be described in the same coordinate system. To determine corresponding edges and

vertices, we check absolute distances and whether corresponding elements are connected in the same configuration in both the model and data.

Two objects match if they have confirmed elements that match, and are merged by (1) merging their corresponding pairs of vertices and edges into single elements by "weighted averaging" of their positions, and (2) deleting hypothesized elements in the model that are inconsistent with newly obtained or changed elements. To determine whether inconsistencies exist, dependencies are recorded for each hypothesis at the time of its creation. A hypothesis is dependent on all elements whose existence directly resulted in the creation of the hypothesis. If one of these elements is modified or deleted, the hypothesis must also be deleted since the conditions under which it was created are no longer valid. After all mergings and deletions, the remaining edges and vertices in the wire-frame object are added to the model object.

At this point, many objects are incomplete because new wire-frame data has been added and many hypothesized elements have been deleted. These objects are completed using the techniques described in the previous section. Fig. 7 shows the resulting updated scene model which, for easier comparison by the reader, is displayed for the same viewpoint as the initial model in Fig. 4.

Note that the updated model contains more buildings since new wire-frame data, some of which represent new buildings, have been incorporated into the initial model. Also note that for many buildings described in both versions of the model, the positions of vertices and edges are more accurate in the updated version. This is because many hypothesized vertices and edges are replaced by accurate ones obtained from the new data, and many confirmed vertices and edges are merged with corresponding ones in the data by "averaging" their positions, generally decreasing the amount of error. This experiment demonstrates how information provided by each additional view allows the model to be incrementally made more complete and accurate.

5. Summary

The **3D Mosaic** system incrementally derives a 3D model of a complex urban scene from multiple images. The model plays the role of a central description which incrementally accumulates scene information and can be used, at any point along its development, for performing various tasks dealing with the scene environment.

We have demonstrated that task-specific knowledge is very useful for interpreting complex images. Such knowledge is used both for the stereo analysis and for reconstructing shapes from the stereo output. The task-specific knowledge deals with block-shaped objects in an urban scene. Surfaces are assumed to be planar, oriented either vertically or horizontally in the scene, and shaped as parallelograms if there is no evidence to the contrary. Also, the position of the ground plane is assumed to be known. Heuristics for generating 3D objects from the stereo output have been developed from these assumptions.

References

1. Baker, H. H., and Binford, T. O. "Depth from Edge and Intensity Based Stereo." *Proc. IJCAI-81* (1981).
2. Barnard, S. T. and Fischler, M. A. "Computational Stereo." *Computing Surveys* 14, 4 (December 1982).
3. Barrow, H. G., Bolles, R. C., Garvey, T. D., Kremers, J. H., Tenenbaum, J. M., and Wolf, H. C. "Experiments in Map-guided Photo Interpretation." *Proc. IJCAI-77* (August 1977).
4. Grimson, W.E.L. "A Computer Implementation of a Theory of Human Stereo Vision." *Phil. Trans. Roy. Soc. Lond. B292* (1980), 217-253.
5. Henderson, R. L., Miller, W. J., and Grosch, C. B. "Automatic Stereo Reconstruction of Man-made Targets." *Proc. SPIE 186* (August 1979).
6. Herman, M., Kanade, T., and Kuroe, S. "Incremental Acquisition of a Three-Dimensional Scene Model from Images." *Proc. ARPA Image Understanding Workshop* (September 1982).
7. Herman, M. and Kanade, T. Incremental Acquisition of a Three-Dimensional Scene Description from Complex Images. Computer Science Department, Carnegie-Mellon University, in preparation.
8. Lucas, B. D., and Kanade, T. "An Iterative Image Registration Technique With an Application to Stereo Vision." *Proc. IJCAI-81* (August 1981).
9. Nevatia, R. and Babu, K.R. "Linear Feature Extraction and Description." *Computer Graphics and Image Processing* 13 (July 1980), 257-269.
10. Rubin, S. *The ARGOS Image Understanding System*. Ph.D. Th., Carnegie-Mellon University, 1978.

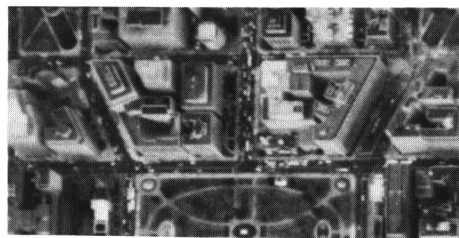
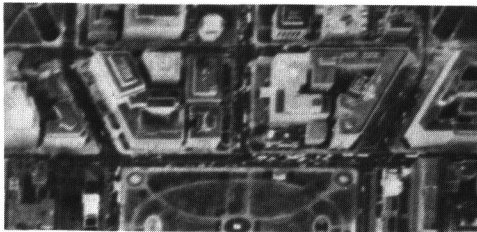


Figure 1: A stereo pair of aerial images.

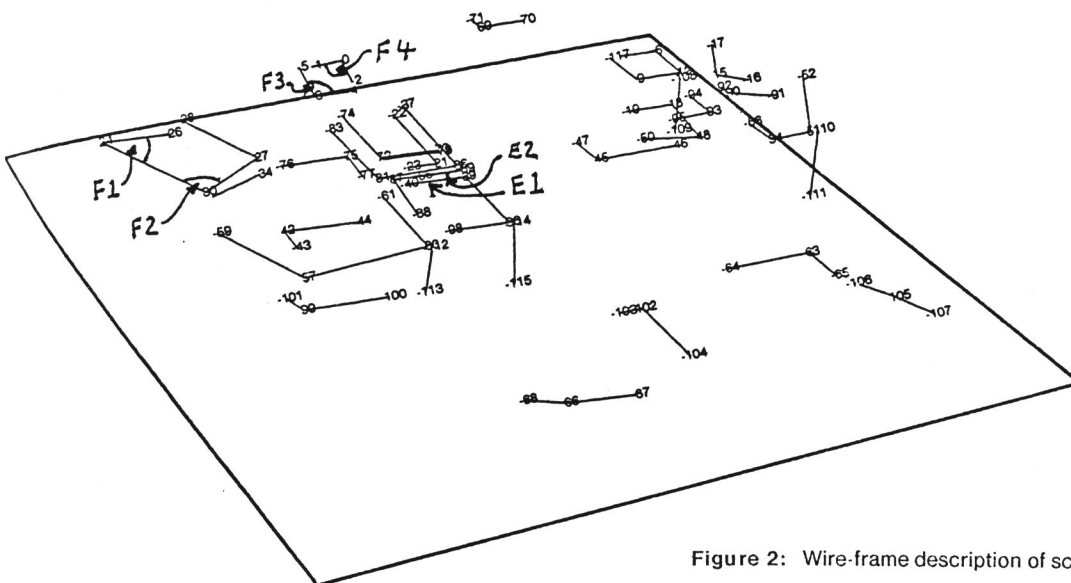


Figure 2: Wire-frame description of scene in Fig. 1.

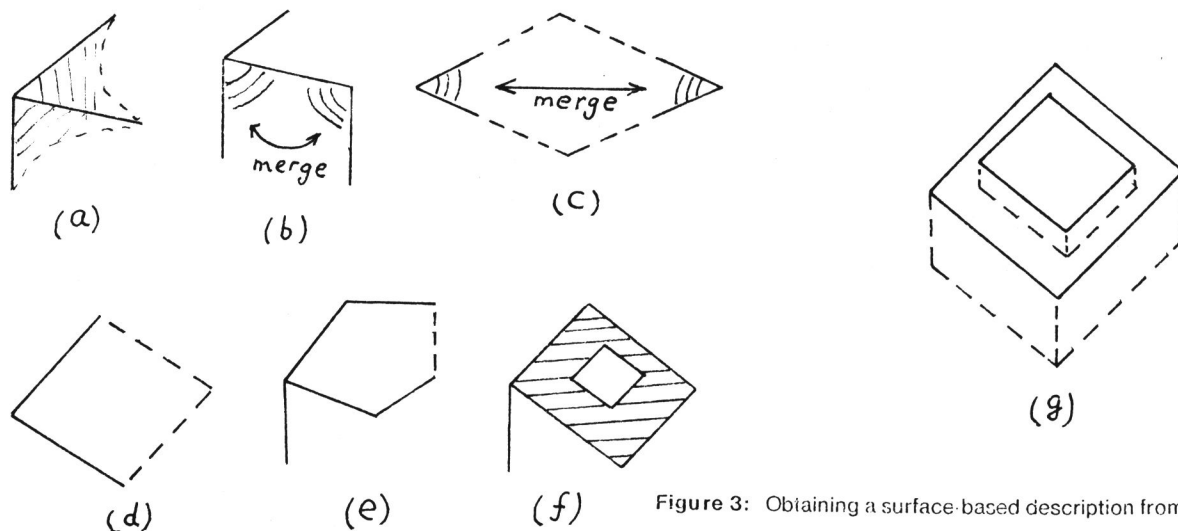


Figure 3: Obtaining a surface-based description from wire frames.

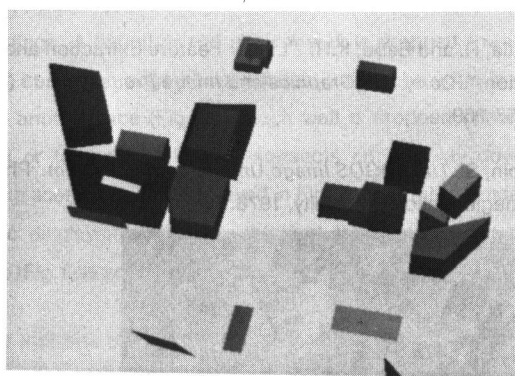


Figure 4: Reconstructed buildings.

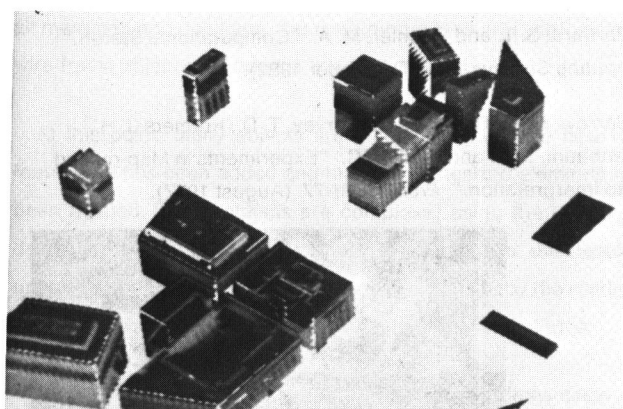


Figure 5: Reconstructed buildings displayed with gray scale.

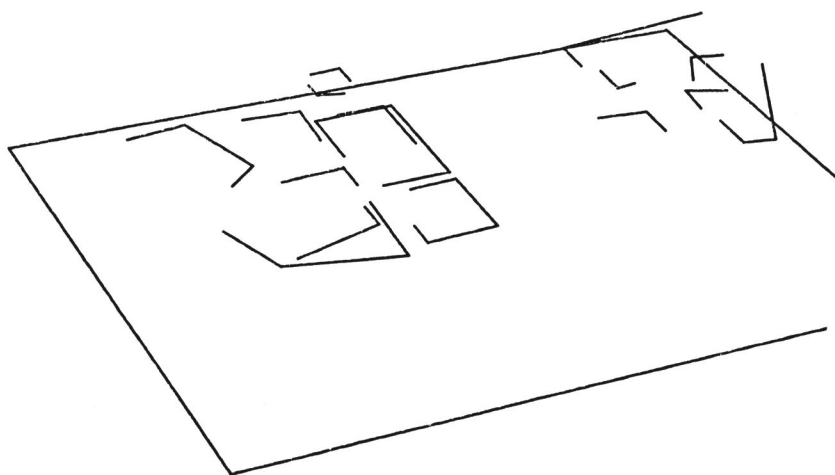


Figure 6: Manually generated wire-frame description.

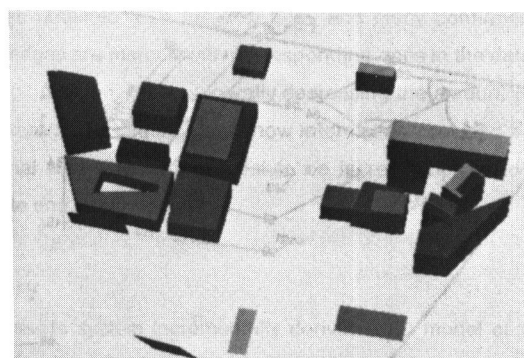


Figure 7: Updated version of scene model in Fig. 4.