A system for automated site model acquisition*

Robert T. Collins, Chris Jaynes, Frank Stolle, Xiaoguang Wang,
Yong-Qing Cheng, Allen R. Hanson, Edward M. Riseman

Department of Computer Science
Lederle Graduate Research Center
University of Massachusetts
Amherst, MA. 01003-4610

ABSTRACT

A system has been developed to acquire, extend and refine 3D geometric site models from aerial imagery. The system hypothesizes potential building roofs in an image, automatically locates supporting geometric evidence in other images, and determines the precise shape and position of the new buildings via multi-image triangulation. Projectively warped image intensity maps are associated with the faces of each recovered building, allowing realistic rendering of the scene from new viewpoints.

Keywords: aerial image understanding, building extraction, 3D site modeling

1 INTRODUCTION

Acquisition of 3D geometric site models from aerial imagery is currently the subject of an intense research effort in the U.S., sparked in part by the ARPA/ORD RADIUS project.[7,8,5,13] We have developed a set of image understanding modules to acquire, extend and refine 3D volumetric building models. The system design emphasizes model-directed processing, rigorous camera geometry, and fusion of information across multiple images for increased accuracy and reliability.

Site *model acquisition* involves processing a set of images to detect both man-made and natural features of interest, and to determine their 3D shape and placement in the scene. This paper focuses on algorithms for automatically extracting models of buildings. The site models produced have obvious applications in areas such as surveying, surveillance and automated cartography. For example, acquired site models can be used for automated model-to-image registration of new images,[4] allowing the model to be overlaid on the image to aid visual change detection and verification of expected scene features. Two other important site modeling tasks are *model extension* – updating the geometric site model by adding or removing features,[6] and *model refinement* – iteratively refining the shape and placement of features as more views become available. Model extension and refinement are ongoing processes that are repeated whenever new images become available, each updated model becoming the current site model for the next iteration. Thus, over time, the site model is steadily improved to become more complete and more accurate.

*This work was funded by the RADIUS project under ARPA/Army TEC contract number DACA76-92-C-0041 and by ARPA/TACOM contract DAAE07-91-C-R035.

UMass has designed and implemented a system for automatically extracting building from multiple, overlapping images of a site. To maintain a tractable goal for our research efforts, we have chosen initially to focus on a single generic class of buildings, namely flat-roofed, rectilinear structures. The simplest example of this class is a rectangular box-shape; however other examples include L-shapes, U-shapes, and indeed any arbitrary building shape such that pairs of adjacent roof edges are perpendicular and lie in a horizontal plane. The system is designed to operate over multiple images exhibiting a wide variety of viewing angles and sun conditions. The system is designed to perform well at one end of a data-vs-control complexity spectrum, namely a large amount of data and a simple control structure, versus the alternative of using less data but more complicated processing strategies. In particular, while the system can be applied to a single stereo pair, it generally performs better (in terms of number of buildings found) when more images are used.

Section 2 begins with a specification of general input requirements of the UMass system. This is followed in Section 3 by a breakdown of the system into its key algorithmic components: 1) line segment feature extraction, 2) monocular building rooftop detection, 3) multi-image epipolar rooftop matching, 4) multi-image wireframe triangulation, and 5) projective intensity mapping. This paper concludes with a brief summary and a statement of future work.

2 General system requirements

The UMass building extraction system was developed on a Sun Sparc 10, using the Radius Common Development Environment (RCDE).[12] The RCDE is a combined Lisp/C++ system that supports the development of image understanding algorithms for constructing and using site models. In particular, the RCDE provides a convenient framework for representing and manipulating images, camera models, object models and terrain models, and for keeping track of their various coordinate systems, inter-object relationships, and transformation/projection equations. The RCDE also provides utilities for interactively developing site models, specifying tie points, and for performing photo-resection.

2.1 Images

Acquisition of a 3D site model requires a set of overlapping images of the site. The UMass system is designed to operate over multiple images, typically five or more, exhibiting a wide variety of viewing angles and sun conditions. The number five is chosen arbitrarily to allow one nadir view plus four oblique views from each of four perpendicular directions (e.g. North, South, East and West). This configuration is not a requirement, however. Indeed, some useful portions of the system require only a single image, namely line segment extraction and building rooftop detection. On the other hand, epipolar rooftop matching and wireframe triangulation require, by definition, at least two images, with robustness and accuracy increasing when more views are available. Once again, the number five has been chosen arbitrarily, and perhaps only three well-chosen images would suffice, but verification of this is a matter for further experimentation.

Although best results require the use of many images with overlapping coverage, the system allows considerable freedom in the choice of images to use. Unlike most other building extraction systems, this system does not currently use shadow information, and works best if used on images with different sun angles, or with no strong shadows at all. Also, the term "epipolar" as used here does not imply that images need to be in scan-line epipolar alignment, as required by many traditional stereo techniques. The term is used instead in its general sense as a set of geometric constraints imposed on potentially corresponding image features by the relative orientation of their respective cameras. The relative orientation of any pair of images is computed from the absolute orientation of each individual image (see Section 2.3).

2.2 Site coordinate system

Reconstructed building models are represented in a local site coordinate system that must be defined prior to the reconstruction process. The system assumes this is a "local-vertical" Euclidean Coordinate System, that is, a Cartesian X-Y-Z coordinate system with its origin located within or close-to the site, and the positive Z-axis facing upwards (parallel to gravity). The system can be either right-handed or left-handed. Under a local-vertical coordinate system, the Z values of reconstructed points represent their vertical position or "height" in the scene, and X-Y coordinates represent their horizontal location in the site.

2.3 Camera models

For each image given to the system, the absolute orientation of the camera with respect to the local site coordinate system must be known. This includes both the internal orientation (lens/digitizer parameters) and the external orientation (pose parameters) of the camera. Given the absolute orientation for each image, the system computes all the necessary relative orientation information needed for determining the epipolar geometry between images. Camera models can be specified in two ways. For the **perspective frame camera model,** absolute orientation for each camera is supplied as a 3×4 projective transformation matrix describing (in homogeneous coordinates) how points in the site coordinate system project into points in the image coordinate system. This simple representation makes no distinction between internal and external camera parameters. Translation between "standard" photogrammetric parameterizations (e.g. focal length, principle point coordinates, camera location vector and rotation Euler angles) and the 3×4 matrix representation is provided by the RCDE.

Many aerial photographs, particularly satellite images, are generated by nontraditional imaging systems for which the standard perspective frame camera model is not an adequate description. The **fast block interpolation projection** (FBIP) camera model has been proposed as an alternative description of the imaging process in these situations. The general idea is to break space into "blocks" and then generate local frame camera approximations within each block in such a way that adjacent frame approximations agree at the block boundary, in a manner somewhat analogous to approximating a nonlinear function by a piecewise linear one. This representation easily handles 2D image nonlinearities such as camera lens distortion, as well as 3D space nonlinearities caused by the refraction of light through layers of the atmosphere.

Integrating the FBIP camera model into image understanding algorithms is potentially tricky, since it violates the fundamental assumption underlying most work with traditional, perspective camera models, namely the assumption that straight lines in the world will appear straight in the image. The FBIP camera model not only raises representational concerns such as whether the edge of a building in the image can be adequately characterized by a single straight line segment, but also strikes at a deeper level, invalidating such fundamental geometric notions as vanishing points and epipolar geometry. Our interpretation of FBIP camera model is that it is possible to derive a local 3×4 projective transformation matrix that provides an accurate approximation to the imaging process within a given 3D region of interest spanning the spatial extents of a single building.

2.4 Digital terrain map

Currently, the UMass system explicitly reconstructs only the rooftops of building structures, and relies on vertical extrusion to form a volumetric 3D wireframe model of the whole building. In other words, perpendiculars are dropped from each corner of the reconstructed building rooftop down to the ground, and connected by a building base formed as a vertical translation of a copy of the roof polygon. The extrusion process relies on knowing the local terrain, namely the ground height (Z value) at each location in the scene. We assume this information is represented as an array of elevations, or in the special case of flat ground planes as a horizontal plane equation $Z = z_0$. Representation of digital terrain maps in either format, along with their use in providing a basic ground level for vertical extrusion, is supported by the RCDE. Future versions of the system will use digital terrain maps automatically extracted from stereo image pairs (nadir or oblique) by a correlation-based terrain reconstruction system developed recently at UMass. The technical details of that system also appear in these processings.[14]

2.5 Other required parameters

In addition to the general information described above, a few miscellaneous parameters and thresholds are required to be supplied by the user before the system can be run. The most important of these are:

- **max-building-height** – the maximum possible height of any building that will be included in the site model. This threshold is used to limit the extent of epipolar search regions. The lower this threshold can be, the smaller the search area for rooftop feature matches will be, leading to faster searches with higher likelihood of finding the correct matches.
- **min-building-width** – the minimum horizontal extent (width or length) of any building that will be included in the site model. This is, loosely speaking, a way of specifying the desired "resolution" of the resulting site model, since any buildings having horizontal edges shorter than this threshold will probably not be found. Setting this value to a relatively long length essentially ensures that only large buildings in the site will be modelled.

3 Algorithmic building blocks

The UMass building extraction system currently follows a simple processing strategy. To acquire a new site model, an automated building detector is run on one image to hypothesize potential building rooftops. Supporting evidence is the located in other images via epipolar line segment matching, and the precise 3D shape and location of each building is determined by multi-image triangulation and extrusion. Image intensity information is backprojected onto each face of these polyhedral building models, to facilitate realistic rendering from new views.

This section outlines the key algorithms that together comprise the UMass building extraction system. These algorithms are: line segment extraction, building rooftop detection, epipolar rooftop matching, multi-image wire-frame triangulation, and projective intensity mapping. The description of these algorithms is illustrated with sample results from two sites, the Schenectady County Air National Guard base (Figure 1), and Radius Model Board 1 (Figure 2).

3.1 Line segment extraction

To help bridge the huge representational gap between pixels and site models, a straight line feature extraction routine is applied to produce a set of symbolic line segments, representing geometric image features of potential interest such as building roof edges. We use the Boldt algorithm for extracting line segments.[3] At the heart of the Boldt algorithm is a hierarchical grouping system inspired by the Gestalt laws of perceptual organization. Zero-crossings of the Laplacian of the intensity image provide an initial set of local intensity edges. Hierarchical grouping then proceeds iteratively; at each iteration edge pairs are linked and replaced by a single longer edge if their end points are close and their orientation and contrast (difference in average intensity level across the line) values are similar. Each iteration results in a set of increasingly longer line segments. The final set of line segment features (Figures 3 and 4) can be filtered according to length and contrast values supplied by the user.

Although the Boldt algorithm does not rely on any particular camera model, the utility of extracting straight lines as a relevant representation of image/scene structure is based on the assumption that straight lines in the world (such as building edges) will appear reasonably straight in the image. To the extent that this assumption remains true at the scale of the objects being considered, such as over a region of the image containing a single

Figure 1: Sample subimage from Schenectady dataset.

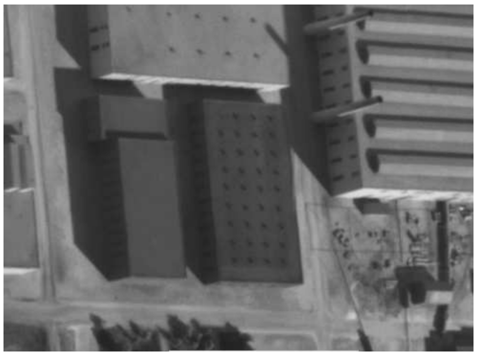

Figure 2: Sample subimage Radius Model Board 1.

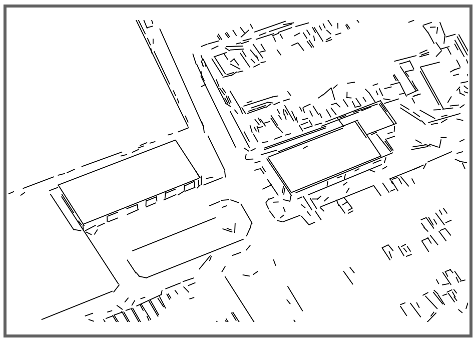

Figure 3: Boldt lines for Figure 1.

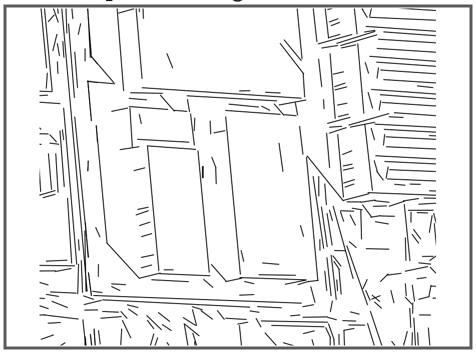

Figure 4: Boldt lines for Figure 2.

building, then straight line extraction remains a viable feature detection method. However, very long lines spanning a significant extent of the image, such as the edges of airport runways, may become fragmented depending on the amount of curvature introduced into the image by nonlinearities in the imaging process.

3.2 Building rooftop detection

The goal of automated building detection is to roughly delineate building boundaries that will later be verified in other images by epipolar feature matching and triangulated to create 3D geometric building models. The UMass building detection algorithm[9] is based on perceptual grouping of line segments into image polygons corresponding to the boundaries of flat, rectilinear rooftops in the scene. Perceptual organization is a powerful method for locating and extracting scene structure. The rooftop extraction algorithm proceeds in three steps; low level feature extraction, collated feature detection, and hypothesis arbitration. Each module generates features that are used at during the next phase and interacts with lower level modules through top-down feature extraction.

Low level features in this system are straight line segments and corners. The domain assumption of flat-roofed rectilinear structures implies that rooftop polygons will be produced by flat horizontal surfaces with orthogonal corners. Orthogonal corners in the world are not necessarily orthogonal in the image, however. To determine a set of relevant corner hypotheses, pairs of line segments with spatially proximate endpoints are grouped together into candidate image corner features. Each potential image corner is then backprojected into a nominal Z-plane in the scene, and that hypothetical *scene corner* is tested for orthogonality.

Mid-level collated features are sequences of perceptually grouped corners and lines that form a chain (Figures 5 and 6). A valid chain group must contain an alternation of corners and lines, and can be of any length. Chains are a generalization of the collated features in earlier work[8] and allow final polygons of arbitrary rectilinear shape to be constructed from low level features. Collated feature chains are represented by paths in a feature relation graph. Low level features (corners and line segments) are nodes in the graph, and perceptual grouping relations between these features are represented by edges in the graph. Nodes have a certainty measure that represents the confidence of the low level feature extraction routines; edges are weighted with the certainty of the grouping that the edge represents. A chain of collated features inherits an accumulated certainty measure from all the nodes and edges along its path.

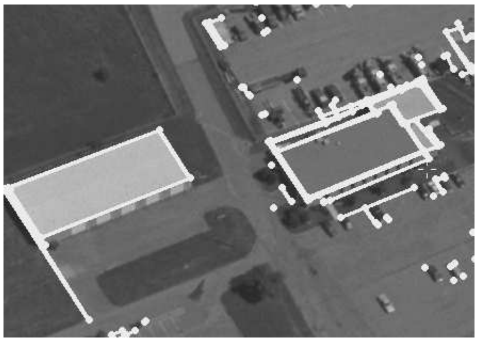

Figure 5: Feature relation graph for Figure 1.

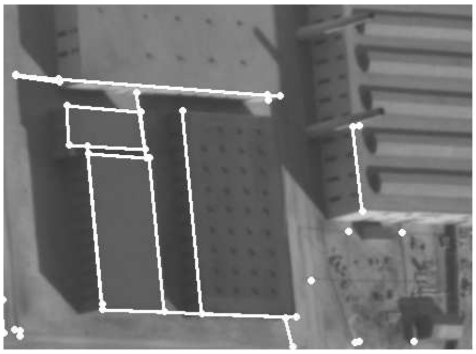

Figure 6: Feature relation graph for Figure 2.

Figure 7: Final rooftop hypotheses for Figure 1.

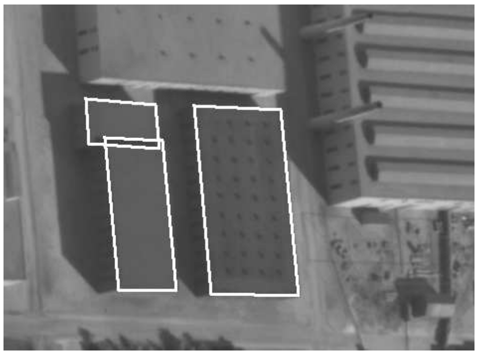

Figure 8: Final rooftop hypotheses for Figure 2.

High level polygon hypothesis extraction proceeds in two steps. First, all possible polygons are computed from the collated features. Then, polygon hypotheses are arbitrated in order to arrive at a final set of non-conflicting, high confidence rooftop polygons (Figures 7 and 8). Polygon hypotheses are simply closed chains, which can be found as cycles in the feature relation graph. All of the cycles in the feature relation graph are searched for in a depth first manner, and stored in a dependency graph where nodes represent complete cycles (rooftop hypotheses). Nodes in the dependency graph contain the certainty of the cycle that the node represents. An edge between two nodes in the dependency graph is created when cycles have low level features in common. The final set of non-overlapping rooftop polygons is the set of nodes in the dependency graph that are both independent (have no edges in common) and are of maximum certainty. Standard graph-theoretic techniques are employed to discover the maximally-weighted set of independent cycles is, which is output by the algorithm as a set of independent high confidence rooftop polygons.

While searching for closed cycles, the collated feature detector may be invoked in order to attempt closure of chains that are missing a particular feature (an example occurs in Figure 6). The system then searches for evidence in the image that such a virtual feature can be hypothesized. In this way, the rooftop detection process does not have to rely on the original set of features that were extracted from the image. Rather, as evidence for a polygon accumulates, tailor-made searches for lower level features can be performed. This type of top-down inquiry increases system robustness.

3.3 Epipolar line segment matching

After detecting a potential rooftop in one image, corroborating geometric evidence is sought in other images (often taken from widely different viewpoints) via epipolar feature matching. The primary difficulty to be overcome during epipolar matching is the resolution of ambiguous potential matches, and this ambiguity is highest when only a single pair of images are used. For example, the epipolar search region for a roof edge match will often contain multiple potentally matching line segments of the appropriate length and orientation, one of which comes from the corresponding roof edge, but the others coming from the base of the building, the shadow edge of the building on the ground, or from roof/base/shadow edges of adjacent buildings (see Figure 9). This situation is exacerbated when the roof edge being searched for happens to be nearly aligned with an epipolar line in the second image. The resolution of this potential ambiguity is the reason that simultaneous processing of multiple images with a variety of viewpoints and sun angles is preferred in the UMass system.

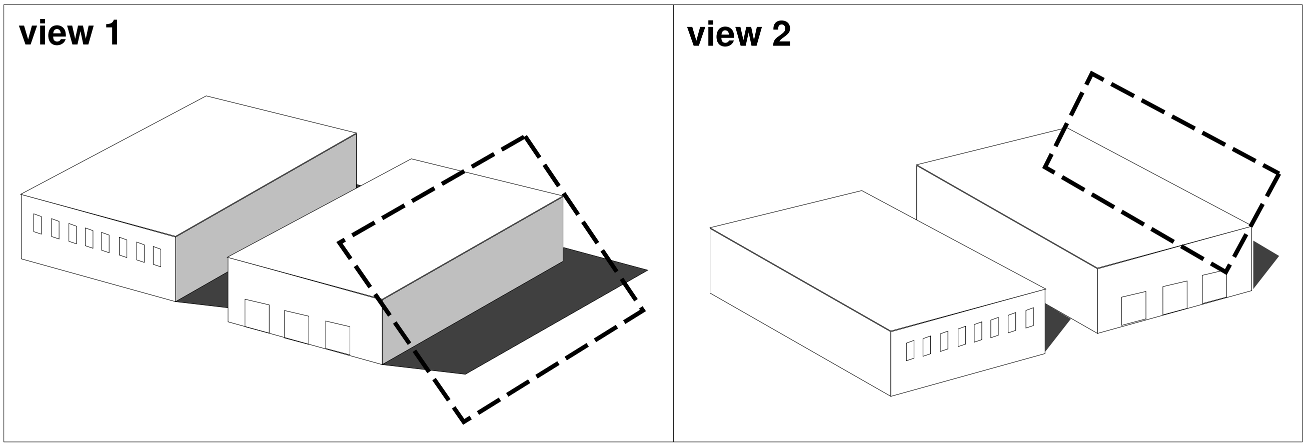

Figure 9: Multiple ambiguous matches can often be resolved by consulting a new view.

We match rooftop polygons by searching for each component line segment separately and then fusing the results. For each polygon segment from one image, an appropriate epipolar search area is formed in each of the other images, based on the known camera geometry and the assumption that the roof is flat. This quadrilateral search area is scanned for possible matching edges, the disparity of each potential match implying a different roof height in the scene. Results from each line search are combined in a 1-dimensional histogram, each potential match voting for a particular roof height. Each vote is weighted by compatibility of the match in terms of expected line segment orientation and length. This allows for correct handling of fragmented line data, for example, since the combined votes of all subpieces of a fragmented line count the same as the vote of a full-sized, unfragmented line. A single global histogram accumulates height votes from multiple images, and for multiple edges in a rooftop polygon. After all votes have been tallied, the histogram bucket containing the most votes yields an estimate of the roof height in the scene and a set of correspondences between rooftop edges and image line segments from multiple views.

3.4 Wireframe triangulation/extrusion

After finding a set of rooftop edge correspondences via epipolar matching, multi-image triangulation is performed to determine the precise size, shape, and position of the roof polygon in the local 3D site coordinate system. A nonlinear estimation algorithm has been developed for simultaneous multi-image, multi-line triangulation of 3D line structures.

Two versions of the triangulation subsystem have been developed. In the first, the parameters estimated for each rooftop edge are the Plücker coordinates of the algebraic 3D line coinciding with the edge. Specific points of interest, like vertices of the rooftop polygon, are computed as the intersections of these infinite algebraic lines. Plücker coordinates are a way of embedding the 4-dimensional manifold of 3D lines into R^6. Although the Plücker representation requires 6 parameters to be estimated for each line rather than 4, it simplifies the representation of geometric constraints between lines. For the generic flat-roofed rectilinear building class being considered here, a set of constraints is specified to ensure that pairs of adjacent lines in a traversal around the polygon are perpendicular, that all lines are coplanar, and that all lines are perpendicular to the Z-axis of the local site coordinate system. An iterative, nonlinear least-squares procedure determines the Plücker coordinates for all lines simultaneously such that all the object-level constraints are satisfied and an objective "fit" function is minimized that measures how well each projected algebraic line aligns with the 2D image segments that correspond to it.

Although triangulation of line structures via Plücker coordinates is general, in the sense that any set of 3D lines can be represented, we have found this approach to be computationally burdensome and numerically unstable. The reason for this is mainly due to the number of parameters in the representation and the number of constraints that must be imposed to achieve a unique, geometrically accurate solution. In particular, triangulation of a rooftop polygon containing n lines requires $6 \times n$ parameters to represent the Plücker coordinates, plus an addition $2 \times n$ Lagrange multipliers to ensure a unique solution (recall that the dimension of the line manifold is 4, thus $6-4=2$ additional constraints are required for each line to make the solution vector unique). Further constraints (and thus more Lagrange multiplier parameters) are necessary to impose the required geometric configuration on the lines in the final polygon, namely that all are coplanar and horizontal, and that adjacent pairs are perpendicular.

In response to these computational difficulties, a second version of the triangulation system has been developed using a specialized parameterization for representing flat, rectilinear polygons. The types of line structures that can be triangulated are considerably more restrictive than in the earlier, general version, however the restrictions mesh well with current system assumptions and result in a much more streamlined optimization problem. Instead of each line being represented separately, a whole rectilinear polygon is parameterized at once, using the variables shown in Figure 10. The horizontal plane containing the polygon is parameterized by a single variable $\boldsymbol{Z}$. The orientation of the rectilinear structure within that plane is represented by a single parameter$\boldsymbol{\theta}$. Finally, each separate line within the polygon is represented by a single value $\boldsymbol{r_i}$ representing the signed perpendicular distance of that line from some nominal point in the plane, usually chosen to be near the center of mass of the polygon being estimated. The representation is simple and compact, and the method of Lagrange multipliers is no longer necessary since the coplanarity and rectilinearity constraints on the polygon's shape are already built in to the representation.

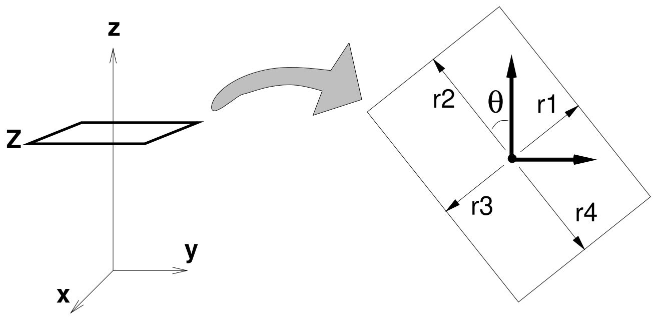

Figure 10: Parameterization of a flat, rectilinear polygon for multi-image triangulation.

Regardless of which parameterization is chosen, nonlinear estimation algorithms typically require an initial estimate that is then iteratively refined. In this system, the original rooftop polygon extracted by the building detector, and the roof height estimate computed by the epipolar matching algorithm, are used to generate an initial, flat, roof polygon. After triangulation, each 3D rooftop polygon is extruded down to the ground, as determined by the digital terrain map for the site (see Section 2.4), to form a volumetric wireframe model.

3.5 Projective intensity mapping

To provide added realism for visual displays, and as a convenient means of storage for later detailed processing of building surface information, mechanisms have been developed for projectively warping image intensities onto polygonal building facets. Planar projective transformations provide a mathematical description of how surface structure from a planar building facet maps into an image. By inverting this transformation using known building position and camera geometry, intensity information from each image can be backprojected to "paint" the walls and roof of the building model. Since multiple images are used, intensity information from all faces of the building polygon can be recovered, even though they are not all seen in any single image (see Figure 11). The full intensity-mapped site model can then be rendered to predict how the scene will appear from a new view (Figure 12), and on high-end workstations realistic real-time "fly-throughs" can be generated. For more details on the construction of the site model used to generate Figure 12, see (Collins, 1994).

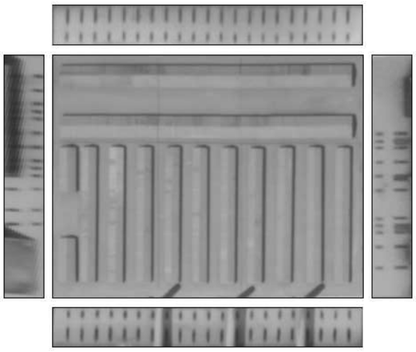

Figure 11: Intensity maps are stored with the planar facets of a building model.

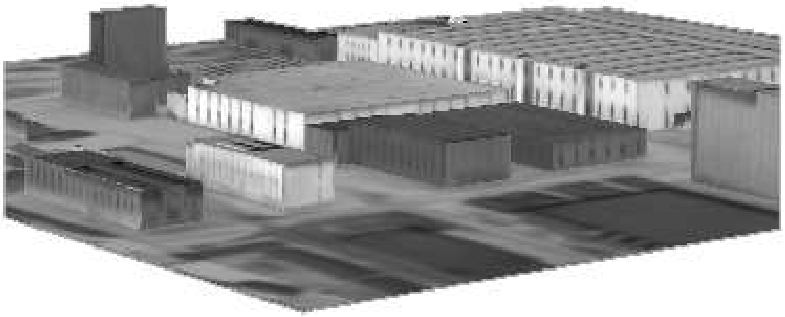

Figure 12: Intensity-mapped site model rendered from a new view.

By storing surface information with the object, intensity mapping provides a convenient storage method for later symbolic extraction of detailed surface structures like windows, doors and roof vents. Furthermore, this subsequent processing becomes greatly simplified. For example, rectangular lattices of windows or roof vents can be searched for in the unwarped intensity maps without complication from the effects of perspective distortion. Secondly, specific surface structure extraction techniques can be applied only where relevant, i.e. window and door extraction can be focused on building wall intensity maps, while roof vent computations are performed only on roofs.

When processing multiple overlapping images, each building facet will often be seen in more than one image, under a variety of viewing angles and illumination conditions. This has led to the development of a systematic mechanism for managing intensity map data, called the Orthographic Facet Library. The orthographic facet library is an indexed data set storing all of the intensity-mapped images of all the polygonal building facets that have been recovered from the site. Usually, a horizontal roof facet appears in all the aerial site images and thus has a complete set of intensity-map versions in the library. Vertical wall facets usually show up only in a subset of the site images, however, so fewer intensity-map versions are available to choose from. Each intensity-map version is tagged with a variety of spatial and photometric indices (e.g. viewing angle, resolution, sun angle) in order to facilitate retrieval and analysis by image understanding algorithms. As intensity-mapped building facets accumulate in the facet image library, knowledge about the site improves; albeit in an implicit, image-based form.

When using the facet library to render a new view of the site, it is necessary to distill the information contained in multiple intensity-mapped versions of each building facet into a single "best" image representation for that facet. Two alternative solutions have been tried so far. The first approach is to use the pixels in the **best representative version** of each facet to paint the given surface. The "goodness" of an image with respect to a particular building facet is based on a heuristic measure that takes into account the camera viewing angle, the sun angle, and the placement and geometry of other buildings in the site, all of which allow the system to compute the size, relative orientation, and photometric contrast of the facet in the image, as well as predict the percentage of the facet covered by shadows or occlusion in that view. The advantage of best version representation is its simplicity, in that only a heuristic function is calculated for each view and no further image processing is needed. The drawback of this method is that sometimes occlusions or shadows appear in ever image of a building facet, thus the representative will have to include those artifacts no matter which image is chosen. The best version representation was used to render the building in Figure 11.

In contrast to the best version approach, the **best representative piece** method takes occlusions and shadows into account. As intensity-map versions are placed in the library, pixels in the facet are partitioned into "pieces" according to whether they are sunlit or in shadow. Pixels that are labeled as occluded areas are discarded and are not considered to be a part of any piece. The idea of the best piece representation is to assign a heuristic value to each piece of an intensity-map version, rather than to the entire version. When rendering a new view, each pixel on a building's surface is backprojected to determine which pieces it is associated with. This set of pieces is ordered according o their heuristic values, and the photometric value for the pixel is selected from the highest-rated piece. Hence, all the pixels in the rendered image are the best ones available. Note, however, that some pixels in the rendered image might not exist in any of the pieces in the library, when they correspond to portions of building that have never been seen in any of the images. These pixels are painted black by default. The best version representation was used to render the site model in Figure 12.

The best piece representation is a method of data fusion, and compatability problems arise in that different pieces of each building face can appear under different sunlight conditions in different images, and thus different portions of the same building face may be assigned significantly different grey-levels, leading to a patchy appearance. One reasonable way to solve this problem is to make all the versions of the facet "similar" in intensity. Currently, a simple histogram adjustment technique is used to make the intensity distributions of all the pieces associated with a single building face uniform with respect to each other. The biggest sunlit piece of the facet is chosen as the model piece against which all other pieces are transformed.

4 SUMMARY AND FUTURE WORK

UMass has developed an image understanding system for automated site model acquisition. The algorithms currently assume a generic class of flat roofed, rectilinear buildings. To acquire a new site model, an automated building detector is run on one image to hypothesize potential building rooftops. Supporting evidence is located in other images via epipolar line segment matching, and the precise 3D shape and location of each building is determine by multi-image triangulation. Projective mapping of image intensity information onto these polyhedral building models results in a realistic site model that can be rendered using virtual "fly-through" graphics. In an operational scenario, this process would be repeated as new images become available, gradually accumulating evidence over time to make the site model database more complete and more accurate.

Several avenues for system improvement are open. One high priority is to add capabilities for detecting and triangulating peaked roof buildings. Another significant improvement would be extending the epipolar matching and triangulation portions of the system to analyze why a particular building roof hypothesis failed to be verified. There are many cases where the rooftop detector has outlined split-level buildings with a single roof polygon; automatic detection of these situations, followed by splitting of the rooftop hypothesis into two separate hypotheses, would result in an improvement in system performance.

These symbolic building extraction procedures will soon be combined with a correlation-based terrain extraction system.[14] The two techniques clearly complement each other: the terrain extraction system can determine a digital elevation map upon which the volumetric building models rest, and the symbolic building extraction procedures can identify building occlusion boundaries where correlation-based terrain recovery is expected to behave poorly. A tighter coupling of the two systems, where an initial digital elevation map is used to focus attention on distinctive humps that may be buildings, or where correlation-based reconstruction techniques are applied to building rooftop regions to identify fine surface structure like roof vents and air conditioner units, may also be investigated.

5 ACKNOWLEDGEMENTS

We would like to acknowledge the software and technical support of Robert Heller and Jonathan Lim, the video wizardry of Fred Weiss, and the administrative support of Janet Turnbull and Laurie Waskiewicz.

6 REFERENCES

[1] American Society of Photogrammetry, *Manual of Photogrammetry,* Fourth Edition, American Society of Photogrammetry, Falls Church, VA, 1980.

[2] J.R. Beveridge and E. Riseman, "Hybrid Weak-Perspective and Full-Perspective Matching," *Proc. Computer Vision and Pattern Recognition,* Champaign, IL, 1992, pp. 432-438.

[3] M. Boldt, R. Weiss and E. Riseman, "Token-Based Extraction of Straight Lines," *IEEE Transactions on Systems, Man and Cybernetics,* Vol. 19, No. 6, 1989, pp. 1581–1594.

[4] R. Collins, A. Hanson, E. Riseman and Y. Cheng, "Model Matching and Extension for Automated 3D Site Modeling," *Proceedings Arpa Image Understanding Workshop,* Washington, DC, April 1993, pp. 197–203.

[5] R. Collins, A. Hanson and E. Riseman, "Site Model Acquisition under the UMass RADIUS Project," *Proceedings Arpa Image Understanding Workshop,* Monterey, CA, November 1994, pp. 351–358.

[6] R. Collins, Y. Cheng, C. Jaynes, F. Stolle, X. Wang, A. Hanson and E. Riseman, "Site Model Acquisition and Extension from Aerial Images," *Proceedings IEEE International Conference on Computer Vision,* Cambridge, MA, June 1995, **to appear**.

[7] D. Gerson, 'RADIUS : The Government Viewpoint," *Proceedings of the Darpa Image Understanding Workshop,* San Diego, CA, January 1992, pp. 173–175.

[8] A. Huertas, C. Lin and R. Nevatia, "Detection of Buildings from Monocular Views of Aerial Scenes using Perceptual Grouping and Shadows," *Proc. Arpa Image Understanding Workshop,* Washington, DC, April 1993, pp. 253–260.

[9] C. Jaynes, F. Stolle and R. Collins, "Task Driven Perceptual Organization for Extraction of Rooftop Polygons," *Proceedings Arpa Image Understanding Workshop,* Monterey, CA, November 1994, pp. 359–365.

[10] R. Kumar and A. Hanson, "Robust Methods for Estimating Pose and Sensitivity Analysis," *CVGIP: Image Understanding,* Vol. 60, No. 3, November 1994, pp. 313-342.

[11] D. McKeown, "Toward Automatic Cartographic Feature Extraction," in *Mapping and Spatial Modelling for Navigation,* Nato ASI Series, Vol. F65, pp. 149–180, 1990.

[12] J. Mundy, R. Welty, L. Quam, T. Strat, W. Bremner, M. Horwedel, D. Hackett and A. Hoogs, "The RADIUS Common Development Environment," *Proceedings of the Darpa Image Understanding Workshop,* San Diego, CA, January 1992, pp. 215–226.

[13] M. Roux and D. McKeown, "Feature Matching for Building Extraction from Multiple Views," *Proceedings Arpa Image Understanding Workshop,* Monterey, CA, November 1994, pp. 331–349.

[14] H. Schultz, "Terrain Reconstruction from Widely Separated Images," *Integrating Photogrammetric Techniques with Scene Analysis and Machine Vision II,* Spie Proceedings Vol. 7617, Orlando, FL, April 1995, **this proceedings**.