TEC-0055

AD-A282 796

Site Model Based Image Registration and Change Detection - First Annual Report on RADIUS

R. Chellappa
L.S. Davis
X. Zhang
A. Rosenfeld
Q. Zheng
C.L. Lin
C. Rodriguez

Computer Vision Laboratory
Center for Automation Research
University of Maryland
College Park, MD 20742-3275

US Army Corps of Engineers
Topographic Engineering Center

June 1994

Approved for public release; distribution is unlimited.

Prepared for:
Advanced Research Projects Agency
3701 North Fairfax Drive
Arlington, VA 22203-1714

Monitored by:
U.S. Army Corps of Engineers
Topographic Engineering Center
7701 Telegraph Road
Alexandria, Virginia 22310-3864

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*

2. REPORT DATE
June 1994

3. REPORT TYPE AND DATES COVERED
First Annual Sep. 1992 - Sep. 1993

4. TITLE AND SUBTITLE
Site Model Based Image Registration and Change Detection - First Annual Report on RADIUS Project

5. FUNDING NUMBERS
DACA76-92-C-0024

6. AUTHOR(S)
R. Chellappa Q. Zheng L.S. Davis C.L. Lin X. Zhang
C. Rodriguez A. Rosenfeld

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Computer Vision Laboratory, Center for Automation Research
Univeristy of Maryland
College Park, MD 20742-3275

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Advanced Research Projects Agency
3701 North Fairfax Drive, Arlington, VA 22203-1714

U.S. Army Topographic Engineering Center
7701 Telegraph Road, Alexandria, VA 22310-3864

10. SPONSORING/MONITORING AGENCY REPORT NUMBER
TEC-0055

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution is unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

The University of Maryland (with TASC as a subcontractor) is one of a group of institutions doing research on aerial image understanding in support of the RADIUS program. The emphasis of our research is on knowledge-based change detection (CD) using site models and the domain expertise of image analysts (IAs). We are designing a system that allows the IA to specify what are to be considered as significant changes through quick look (QL) profiles, and to select appropriate image understanding algorithms for detecting these changes. Before CD can be attempted, the acquired images have to be registered to the site model. Two algorithms for image registration have been developed. When no information about the camera is available, we use an efficient constrained search mechanism for image-to-image registration. When an approximate camera model is available, as in RADIUS applications, we use a fast image-to-site model registration algorithm which first projects the site model into the new image domain using the given approximate camera model, and then use five control points to do camera resection and to obtain an accurate camera mode. To enable efficient transfer of technology to the IAs, we have developed our algorithms under the RADIUS Common Development Environment (RCDE).

14. SUBJECT TERMS
Aerial image understanding, change detection, image analysis, quick-look profile, registration

15. NUMBER OF PAGES
80

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UNLIMITED

Contents

List of Figures

List of Tables

PREFACE

This research is sponsored by the Advanced Research Projects Agency (ARPA) and monitored by the U.S. Army Topographic Engineering Center (TEC) under Contract DACA76-92-C-0024, titled "Site Model Based Image Registration and Change Detection - First Annual Report on RADIUS Project". The ARPA Program Manager is Dr. Oscar Firschein, and the TEC Contracting Officer's Representative is Ms. Laurette Williams.

1. Introduction

The process of locating and identifying significant changes or new activities, known as change detection (CD), is one of the most important imagery exploitation tasks [5]. Previous research on CD has emphasized the development of general-purpose methods that can be employed to screen a wide variety of imagery and determine, without access to any site-specific model information, whether any significant changes or events have occurred between the times of acquisition of the imagery. These methods have been found to be unreliable for two reasons: First, CD techniques based on more or less sophisticated differencing of images (possibly after attempted corrections for viewpoint and illumination differences) are extremely sensitive to errors in registration and in the photometric models (e.g. reflectance, illumination) that are used. Second, too many inconsequential changes occur in any natural environment. Even if general-purpose methods could be developed for screening out all changes due to variations in viewpoint, sensor and illumination, there would still be many differences between the images whose significance could only be determined by an image analyst (IA) using comprehensive site knowledge and the relevant intelligence agenda. Thus the goal of relieving the IA of the burden of screening large subsets of acquired imagery is unlikely to be achieved using such general-purpose methods.

We plan, instead, to develop a model-based vision system for CD, incorporating image understanding (IU) techniques whose primitives are specific to a particular site type. The system can be employed by the IA to use the IU techniques to conduct spatially constrained analyses whose outcomes may be indicative of occurrences of changes that have intelligence significance. The system is site model driven and will be based on three classes of primitives: *object primitives*, which correspond to the specific objects that occur in a particular site model and to the generic object classes supported by the IU system; *spatial primitives*, for the construction of search locales and the specification of constraints on the search for object types within locales; and *temporal primitives*, which can constrain or parameterize the analysis by factors such as time of day, day of week, time of year, etc. The system will assist the IA by highlighting areas on an image where there are relevant activities, new or upgraded facilities.

As reported in [5], IAs have identified two ways in which IU can be useful in CD: the "quick-look" (QL) and "final-look" (FL) modes. In the QL mode, small areas where any change would be considered significant are declared a priori, and when the system is presented with a series of images, only those that satisfy the conditions in the QL profile are marked. In the FL mode, a set of less important areas to be examined for change is specified. These areas are less important, but the IA wants to examine them to ensure complete coverage of the site. As the IA gains experience, both the QL and FL profiles can be modified. The CD system that we plan to build will primarily be guided by QL profiles.

The site models considered in the current phase of RADIUS encode only the spatial relationships between fixed objects of interest in a site, such as buildings, roads, etc. An important issue in training new analysts or reviewing infrequently analyzed sites is the coding of the temporal relationships which describe changes in the site such as movements of vehicles under normal or abnormal circumstances—i.e., a site activity model. The CD system described above will be a valuable step toward the development of a site activity modeling capability.

Generally the first step in a CD task is the registration of an image to an existing site model. Depending on the CD task, using the existing site model and camera parameters, regions of interest in the given image can be delineated. Subsequently, objects such as buildings and vehicles that are characteristically present in the site can be extracted and analyzed for CD purposes. Such object extraction algorithms cannot be purely bottom-up. For example, in extracting buildings [13], heuristics based on the expected shapes of roofs (site-specific information) are very useful for completing any partial roof hypotheses that result from imperfect bottom-up processing. Likewise, shadow analysis is very useful for obtaining height information [6, 7], or allowing the IU system to explain why some building features that are in the field of view cannot be identified in the image. Site models can also be very useful for providing geometric and photometric constraints that reduce matching ambiguities.

In addition to image-to-site-model registration, we are also interested in image-to-image registration where two images acquired from possibly severe off-nadir viewing conditions need to be registered prior to performing change detection. Image-to-image registration is useful for building site models, for developing automatic image-to-site model registration algorithms, and for performing the subtask of transforming a given image to a "favored orientation" [5]. The images to be analyzed as part of the RADIUS-related research program are high-resolution images of complicated sites. In many of the currently used image registration algorithms, tie points need to be manually selected. This can be a laborious task. Automatic registration of the two images is desirable. Given the variability of viewing directions, illumination conditions and resolution, the features used for matching may be poorly localized or occluded. Automatic image-to-image registration is accomplished using appropriate cues from site models and camera models.

It is evident that the IA must perform a crucial role in directing, manipulating and correcting the results of IU algorithms. An important part of our approach is the inclusion of early feedback, by users familiar with the final application, as to the usability of the algorithms developed under this program. These evaluations will provide valuable information with respect to the likely models and levels of interaction to be expected from IAs, the clarity and intuitive understandability of the IU algorithms, and whether the typical IA is able to tailor the responses of the algorithm to his/her needs.

2. Research Areas

2.1. Site Model Supported Monitoring

Our approach to image monitoring is based on the idea of QL profiles. QL profiles are the image exploitation recipes constructed by an IA for a given site; they characterize changes that are significant to the site. The tasks in a QL profile are related to each other both temporally and spatially. For example, if the interest of the IA in a given site concerns military activity, the first task to be performed depends on previous knowledge about the site (if it exists). If the reports from previous analyses indicate that armament was present in a training ground, the QL profile will call for vehicle detection in the training ground first.

If there are still many vehicles in the training ground, the QL profile will report "*the exercise continues*" and call for a vehicle pattern analysis task. On the other hand, if the first task reports that there are almost no vehicles in the training ground, then the QL profile will trigger vehicle detection on the roads and in the garage area. If many vehicles are found in the garage area, then the report "*the training is finished and the armament is back in the camp*" is sent. If the vehicles are not in the garage area, the QL profile will trigger pattern analysis on the road, send a report about the heading of the formation, etc. In another situation, if no previous information about troop formation is available or the reports from previous image analyses indicate that the armament was in the garage area, the first task to be called from the QL profile will be vehicle detection in the garage area; based on its results, further analysis of the road and the training ground may be called for.

In a typical site model supported monitoring task, given a new image, we first register the new image to the site model or the old images in the existing site folder. We then delineate the regions of interest according to the task. Next, 2-D templates of the objects to be monitored are formed based on their 3-D structure and information from the site model. Primitive features such as circles, ellipses, rectangles, and parallel lines are extracted, grouped and compared to the templates of the objects. Candidates with sufficient high scores of consistency with the object templates are further verified and reported to the IA. Figure 1 shows a general flowchart of our image monitoring system. For different monitoring tasks, the 2-D object models, primitive features to be extracted, and grouping mechanism are defined differently. For example, for vehicle detection from aerial images, a vehicle can be modeled as a rectangle of a certain size oriented along the road line. For detection of activities such as construction of chimneys, the needed model (for a cylindrical object) is a little more complicated. It should have an ellipse on top of two parallel lines; the minor axis of the ellipse should be parallel to the two supporting lines, which in turn are parallel to the camera viewing direction.

We have developed a preliminary design of an IU system for monitoring aerial images. The system is guided by an underlying site model, and by available knowledge of acquisition and illumination parameters, and performs task-specific image analyses for answering possible queries from an IA. We plan to extend the capabilities of our system by integrating collateral information about the various objects in the site, a user interface, and a more comprehensive set of QL profiles. Extensions to images acquired by synthetic aperture radar are also planned.

2.2. Registration Algorithms

We are investigating two types of registration processes, image-to-site-model registration and image-to-image registration. Depending on the particular CD task, e.g., if building or vehicle related activity is being monitored, we can use the site model and viewing direction of the new image to identify regions in the image that need further analysis. We can subsequently invoke the necessary IU algorithms related to detection of construction activities, vehicle location and counting (and road extraction, if construction of roads is monitored). For tasks such as these, the newly acquired image needs to be registered to the existing site model prior to any CD task.

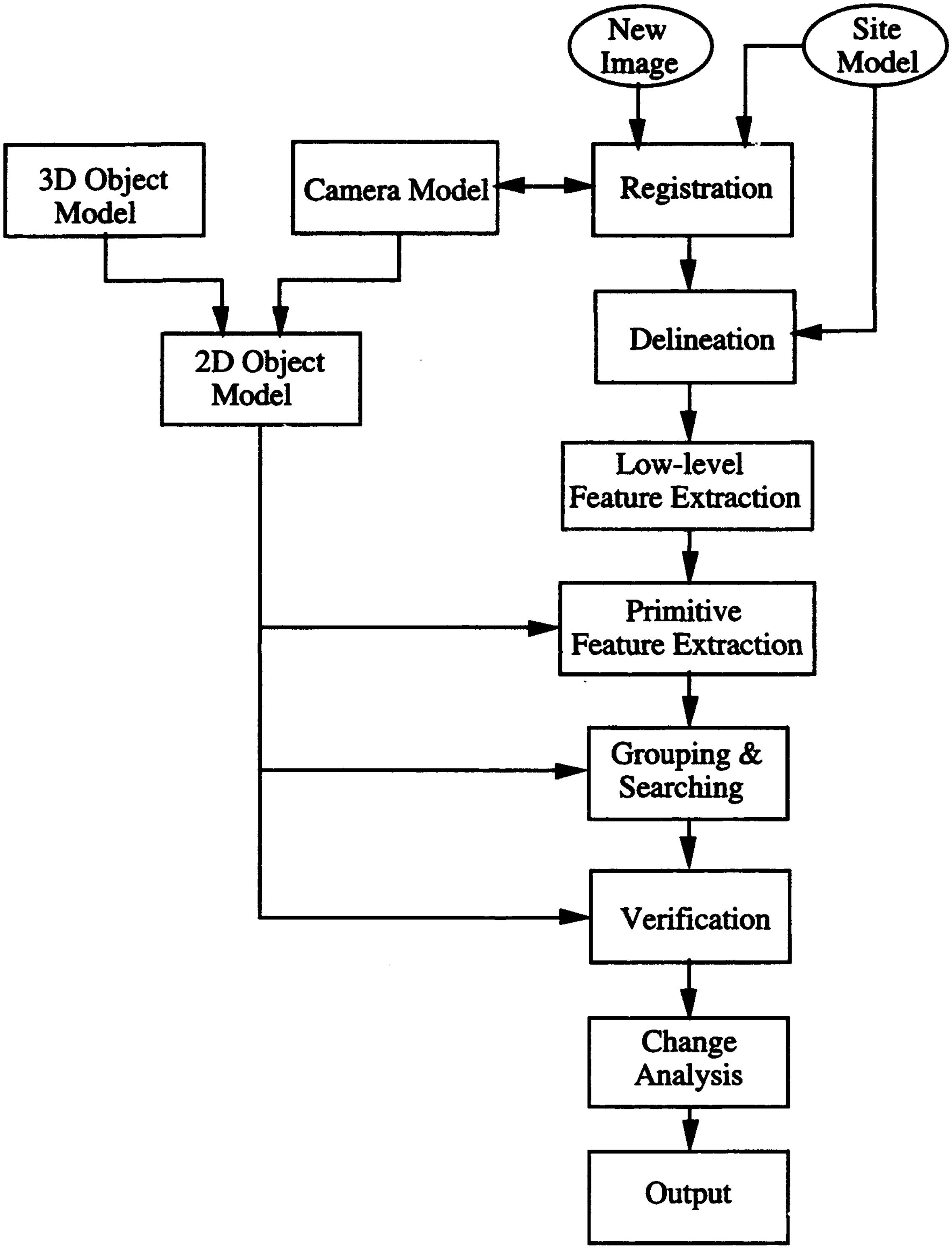


Figure 1: A block diagram of the image monitoring system.

We have developed an image-to-site-model registration procedure which first transfers the given approximate camera model to RCDE format, then requires the IA to manually adjust the locations of some points whose 3-D coordinates are known, and finally uses the RCDE camera resection function to get an accurate camera model for the newly acquired image. Our final goal in image-to-site-model registration is to make the process totally automatic. Both the selection of control points and the search for their matches in the newly acquired image will be performed automatically.

In addition to image-to-site-model registration, which will be directly useful for CD, we are also developing a general-purpose image-to-image registration algorithm. Such an algorithm will be useful for building site models, orienting an image in a "favored position", and delineating regions of interest. The traditional stereo paradigm [14] for inferring 3-D structure is not applicable to images acquired from severe off-nadir viewing directions. Our goal is to develop a completely automatic registration algorithm using site models and any auxiliary information such as camera parameters. Site models will be useful for registering two severely off-nadir images, as we can predict the contrasts of features in both images, occlusions of features and shadow regions.

2.3. Region Delineation

Region delineation is an important step for outlining the regions to be exploited by IU algorithms and providing collateral information for IU algorithms. Two kinds of region delineation are useful for CD tasks: macro region delineation and micro region delineation. Macro region delineation labels the regions of interest to the IA, hence saving computation by not monitoring irrelevant areas. Two methods for macro region delineation have been developed in our system. When the region object is available from the site model, we directly project the region boundaries onto the image to be monitored and label the region(s) in the image domain. When the region of interest is given on a map or an old image, we use the image-to-image registration to transform the regions of interest into the new image. Both methods use camera model information available from the site model. Micro region delineation further labels regions of occlusion and shadow according to the camera model and local objects. Consider, for example, the problem of identifying the region in an aerial image corresponding to a given parking lot. While estimates of sensor and platform parameters are known, it is not sufficient to simply project the parking lot boundaries onto the image plane using these parameters, since these parameters are subject to errors. Furthermore, determining which parts of the parking lot are visible in the image (since parts of the parking lot can be occluded by other objects in the site) and the illumination conditions in the visible part of the parking lot (parts of which may be in shadow depending on sun angle and site model geometry) are critical to subsequently making a correct decision as to whether there is a significant difference between the numbers of observed and expected vehicles in the parking lot. In fact, the feasibility of performing a CD task depends on the IU system correctly modeling the relationship between a given image and the site model (for example, if we were interested in whether a large number of vehicles are parked near a certain building, it could be important to determine if that part of the parking lot is, in fact, visible in the image). We have working algorithms for macro region delineation and will develop a method for micro region delineation in the second year of the project.

2.4. Site Model Construction

An integral component of site model based registration and change detection is the availability of site models. We have made considerable progress on site model construction using RCDE. We are working on updating a site model on an ongoing basis. The solution to site model construction assumes that several overlapping coverage images are available. We have constructed a site model for model-board-2 images using RCDE. The recently developed site model-to-image registration algorithm (detailed in Section 3.2.) has been used to register model board 2 images to the site model. Using the model supported construction monitoring algorithms that we have developed, as well as others under development, we will be able to form hypotheses about objects in the site. When two or more images confirm the same hypotheses about the underlying object, the initial assertions about the object will be replaced by image-derived assertions. This will be done in an incremental fashion. During the early stages, the errors due to incomplete specification of site models may be handled by allowing more tolerance in the predicted positions of features and their computed attributes. As more images become available, the representation error will decrease.

2.5. Integration of RCDE

Since the RADIUS research team includes several institutions to enable efficient sharing of research results within the community and efficient transfer of technology to IAs, it is required that all developed software be integrated into RCDE. For the RADIUS project, a program is considered as being integrated into RCDE if it is either written in Lucid Common Lisp or is a foreign function executable from RCDE (preferably through an online menu). In the first year of the RADIUS project, we have been one of the RCDE test sites and have gained considerable experience in using RCDE. We have used RCDE to build a site model which includes all the images for model board 2. We have developed a method for delineating regions of interest using RCDE basic functions. We have also transferred some of our algorithms into RCDE and made them selectable from RCDE menus. In addition, the parameters can be specified through the RCDE environment and the results are represented as RCDE objects which can be easily used by RCDE functions. Many of these programs have been ported to the Martin Marietta Group, King of Prussia, PA and tested on real images.

3. Accomplishments to Date

During the first year under the contract, we have made considerable progress on several fronts:

1. We have installed RCDE on all of our SPARC-10 systems and built a site model for the model-board-2 images.

2. We have added new functions into RCDE.

3. We have developed a novel image-to-image registration algorithm that can automatically register two off-nadir images, when no information about the camera is available.

4. We have developed a simple image-to-site-model registration mechanism that uses available (approximate) information about camera parameters.

5. We have developed image delineation algorithms that outline regions of interest useful for change detection tasks.

6. We have developed site-model-supported change detection algorithms and illustrated them for monitoring new construction and detecting and counting vehicles.

7. We have integrated algorithms for image-to-site-model registration, image delineation, and monitoring into RCDE. Many of these algorithms have been ported to the Martin Marietta Group, King of Prussia, PA.

More details about the algorithms and experimental results obtained on model board images, as well as real images, are given in the remainder of this report.

3.1. Site Model Construction

A site model is a 3-D mathematical representation of the site [1]. As minimum requirements, it includes: (a) 2-D and 3-D geometric descriptions of site features such as areas, buildings and structures, roads, etc. (b) A set of images associated with the site and their imaging conditions such as camera position, camera orientation, focal length, illuminant direction, etc. (c) Object attributes such as name, type, and status (inactive, under construction, etc.) associated with each feature.

We use the following procedure to build a site model:

1. Display two or more input images.

2. Create a default world coordinate system.

3. Create default camera models for the input images.

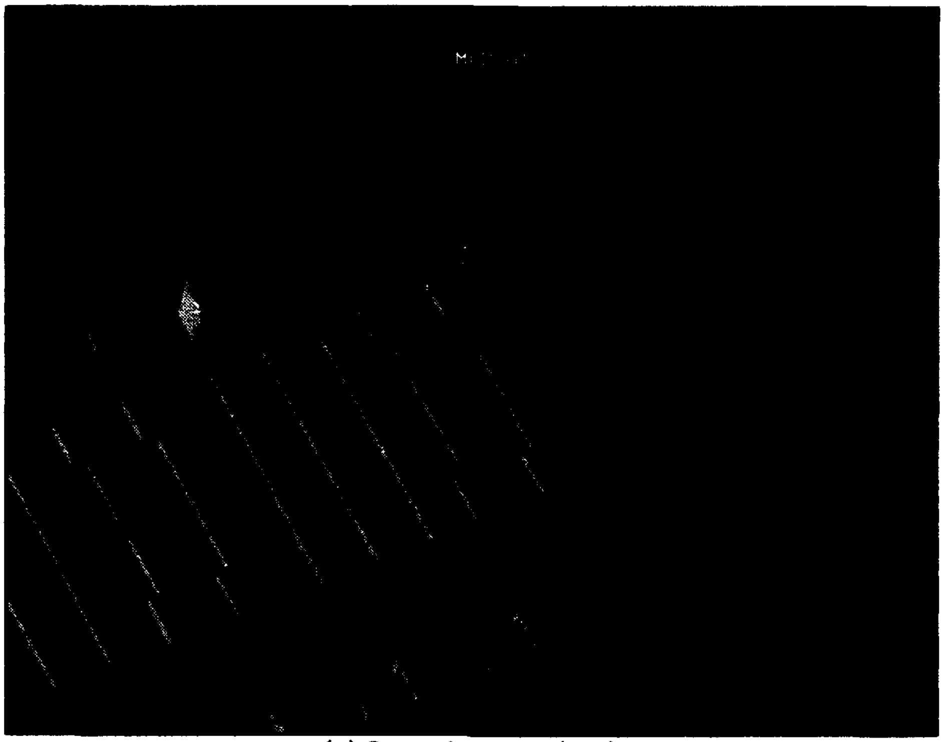
4. Manually locate (at least) four control points for each input image; the 3-D coordinates of these control points in the world coordinate system are assumed known.

5. Input the camera focal length and the location of the principal point (in the image plane) for each input image.

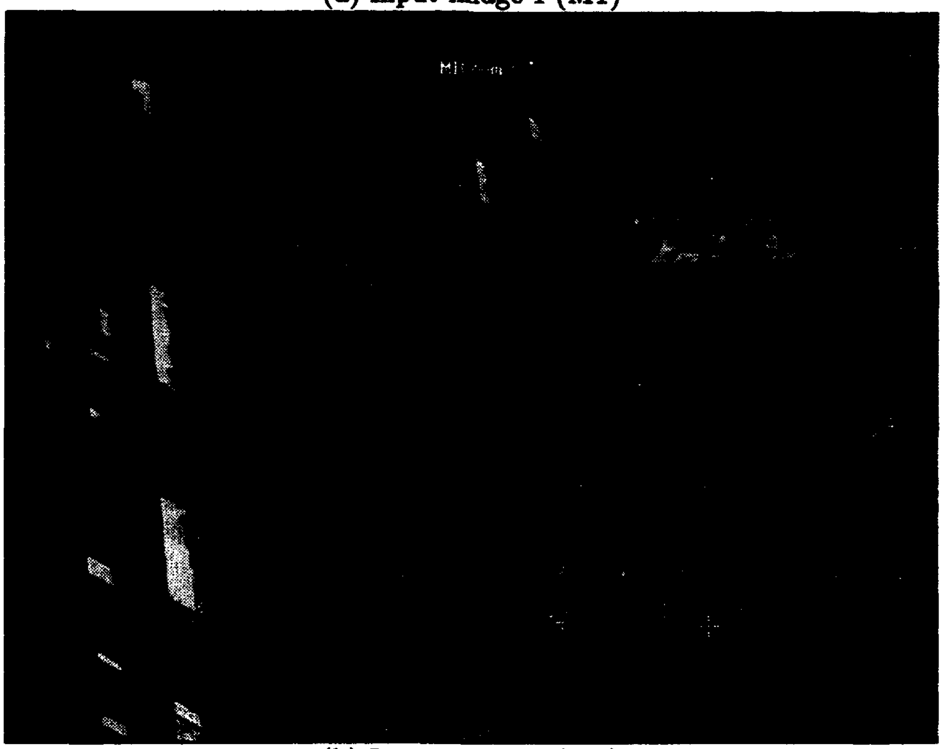
6. Do camera resection [11] to get the correct camera model for each input image.

7. Add objects to the site model interactively, using object templates such as box, cylinder, house, and their compositions, which are provided in RCDE.

Figure 2 shows an example of building a new site model for model board 2. (a) and (b) show two input images (M1 and M2) with control points marked. The 3-D coordinates and image plane indices of these control points are used to get an accurate camera model for each input image. After the new images are registered to the world coordinates, Figure 2(c) and (d) show an example of adding a building object to the site model. When adding a



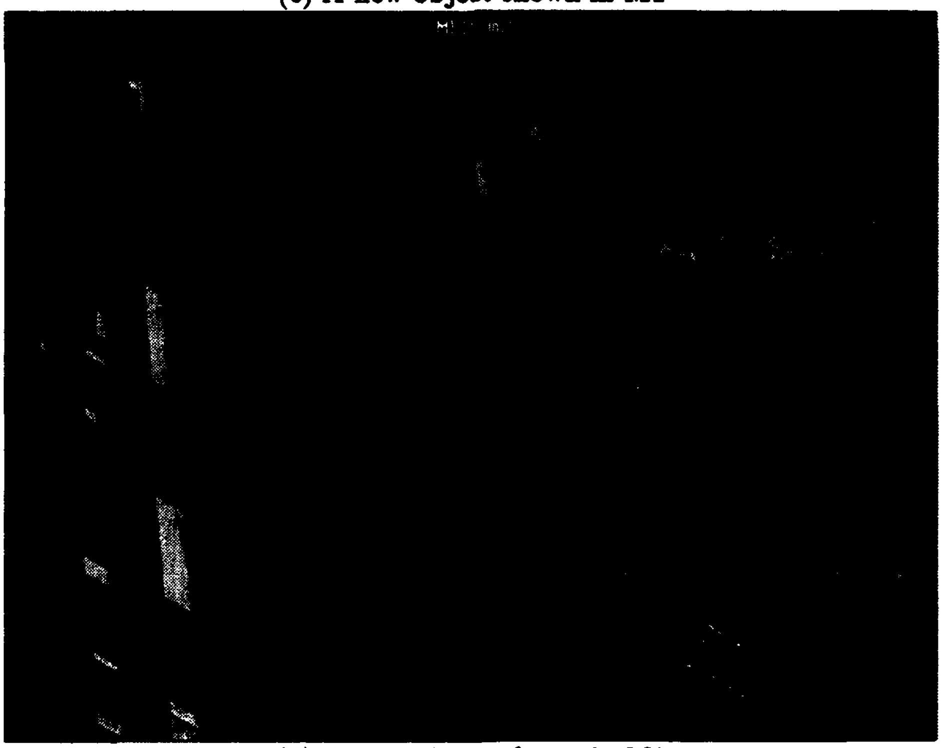
(a) Input image-1 (M1)



(b) Input image-2 (M2)

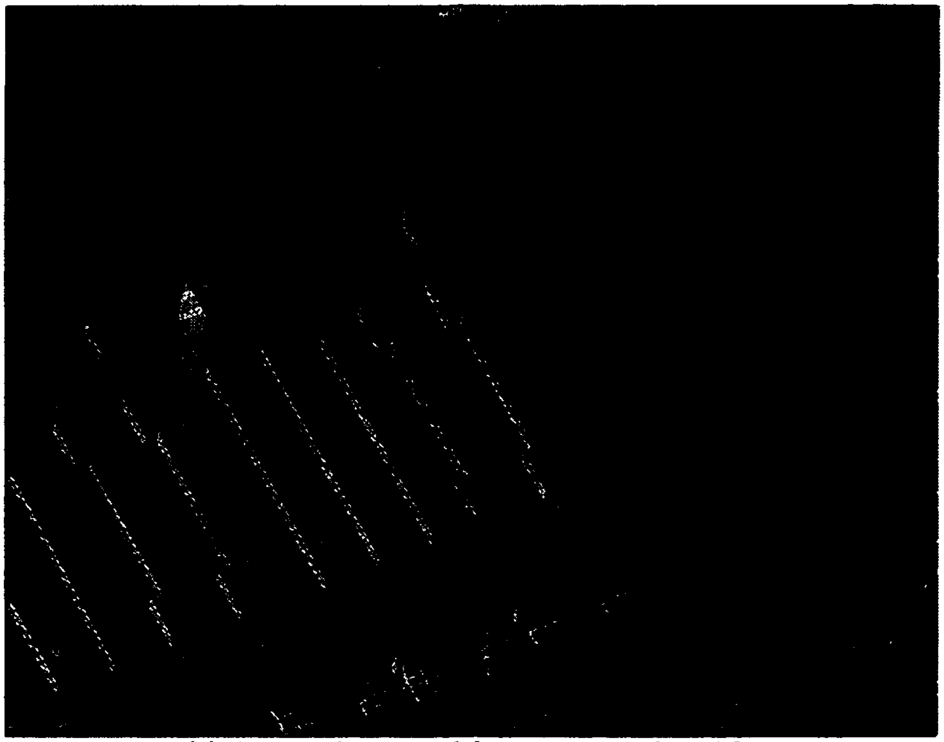
Figure 2: Building a site model using RCDE.

(c) A new object shown in M1

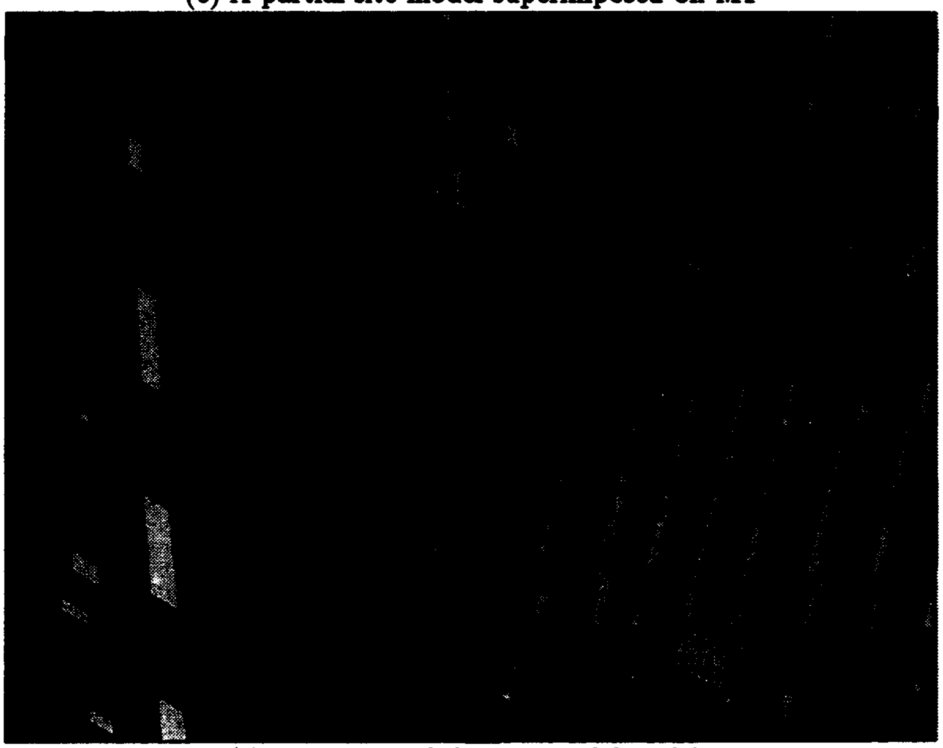


(d) A new object shown in M2

Figure 2: (cont.) Building a site model using RCDE.



(e) A partial site model superimposed on M1



(f) Rendering of the site model on M2

Figure 2: (cont.) Building a site model using RCDE.

new object, the 3-D frame of the object is displayed in all the images in the site model, and the images are used as references in adjusting the size and orientation of the 3-D object. Figure 2(e) shows a partial site model superimposed on M1, and (f) shows a rendering of the site model according to the camera model for M2.

3.2. Image-to-Site-Model Registration

In order to use information from a site model, an image has to be registered first to the site model. To register an image to the site model, we first need to understand and unify the camera models. In many image exploitation tasks, the camera parameters are available in terms of camera position and orientation in a world coordinate system, while the camera model used in photogrammetry is represented by the conformal transformation [10] in which camera parameters are represented in a camera-centered coordinate system. Here we present an image-to-site-model registration algorithm. Assuming that approximate camera parameters are available (as in the RADIUS project), a method for computation of initial camera parameters from given imaging conditions is introduced. Next, the existing site model is projected onto the new image using the initial camera model. Finally, control points with known 3-D coordinates are manually adjusted to the correct locations in the new image domain and the RCDE resection operation [11] is employed to refine the camera parameters. Using the image-to-site-model registration algorithm, we have successfully built a site model for all forty images in the model board 2 data set, verified the given control points for model board 2, and refined the camera parameters for each model board image.

In the remainder of this section, first we briefly summarize the conformal transformation used in photogrammetry. Next, we study the camera representations given in our data base. The relationship between the two representations is pointed out, followed by an algorithm for image-to-site-model registration. Experimental results on initial camera parameter estimation, camera parameter refinement, and control point verification are presented at the end of the section.

3.2.1. Conformal Transformations

In conformal transformations, camera-centered coordinates are represented by first shifting the world coordinates by $(x_o,\ y_o,\ z_o)$, then rotating the resulting coordinates around the x-axis by ω, followed by a rotation by ϕ around the resulting y-axis, and finally, a rotation by κ around the resulting z-axis. A positive rotation is defined as a clockwise rotation when viewed from the origin in the direction of the positive axis. Assuming the coordinates of a point in the world coordinate system are (x_w, y_w, z_w), and the coordinates of the point in the camera centered coordinate system are (x_c, y_c, z_c), the transform from (x_w, y_w, z_w) to (x_c, y_c, z_c) is given by

$$\begin{pmatrix} x_c \\ y_c \\ z_c \end{pmatrix} = \mathbf{R}_z(\kappa)\,\mathbf{R}_y(\phi)\,\mathbf{R}_x(\omega) \begin{pmatrix} x_w - x_o \\ y_w - y_o \\ z_w - z_o \end{pmatrix}$$

$$
\begin{aligned}
&= \begin{bmatrix} \cos\kappa & \sin\kappa & 0 \\ -\sin\kappa & \cos\kappa & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\phi & 0 & -\sin\phi \\ 0 & 1 & 0 \\ \sin\phi & 0 & \cos\phi \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\omega & \sin\omega \\ 0 & -\sin\omega & \cos\omega \end{bmatrix} \begin{pmatrix} x_w - x_o \\ y_w - y_o \\ z_w - z_o \end{pmatrix} \\
&= \mathbf{R} \begin{pmatrix} x_w - x_o \\ y_w - y_o \\ z_w - z_o \end{pmatrix} \qquad (1)
\end{aligned}
$$

where

$$
\begin{aligned}
\mathbf{R} &= \begin{bmatrix} \cos\kappa & \sin\kappa & 0 \\ -\sin\kappa & \cos\kappa & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\phi & 0 & -\sin\phi \\ 0 & 1 & 0 \\ \sin\phi & 0 & \cos\phi \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\omega & \sin\omega \\ 0 & -\sin\omega & \cos\omega \end{bmatrix} \\
&= \begin{bmatrix} \cos\kappa & \sin\kappa & 0 \\ -\sin\kappa & \cos\kappa & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\phi & \sin\phi\sin\omega & -\sin\phi\cos\omega \\ 0 & \cos\omega & \sin\omega \\ \sin\phi & -\cos\phi\sin\omega & \cos\phi\cos\omega \end{bmatrix} \\
&= \begin{bmatrix} \cos\phi\cos\kappa & \sin\omega\sin\phi\cos\kappa + \cos\omega\sin\kappa & -\cos\omega\sin\phi\cos\kappa + \sin\omega\sin\kappa \\ -\cos\phi\sin\kappa & -\sin\omega\sin\phi\sin\kappa + \cos\omega\cos\kappa & \cos\omega\sin\phi\sin\kappa + \sin\omega\cos\kappa \\ \sin\phi & -\sin\omega\cos\phi & \cos\omega\cos\phi \end{bmatrix} \qquad (2)
\end{aligned}
$$

Note that

$$
\begin{aligned}
\mathbf{R}_x^{-1}(\cdot) &= \mathbf{R}_x^t(\cdot) \\
\mathbf{R}_y^{-1}(\cdot) &= \mathbf{R}_y^t(\cdot) \\
\mathbf{R}_z^{-1}(\cdot) &= \mathbf{R}_z^t(\cdot)
\end{aligned}
$$

We have

$$
\begin{aligned}
\mathbf{R}^{-1} &= \mathbf{R}_x^{-1}\mathbf{R}_y^{-1}\mathbf{R}_z^{-1} \\
&= \mathbf{R}_x^t\mathbf{R}_y^t\mathbf{R}_z^t \\
&= \mathbf{R}^t
\end{aligned}
$$

so that

$$
\begin{pmatrix} x_w \\ y_w \\ z_w \end{pmatrix} = \mathbf{R}^t \begin{pmatrix} x_c \\ y_c \\ z_c \end{pmatrix} + \begin{pmatrix} x_o \\ y_o \\ z_o \end{pmatrix} \qquad (3)
$$

3.2.2. Camera Specification in Model-Board Data

Although it is simple to represent a camera model in conformal form, the three rotation angles ω, ϕ and κ are not intuitive. Commonly available camera parameters are camera position and camera viewing direction with respect to the world coordinate system. Given the camera viewing direction, i.e. off-nadir angle α and azimuth angle β measured east of north (as shown in Figure 3), alignment of the world coordinates to the camera-centered coordinates can be achieved through the following four operations:

1. Translate the world coordinates by (x_o, y_o, z_o);

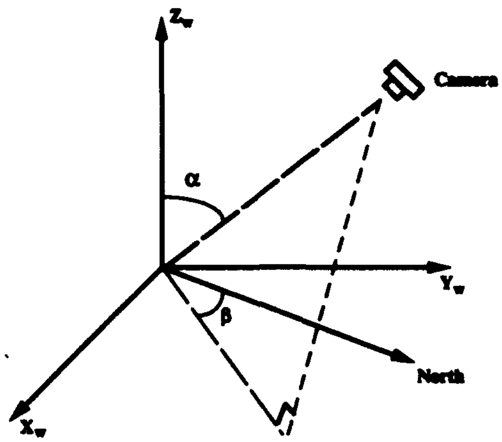


Figure 3: Camera orientation in the world coordinates.

2. Rotate around the resulting z-axis by $\hat{\beta} = N_w - \frac{\pi}{2} - \beta$ to align the y-axis with the camera azimuth direction (so that the camera is looking at the origin from the resulting positive y-axis), where N_w is the angle from the positive x-axis of the world coordinates to the north direction;

3. Rotate around the resulting x-axis by $-\alpha$, where α is the given camera elevation angle;

4. Rotate around the resulting z-axis by γ to let the north direction be N_i measured in the image domain; $\gamma = N_c - N_i$ where N_c is the (predicted) angle for the north direction after Step 3.

The rotation matrices for steps 2–4 are

$$\mathbf{R}_2 = \mathbf{R}_z(\hat{\beta}) = \begin{bmatrix} \cos\hat{\beta} & \sin\hat{\beta} & 0 \\ -\sin\hat{\beta} & \cos\hat{\beta} & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{4}$$

$$\mathbf{R}_3 = \mathbf{R}_x(\alpha) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha \\ 0 & \sin\alpha & \cos\alpha \end{bmatrix} \tag{5}$$

$$\mathbf{R}_4 = \mathbf{R}_z(\gamma) = \begin{bmatrix} \cos\gamma & \sin\gamma & 0 \\ -\sin\gamma & \cos\gamma & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{6}$$

where

$$\begin{aligned} \hat{\beta} &= N_w - \frac{\pi}{2} - \beta \\ \gamma &= N_c - N_i \end{aligned}$$

Thus, the total rotation matrix is

$$\mathbf{R} = \mathbf{R}_4\mathbf{R}_3\mathbf{R}_2$$
$$= \begin{bmatrix} \cos\gamma & \sin\gamma & 0 \\ -\sin\gamma & \cos\gamma & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\hat{\beta} & \sin\hat{\beta} & 0 \\ -\cos\alpha\sin\hat{\beta} & \cos\alpha\cos\hat{\beta} & -\sin\alpha \\ -\sin\alpha\sin\hat{\beta} & \sin\alpha\cos\hat{\beta} & \cos\alpha \end{bmatrix}$$
$$= \begin{bmatrix} \cos\gamma\cos\hat{\beta} - \sin\gamma\cos\alpha\sin\hat{\beta} & \cos\gamma\sin\hat{\beta} + \sin\gamma\cos\alpha\cos\hat{\beta} & -\sin\gamma\sin\alpha \\ -\sin\gamma\cos\hat{\beta} - \cos\gamma\cos\alpha\sin\hat{\beta} & -\sin\gamma\sin\hat{\beta} + \cos\gamma\cos\alpha\cos\hat{\beta} & -\cos\gamma\sin\alpha \\ -\sin\alpha\sin\hat{\beta} & \sin\alpha\cos\hat{\beta} & \cos\alpha \end{bmatrix} \quad (7)$$

When the distance from the camera to the stare point, r, is given, the initial camera translation is computed as

$$x_o = x_p + x_{or} \quad (8)$$
$$y_o = y_p + y_{or} \quad (9)$$
$$z_o = z_p + z_{or} \quad (10)$$

where (x_p, y_p, z_p) are the 3-D coordinates of the stare point, and

$$x_{or} = \mathbf{r}\sin\alpha\cos(N_w - \beta) \quad (11)$$
$$y_{or} = \mathbf{r}\sin\alpha\sin(N_w - \beta) \quad (12)$$
$$z_{or} = \mathbf{r}\cos\alpha \quad (13)$$

3.2.3. Relationship Between the Two Representations

As the values of x_o, y_o, z_o and $\mathbf{R}$ are independent of the interpretation of how the camera is aligned, by comparing the corresponding terms in (2) and (7) we can convert the camera parameters from one representation to the other. Note that the camera is always above the horizon, so that

$$-\frac{\pi}{2} \leq \omega \leq \frac{\pi}{2}$$
$$-\frac{\pi}{2} \leq \phi \leq \frac{\pi}{2}$$
$$0 \leq \alpha \leq \frac{\pi}{2}$$

Hence

$$\cos\omega \geq 0$$
$$\cos\phi \geq 0$$
$$\sin\alpha \geq 0$$

The relationship between the camera parameters in the two representations are

$$\omega = \arctan\left(\frac{-\sin\alpha\cos\hat{\beta}}{\cos\alpha}\right) \quad (14)$$

$$\begin{aligned}
\phi &= \arcsin(-\sin\alpha\sin\hat{\beta}) && (15)\\
\kappa &= \arctan\left(\frac{\sin\gamma\cos\hat{\beta}+\cos\gamma\cos\alpha\sin\hat{\beta}}{\cos\gamma\cos\hat{\beta}-\sin\gamma\cos\alpha\sin\hat{\beta}}\right) && (16)\\
\alpha &= \arccos(\cos\omega\cos\phi) && (17)\\
\hat{\beta} &= \arctan\left(\frac{-\sin\phi}{-\sin\omega\cos\phi}\right) && (18)\\
\gamma &= \arctan\left(\frac{-\sin\omega\sin\kappa+\cos\omega\sin\phi\cos\kappa}{-\sin\omega\cos\kappa-\cos\omega\sin\phi\sin\kappa}\right) && (19)
\end{aligned}$$

3.2.4. Camera Roll Estimation

Given the viewing direction of the camera, the camera can still rotate around its optical axis, leaving one degree of freedom undetermined. If the north direction is known in world coordinates, we can determine the orientation of the north vector in camera-centered coordinates which are free of camera roll; the angle between the predicted north direction, and the north direction in the image plane is equal to the camera roll angle. Since the camera azimuth angle β is measured east of north, after aligning the y-axis with the camera viewing direction, the angle from the x-axis to the north direction is $\frac{\pi}{2}+\beta$ where β is the azimuth angle of the (given) camera viewing direction. We then rotate the axes around the resulting x-axis by $-\alpha$ to align the z-axis with the camera viewing direction, the angle from the resulting x-axis to the north direction projected onto the x–y plane is

$$N_c = \arctan\frac{\cos\alpha\ \sin(\frac{\pi}{2}+\beta)}{\cos(\frac{\pi}{2}+\beta)} = \arctan\frac{\cos\alpha\ \cos\beta}{-\sin\beta} \qquad (20)$$

In our work, we estimate the north direction in an image plane by hand picking two points along "access road 1", (X_1, Y_1) and (X_2, Y_2), and computing

$$N_i = \arctan\frac{Y_2 - Y_1}{X_2 - X_1} \qquad (21)$$

The camera roll angle is then computed as

$$\gamma = N_c - N_i = \arctan\frac{\cos\alpha\ \cos\beta}{-\sin\beta} - N_i \qquad (22)$$

3.2.5. Stare Point Estimation

With the camera rotation angles determined, we still need the coordinates of camera center (in the world coordinate system) to register the camera coordinates to the world coordinates. The camera position is usually determined by giving the stare point and the distance between the stare point and the camera center. The stare point information is important for automatical camera model refinement. For model board 2 data the stare points are not available. We estimate the stare point by the difference between the coordinates of a known 3-D point in the approximated camera model and its correct coordinates. First assuming the stare points (x_p, y_p, z_p) is available, we have the transform from the world coordinates (x_w, y_w, z_w) to the camera centered coordinates (x_c, y_c, z_c) as

$$\begin{pmatrix} x_c \\ y_c \\ z_c \end{pmatrix} = \mathbf{R} \begin{pmatrix} x_w - x_{or} - x_p \\ y_w - y_{or} - y_p \\ z_w - z_{or} - z_p \end{pmatrix} \tag{23}$$

where (x_{or}, y_{or}, z_{or}) are determined by (11–13). Next consider the case that the stare point is either not available or not correct, assuming it be ($\hat{x}_p$, $\hat{y}_p$, $\hat{z}_p$). The camera transformation becomes

$$\begin{pmatrix} \hat{x}_c \\ \hat{y}_c \\ \hat{z}_c \end{pmatrix} = \mathbf{R} \begin{pmatrix} x_w - x_{or} - \hat{x}_p \\ y_w - y_{or} - \hat{y}_p \\ z_w - z_{or} - \hat{z}_p \end{pmatrix} \tag{24}$$

Now, pick a point ($\check{x}_w$, $\check{y}_w$, $\check{z}_w$) whose coordinates under transform (24) equal to (x_c, y_c, z_c) under (23) as

$$\begin{pmatrix} x_c \\ y_c \\ z_c \end{pmatrix} = \mathbf{R} \begin{pmatrix} \check{x}_w - x_{or} - \hat{x}_p \\ \check{y}_w - y_{or} - \hat{y}_p \\ \check{z}_w - z_{or} - \hat{z}_p \end{pmatrix} \tag{25}$$

By taking the difference between (23) and (25) we get

$$\begin{pmatrix} x_w - x_p - \check{x}_w + \hat{x}_p \\ y_w - y_p - \check{y}_w + \hat{y}_p \\ z_w - z_p - \check{z}_w + \hat{z}_p \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \tag{26}$$

or

$$\begin{pmatrix} x_p - \hat{x}_p \\ y_p - \hat{y}_p \\ z_p - \hat{z}_p \end{pmatrix} = \begin{pmatrix} x_w - \check{x}_w \\ y_w - \check{y}_w \\ z_w - \check{z}_w \end{pmatrix} \tag{27}$$

If we select $(x_w, y_w, z_w) = (0, 0, 0)$ then the adjustment for the initial stare point estimation is ($\check{x}_w$, $\check{y}_w$, $\check{z}_w$). The procedure for adjusting the stare point estimation is as follows

1. Project the site model to the new image domain using the given approximated camera model.
2. Determine the coordinates of the origin under the approximated camera model $(\check{x}_w, \check{y}_w, \check{z}_w)$.
3. The adjustment for the stare point estimation is $(-\check{x}_w, -\check{y}_w, -\check{z}_w)$.

3.2.6. Algorithm

We use the following procedure to register a new image to an existing site model:

1. Manually select two points along the north direction and compute N_i using (21).
2. Compute the camera roll angle γ using (22).
3. Compute the conformal camera parameters using (8–16). Set (x_p, y_p, z_p) to zero if they are unavailable.

4. Project the site model to the new image using the parameters obtained from step 3.
5. Adjust the estimation of the stare point if the offset in the initial projection is too large.
6. Manually adjust (at least) four control points to the correct locations in the image domain and do camera resection to get more accurate conformal camera parameters.
7. Compute the camera parameters in the world coordinate system using (18–19) and

$$\beta = N_w - \frac{\pi}{2} - \hat{\beta} \tag{28}$$

$$\mathbf{r} = \frac{z_o}{\cos\alpha} \tag{29}$$

$$x_p = x_o - \mathbf{r}\sin\alpha\cos(N_w - \beta) \tag{30}$$

$$y_p = y_o - \mathbf{r}\sin\alpha\sin(N_w - \beta) \tag{31}$$

where we have assumed $z_p = 0$.

3.2.7. Experiments

Camera Roll Estimation

For each of the forty model-board-2 images, we manually select two points along the north direction and compute N_i and γ using (21) and (22). The results are listed in Table 1.

Camera Parameter Calibration

Given the camera viewing direction (α, β), the camera range $\mathbf{r}$, and the camera roll angle estimated, we use (11–13) and (14–16) to compute the camera parameters for the conformal representation. The results are listed in Table 1.

We then apply camera resection, based on these initial camera parameters (assume $x_p = y_p = z_p = 0$) and correspondences of five control points, to obtain refined camera parameters $\tilde{x}_o$, $\tilde{y}_o$, $\tilde{z}_o$, $\tilde{\omega}$, $\tilde{\phi}$ and $\tilde{\kappa}$. We further compute the corresponding refined camera parameters in world coordinates, α, β, γ, $\mathbf{r}$, x_p and y_p, using (18–19) and (28–31). Table 2 lists the refined camera parameters for all the model-board-2 images. Table 3 lists the differences between the given and refined camera parameters.

Figure 4 shows an example of registering a new image, M38, shown in (a), to the existing site model. The estimated camera roll angle for M38 is $\gamma = 90.1°$. The given camera elevation and azimuth angles are $\alpha = 40°$ and $\beta = 90°$. The approximate range is $\mathbf{r} = 10850$ feet. From these we compute the initial camera parameters as $x_o = 6974$ feet, $y_o = 0$ feet, $z_o = 8311$ feet, $\omega = 0.0°$, $\phi = 40.0°$, and $\kappa = 0.1°$. In computing x_o, y_o and z_o, we set the unknown parameters x_p, y_p and z_p to zero. Figure 4(b) shows the projection of the existing site model into the new image using the above approximate camera parameters. In many applications an approximate stare point is available. Figure 4(c) shows the projection of the existing site model into the new image domain when an approximate stare point is available.

Table 1: Initial Camera Parameters

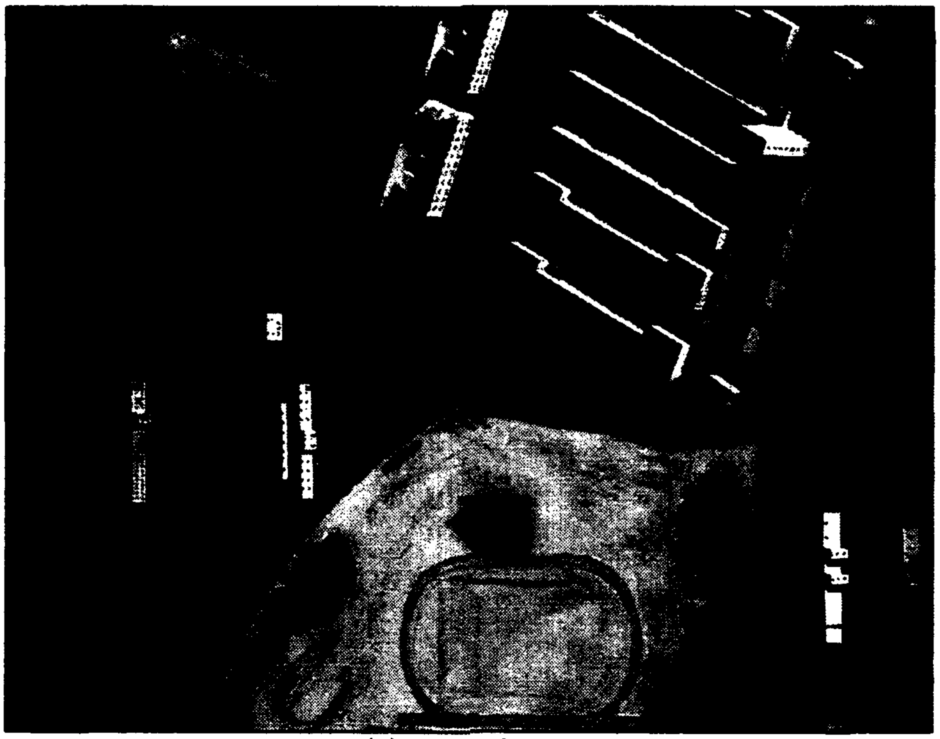
image	N_i (°)	α (°)	β (°)	r (ft)	γ (°)	x_{or} (ft)	y_{or} (ft)	z_{or} (ft)	ω (°)	ϕ (°)	κ (°)
M1	226.2	30	340	10850	-158.9	-1855	5097	9396	-28.5	-9.8	-141.5
M2	28.7	45	54	10850	124.1	6206	4509	7672	-30.4	34.9	79.9
M3	1.0	0	34	10850	123.0	0	0	10850	0.0	0.0	89.0
M4	357.9	30	75	10850	-191.0	5240	1404	9396	-8.5	28.9	96.2
M5	357.3	30	255	10850	-10.4	-5240	-1404	9396	8.5	-28.9	96.8
M6	88.6	15	86	10850	87.5	2801	195	10480	-1.1	15.0	1.6
M7	9[illegible].2	15	333	10850	-30.0	-1274	2502	10480	-13.4	-6.7	-3.8
M8	285.0	45	352	10850	-206.2	-1067	7597	7672	-44.7	-5.6	159.4
M9	269.5	35	180	10850	0.5	0	-6223	8887	35.0	0.0	-179.5
M10	185.4	30	223	10850	-228.3	-3699	-3967	9396	22.9	-19.9	-87.2
M11	206.5	0	100	10850	-16.5	0	0	10850	0.0	0.0	-116.5
M12	356.8	45	317	10850	-319.6	-5232	5611	7672	-36.2	-28.8	73.8
M13	0.1	30	51	10850	144.9	4216	3414	9396	-20.0	22.9	98.0
M14	352.5	40	356	10850	-267.7	-486	6957	8311	-39.9	-2.6	95.4
M15	95.3	30	210	10850	-151.6	-2712	-4698	9396	26.6	-14.5	1.8
M16	287.2	45	29	10850	-159.1	3719	6710	7672	-41.2	20.0	179.5
M17	183.3	25	100	10850	5.8	4515	-796	9833	4.6	24.6	-95.2
M18	354.3	30	10	10850	-252.8	942	5342	9396	-29.6	5.0	98.5
M19	222.5	30	40	10850	-88.4	3487	4155	9396	-23.9	18.7	-124.4
M20	227.8	25	186	10850	-311.2	-479	-4560	9833	24.9	-2.5	-136.7
M21	358.7	30	75	10850	-191.7	5240	1404	9396	-8.5	28.9	95.4
M22	0.3	30	255	10850	-13.4	-5240	-1404	9396	8.5	-28.9	93.8
M23	261.4	45	317	10850	-224.2	-5232	5611	7672	-36.2	-28.8	169.2
M24	176.4	25	80	10850	-5.4	4515	796	9833	-4.6	24.6	-84.4
M25	174.5	30	146	10850	-302.4	3033	-4497	9396	25.6	16.2	-92.1
M26	84.1	45	9	10850	18.5	1200	7577	7672	-44.6	6.4	12.1
M27	349.2	45	80	10850	-176.3	7555	1332	7672	-9.9	44.1	107.7
M28	113.8	15	15	10850	-8.3	726	2712	10480	-14.5	3.8	-22.8
M29	270.3	35	278	10850	-263.7	-6162	866	8887	-5.6	-34.6	176.6
M30	273.5	25	186	10850	3.1	-479	-4560	9833	24.9	-2.5	177.7
M31	351.1	30	350	10850	-272.6	-942	5342	9396	-29.6	-5.0	96.1
M32	79.5	45	115	10850	-241.2	6953	-3242	7672	22.9	39.9	-4.6
M33	186.0	30	40	10850	-51.9	3487	4155	9396	-23.9	18.7	-87.9
M34	359.1	15	55	10850	-213.2	2300	1610	10480	-8.7	12.2	92.7
M35	88.5	45	300	10850	-66.3	-6644	3836	7672	-26.6	-37.8	-15.6
M36	278.1	30	70	10850	-115.6	5097	1855	9396	-11.2	28.0	177.2
M37	12.4	30	115	10850	-170.4	4916	-2292	9396	13.7	26.9	71.3
M38	89.9	40	90	10850	-269.9	6974	0	8311	0.0	40.0	0.1
M39	269.4	30	165	10850	-16.6	1404	-5240	9396	29.1	7.4	176.5
M40	7.6	40	220	10850	-50.0	-4482	-5342	8311	32.7	-24.4	97.3

Table 2: Camera Parameters after Camera Resection

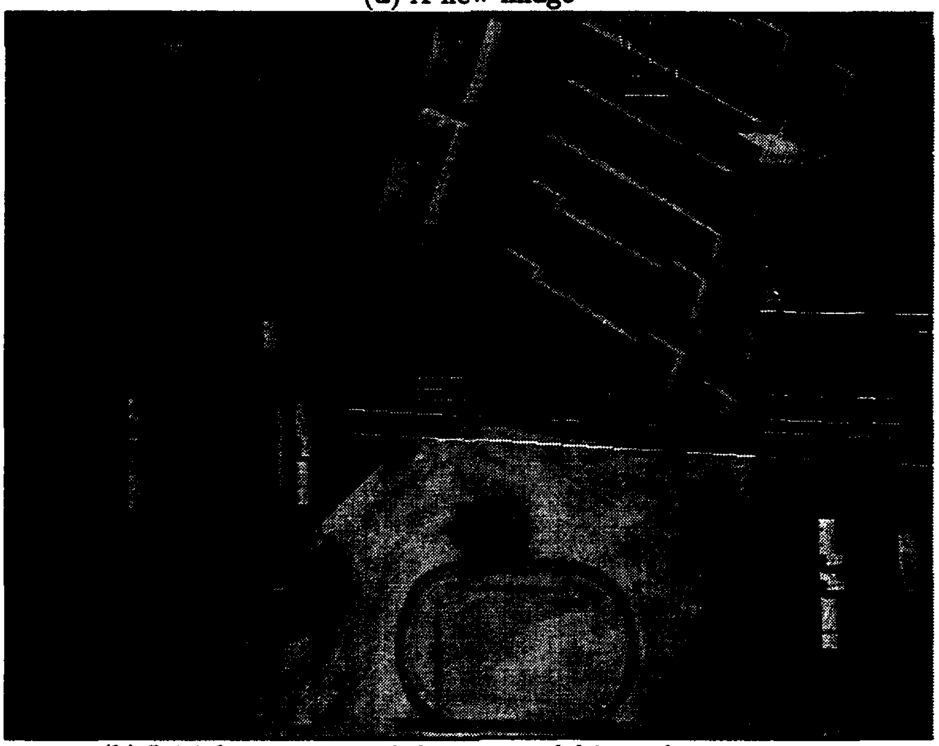
image	α (°)	β (°)	r (ft)	γ (°)	x_o (ft)	y_o (ft)	z_o (ft)	ω (°)	ϕ (°)	κ (°)
M1	30.7	-20.8	10321	-162.4	-681	5751	8871	-29.1	-10.4	-144.3
M2	45.8	51.5	10586	123.7	6715	5350	7379	-32.6	34.1	82.5
M3	1.7	-103.0	10361	-15.5	788	177	10356	0.4	-1.7	87.4
M4	31.3	85.6	10208	176.8	6707	608	8719	-2.7	31.2	91.9
M5	29.9	-108.1	10536	-15.1	-3574	-1395	9133	10.1	-28.3	95.5
M6	20.1	94.6	10439	95.6	4219	194	9802	1.7	20.1	0.7
M7	15.9	3.6	10423	2.9	762	3448	10022	-15.9	1.0	-0.6
M8	44.8	-6.7	10384	150.1	74	7977	7368	-44.6	-4.7	154.8
M9	35.1	-177.7	10156	2.3	793	-5614	8306	35.1	-1.3	-179.5
M10	29.2	-144.1	10773	123.8	-2179	-3815	9404	24.3	-16.6	-88.6
M11	6.5	125.7	10764	7.7	1969	143	10694	3.8	5.3	-118.2
M12	45.7	-47.2	10636	32.8	-4762	5833	7431	-34.8	-31.7	69.9
M13	29.2	78.8	10791	168.0	6097	1496	9417	-6.2	28.6	90.8
M14	46.4	-3.2	10486	88.2	744	8505	7233	-46.3	-2.3	90.4
M15	30.7	-155.8	10541	-158.2	-1277	-4577	9067	28.4	-12.1	0.6
M16	47.0	36.3	10706	-154.1	5772	7219	7297	-40.9	25.7	179.2
M17	24.1	121.1	11061	24.2	4782	-1702	10095	13.0	20.5	-99.3
M18	29.0	-27.3	10873	61.6	-1396	5551	9511	-26.2	-12.8	85.9
M19	33.0	33.5	10285	-97.1	4237	5309	8623	-28.5	17.5	-126.1
M20	26.0	-173.0	10307	48.6	484	-4068	9265	25.8	-3.1	-137.8
M21	31.7	92.0	10313	-178.1	6641	727	8774	1.3	31.7	89.5
M22	29.9	-107.2	10646	-16.3	-3561	-694	9225	9.7	-28.5	93.4
M23	45.0	-50.7	10359	125.4	-4844	5352	7327	-32.3	-33.2	166.2
M24	24.6	90.2	10396	2.2	5305	209	9454	0.1	24.6	-88.0
M25	29.0	165.5	10322	78.3	2125	-4583	9027	28.2	7.0	-88.9
M26	45.8	9.2	10705	14.2	2385	8350	7467	-45.4	6.6	7.8
M27	45.1	81.3	10670	-176.9	8585	1762	7529	-8.7	44.5	105.4
M28	15.8	25.1	10856	0.0	1959	3154	10445	-14.4	6.6	-24.3
M29	34.9	-90.8	10768	88.1	-5293	620	8828	0.6	-34.9	179.1
M30	24.4	-164.6	10630	12.7	-266	-3592	9678	23.7	-6.3	178.6
M31	34.3	-11.4	10583	80.2	-382	6755	8742	-33.8	-6.4	89.7
M32	45.0	129.2	10747	129.4	7079	-4306	7592	32.4	33.2	-9.8
M33	33.6	36.9	10818	-57.1	4719	5597	9011	-28.0	19.4	-89.1
M34	15.3	69.4	10985	158.6	3830	1539	10596	-5.5	14.3	89.9
M35	45.6	-69.5	10962	-75.9	-6414	3280	7664	-19.7	-42.1	-14.0
M36	31.8	58.5	10756	-127.6	6077	3260	9146	-17.9	26.7	178.1
M37	31.4	133.7	10347	-154.5	5106	-3041	8835	22.8	22.1	67.3
M38	42.6	93.3	10563	91.9	7977	309	7776	3.0	42.5	-2.6
M39	33.0	159.5	10649	-24.3	3062	-4889	8935	31.3	11.0	173.1
M40	44.7	-105.7	10805	-19.9	-6508	-1440	7679	15.0	-42.6	91.6

Table 3: Corrections for Camera Parameters after Camera Resection

image	$\delta\alpha$	$\delta\beta$	$\delta\mathbf{r}$	$\delta\gamma$	δx_{or}	δy_{or}	δz_{or}	$\delta\omega$	$\delta\phi$	$\delta\kappa$	x_p	y_p
M1	0.7	-0.8	-529	-3.5	-14	-164	-525	-0.6	-0.6	-2.8	1187	818
M2	0.8	-2.5	-264	-0.4	-267	216	-293	-2.2	-0.8	2.6	776	625
M3	1.7	-137.0	-489	-138.5	-302	-69	-494	0.4	-1.7	-1.6	1091	246
M4	1.3	10.6	-642	7.8	53	-996	-677	5.8	2.3	-4.3	1414	200
M5	-0.1	-3.1	-314	-4.7	247	-225	-263	1.6	0.6	-1.3	1419	234
M6	5.1	8.6	-411	8.1	779	-484	-678	2.8	5.1	-0.9	638	483
M7	0.9	30.6	-427	32.9	1455	353	-458	-2.5	7.7	3.2	581	593
M8	-0.2	1.3	-466	-3.7	217	-331	-304	0.1	0.9	-4.6	924	710
M9	0.1	2.3	-694	1.8	-230	384	-581	0.1	-1.3	0.0	1024	224
M10	-0.8	-7.1	-77	-7.9	615	-289	8	1.4	3.3	-1.4	904	440
M11	6.5	25.7	-86	24.2	995	-715	-156	3.8	5.3	-1.7	974	859
M12	0.7	-4.2	-214	-7.6	-350	-439	-241	1.4	-2.9	-3.9	819	661
M13	-0.8	27.8	-59	23.1	952	-2389	21	13.8	5.7	-7.2	929	471
M14	6.4	0.8	-364	-4.1	67	624	-1078	-6.4	0.3	-5.0	1164	923
M15	0.7	-5.8	-309	-6.6	504	-205	-329	1.8	2.4	-1.2	930	325
M16	2.0	7.3	-144	5.0	921	-398	-375	0.3	5.7	-0.3	1131	907
M17	-0.9	21.1	211	18.4	-647	-1542	262	8.4	-4.1	-4.1	913	635
M18	-1.0	-37.3	23	-45.6	-3356	-659	115	3.4	-17.8	-12.6	1018	867
M19	3.0	-6.5	-565	-8.7	-392	518	-773	-4.6	-1.2	-1.7	1141	635
M20	1.0	1.0	-543	-0.2	-73	80	-568	0.9	-0.6	-1.1	1036	411
M21	1.7	17.0	-537	13.6	175	-1597	-622	9.8	2.8	-5.9	1225	920
M22	-0.1	-2.2	-204	-2.9	164	-167	-171	1.2	0.4	-0.4	1514	877
M23	0.0	-7.7	-491	-10.4	-436	-975	-345	3.9	-4.4	-3.0	824	715
M24	-0.4	10.2	-454	7.6	-191	-811	-379	4.7	0.0	-3.6	981	224
M25	-1.0	19.5	-528	20.7	-1778	-348	-369	2.6	-9.2	3.2	870	261
M26	0.8	0.2	-145	-4.3	25	-5	-205	-0.8	0.2	-4.3	1159	778
M27	0.1	1.3	-180	-0.6	-82	-186	-143	1.2	0.4	-2.3	1112	616
M28	0.8	10.1	6	8.3	528	-34	-35	0.1	2.8	-1.5	705	475
M29	-0.1	-8.8	-82	-8.2	-3	-951	-59	6.2	-0.3	2.5	871	706
M30	-0.6	9.4	-220	9.6	-690	321	-155	-1.2	-3.8	0.9	903	646
M31	4.3	-1.4	-267	-7.2	-238	504	-654	-4.2	-1.4	-6.4	798	908
M32	0.0	14.2	-103	10.6	-1063	-1569	-80	9.5	-6.7	-5.2	1188	505
M33	3.6	-3.1	-32	-5.2	103	633	-385	-4.1	0.7	-1.2	1128	808
M34	0.3	14.4	135	11.8	412	-590	116	3.2	2.1	-2.8	1118	518
M35	0.6	-9.5	112	-9.6	-699	-1095	-8	6.9	-4.3	1.6	928	539
M36	1.8	-11.5	-94	-12.0	-270	1100	-250	-6.7	-1.3	0.9	1250	304
M37	1.4	18.7	-503	15.9	-1021	-1424	-561	9.1	-4.8	-4.0	1210	674
M38	2.6	3.3	-287	1.8	164	-409	-535	3.0	2.5	-2.7	839	719
M39	3.0	-5.5	-201	-7.7	622	-187	-461	2.2	3.6	-3.4	1036	537
M40	4.7	34.3	-45	30.1	-2836	3287	-632	-17.7	-18.2	-5.7	810	614

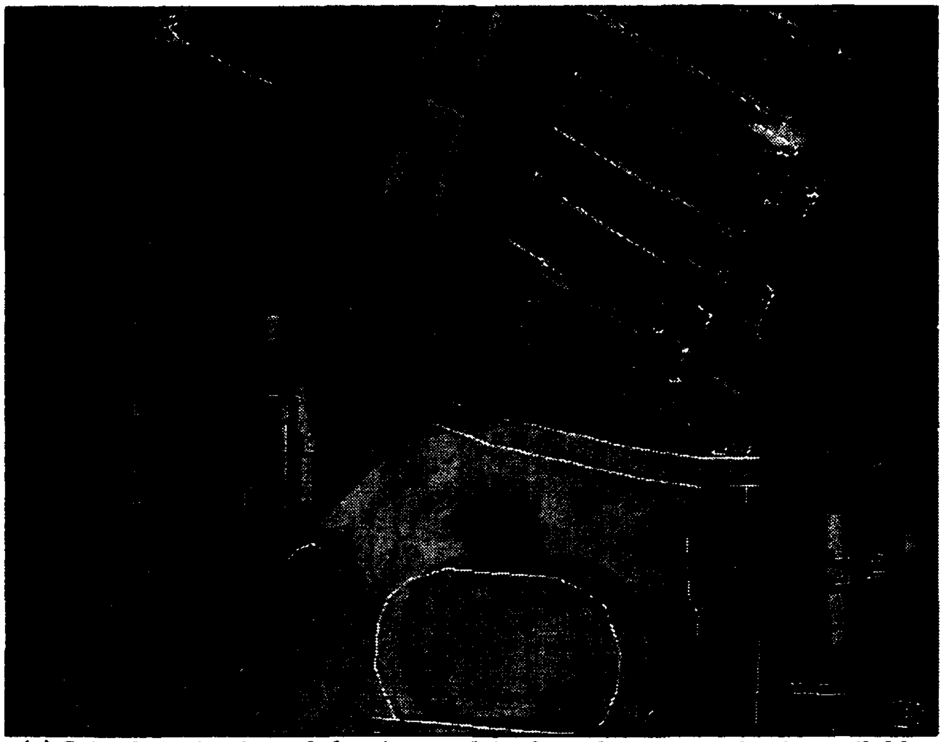


(a) A new image

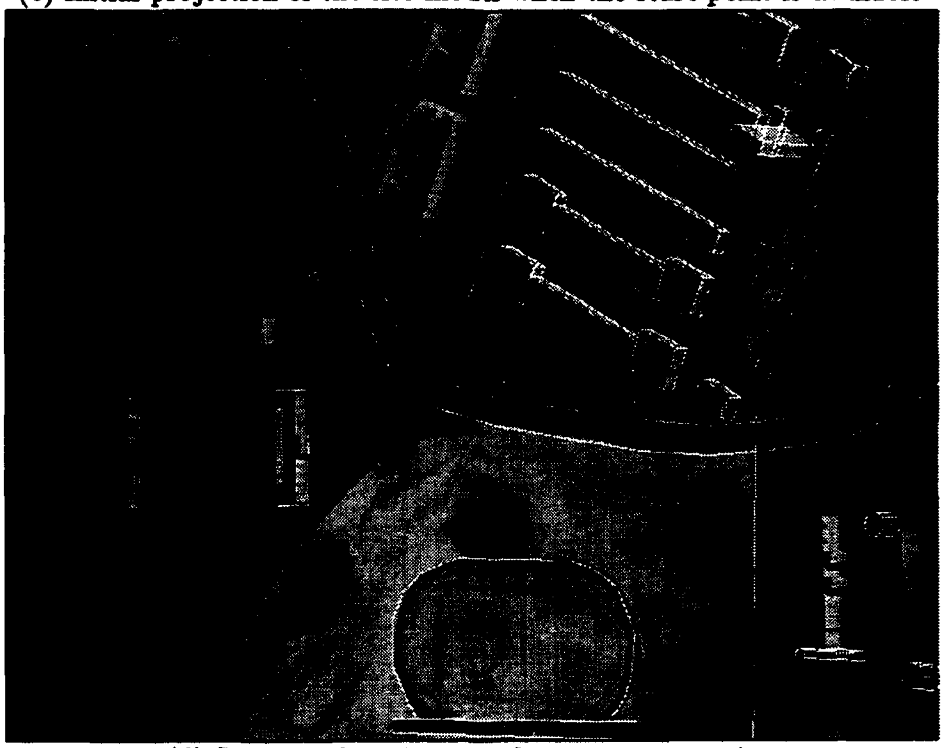


(b) Initial projection of the site model into the new image

Figure 4: Registering a new image to the site model.



(c) Initial projection of the site model when the stare point is available



(d) Site model projection after camera resection

Figure 4: (cont.) Registering a new image to the site model.

For each of the cases shown in (b) and (c), we then manually select five control points, adjust them to the correct positions in the new image domain, and do camera resection to get more accurate camera parameters. Figure 4(d) shows the projection of the site model into the new image after camera resection. The refined camera parameters are: $x_o = 7977$ feet, $y_o = 309$ feet, $z_o = 7776$ feet, $\omega = 3.0°$, $\phi = 42.5°$, $\kappa = -2.6°$. Accordingly, we have $\alpha = 42.6°$, $\beta = 93.3°$, $\gamma = 91.9°$, x_p=839, y_p=719, and r=10563.

The following observations about the given camera parameters can be made from Table 3:

1. The initial camera elevation angles are relatively accurate, with a maximum error of 6.4°.
2. The errors in the camera azimuth angles are relatively large, but except for image M3, which is viewed from the nadir direction ($\alpha = 1.7°$), the errors in camera azimuth angle are within ±38°. The camera azimuth error for M3 is −137.0°.
3. It would be useful to know the camera stare point.
4. Overall, the initial camera parameters obtained by (11–13) and (14–16) give a good initial set of parameters for camera resection.

Control Point Verification

Using the refined camera parameters obtained above, we further evaluated the control points provided with the model board 2 data. It was found that except for point #265, the east intersection of the curved and straight tracks, and some typographical errors, the 3-D coordinates of the control points are quite accurate. The following errors were found and corrected in our experiments.

1. The minus signs of the y coordinates of points #20, #22, and #220 are missing.
2. Point #38 is not marked. It should be added next to point #17.
3. Point #78 is not given; its mark, on building B9, should be removed.
4. The marks for points #231, #234 and #238 are not correct; the control points are located along the junction of the wall and the roof.
5. Point #237 is not correct; it should be on the ground just under the apex of the dormer window.

Based on our camera resection results, we found that the correct y value for point #265 should be about 2.9626 inch.[1] We also computed the correspondences (image domain indices) of the control points on each image:

$$X = \frac{f}{\epsilon}\frac{r_{11}(x-x_o)+r_{12}(y-y_o)+r_{13}(z-z_o)}{r_{31}(x-x_o)+r_{32}(y-y_o)+r_{33}(z-z_o)} \quad (32)$$

$$Y = \frac{f}{\epsilon}\frac{r_{21}(x-x_o)+r_{22}(y-y_o)+r_{23}(z-z_o)}{r_{31}(x-x_o)+r_{32}(y-y_o)+r_{33}(z-z_o)} \quad (33)$$

[1]The y-value for point #265 was not adjusted in computing the correspondences listed in Table 4.

where f is the camera focal length, ϵ is the pixel spacing in image plane, x_o, y_o and z_o are the camera shift parameters obtained from camera resection, and $r_{11}, \ldots, r_{33}$ are computed using either (2) or (7). The results are listed in Table 4. Points which are out of the frame board are printed in Table 4, as dashes, and points which are invisible (blocked by other buildings) are bracketed.

3.3. Region Delineation

Given an image to be exploited, quickly locating the regions of interest according to the tasks in the QL profile, and hence, narrowing the search area and reducing computation and false alarms is an important step in site model supported image monitoring. We have developed two algorithms for quickly delineating regions of interest. When 3-D features for the regions of interest are available from the site model, we can quickly compute the "valid" portion(s) of the regions in the current image domain, and fill the regions of interest with appropriate labels. For regions such as roads and parking lots, we further compute their directions, which are useful for vehicle detection. When the region of interest is available from another image, we use the image-to-image transform equations (60–61) in Section 3.7. to quickly transform the corresponding region to the new image domain. Two examples are presented to illustrate the region delineation step.

In Figure 5 the delineation of regions of interest (roads) from corresponding features stored in the site model is shown: (a) an image to be monitored; (b) regions corresponding to roads in the site model; (c) direction maps for the road region; and (d) shows the image of the region of interest.

In Figure 6 the delineation of the region of interest (a storing garage) from a region map associated with an earlier image is shown: (a) and (b) the earlier image and its region map; (c) the region of the storing garage in the old image; (d) the new image to be exploited; (e) the region of the storing garage in the new image delineated using our algorithm; and (f) shows the image of the region of interest.

3.4. Integration into RCDE

In the past year we have spent a considerable amount of effort on mastering the RCDE. As of the time of preparation of this report, we have made following progress on integration into RCDE:

1. We have successfully installed the latest version of RCDE on all our SPARC-10 work stations, including the latest upgrades from SRI.

2. Using our recently developed image-to-site-model registration algorithm, we have built a site model for model board 2 which includes all forty images.

3. We have added new IU functions into RCDE which can be invoked from augmented menus. Figure 7 shows several new functions that we have added into RCDE:

    - SNF filter,

Table 4: Correspondences from given world coordinates

i	1	2	6	7	8	9	10	11	12	14	15	16	17
x_i	0.0000	0.0000	6.8025	11.9013	6.1086	7.1230	6.7986	11.3535	6.4002	8.7158	11.3535	8.7158	29.5155
y_i	0.0000	0.0000	0.0540	-0.1363	-3.2590	-5.5560	-6.4637	-10.0530	-12.8605	-3.2652	-10.0530	-3.2652	-4.7444
z_i	1.0200	0.0000	1.4430	1.4290	0.7815	0.8120	0.6930	0.0000	0.9250	1.8400	0.8464	0.0000	0.0000
M1	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –
M2	193 211	214 225	203 382	211 521	132 417	77 474	56 479	– –	– –	116 472	– –	154 496	– –
M3	– –	– –	– –	425 27	– –	– –	– –	78 24	– –	– –	77 23	– –	293 650
M4	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –	330 381
M5	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –	316 375
M6	148 906	162 906	358 911	522 922	342 1024	– –	– –	– –	– –	412 1028	– –	435 1026	– –
M7	189 1001	191 990	421 1002	596 1006	– –	– –	– –	– –	– –	– –	– –	– –	– –
M8	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –	265 142
M9	– –	– –	1303 395	1118 387	– –	1298 211	1311 181	1146 54	– –	1238 302	1147 71	(1236 265)	(478 214)
M10	– –	– –	912 892	919 748	1009 909	1071 875	1097 883	1204 750	1273 879	998 828	1195 744	1018 840	1063 264
M11	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –	1211 31
M12	85 84	98 68	145 251	188 368	69 226	24 251	2 242	– –	– –	80 303	– –	103 273	255 728
M13	– –	367 6	357 141	347 260	272 138	209 163	185 159	87 278	14 153	264 183	83 267	272 209	208 713
M14	– –	– –	301 32	298 146	265 19	232 43	221 36	183 137	130 31	247 76	169 136	278 77	254 531
M15	152 667	(149 678)	316 662	438 667	303 741	331 791	324 812	437 899	325 953	369 730	440 890	362 750	854 783
M16	988 208	982 220	884 201	801 197	872 167	843 138	843 128	747 90	815 46	837 153	752 79	826 176	477 150
M17	995 880	997 871	977 768	968 680	1041 770	1082 750	1100 754	1158 665	1221 754	1033 734	1157 671	1037 720	(1015 351)
M18	262 57	272 54	271 186	276 281	222 171	186 189	172 183	(129 263)	67 176	216 223	121 266	233 216	244 592
M19	– –	– –	– –	1305 588	– –	– –	– –	– –	– –	– –	– –	– –	1045 31
M20	– –	– –	– –	1201 600	– –	– –	– –	– –	– –	– –	– –	– –	(846 47)
M21	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –
M22	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –
M23	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –
M24	– –	– –	– –	836 908	– –	– –	– –	1173 914	– –	939 1015	1175 927	938 986	1023 340
M25	– –	– –	820 1033	830 856	– –	– –	– –	1171 902	– –	920 980	1158 906	(950 971)	(1015 239)
M26	– –	– –	95 819	228 842	85 863	113 908	106 921	229 986	101 1033	145 894	224 1002	156 858	695 962
M27	– –	– –	376 61	341 149	293 49	228 56	206 49	84 127	43 12	278 72	83 110	278 109	104 484
M28	373 987	374 977	541 914	670 862	563 994	– –	– –	– –	– –	627 975	– –	627 958	1162 763
M29	1137 10	1153 11	961 22	838 24	– –	– –	– –	– –	– –	– –	– –	– –	– –
M30	1152 172	1152 163	988 182	865 181	1002 101	975 49	982 27	– –	– –	938 111	– –	939 96	(442 79)
M31	237 63	251 63	234 217	232 331	185 198	144 220	130 212	(80 308)	18 199	171 258	68 309	196 255	177 710
M32	88 834	102 843	200 812	290 802	184 887	192 932	184 952	260 1024	– –	215 871	249 1016	(240 888)	(621 867)
M33	973 936	963 929	980 825	986 738	1025 822	1061 799	1074 801	1126 710	1173 790	1037 785	1134 715	1019 774	1058 411
M34	428 37	431 43	427 154	422 245	368 148	325 167	309 162	244 249	191 158	364 188	241 244	369 199	336 575
M35	356 851	342 847	463 822	536 804	445 884	461 922	455 940	(513 982)	– –	497 877	524 985	473 869	758 809
M36	– –	– –	1200 302	1080 300	1190 225	1156 168	1157 146	1026 65	– –	1144 216	1036 56	1120 233	616 202
M37	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –
M38	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –
M39	1231 201	1224 190	1081 224	965 233	1095 145	1074 99	1081 78	974 5	–	1042 163	979 13	(1030 144)	(549 165)
M40	378 281	(374 268)	383 392	381 469	314 383	269 405	250 401	180 471	121 417	319 436	182 483	312 412	295 714

Table 4: Correspondences from given world coordinates (cont.)

i	18		[illegible]		20		21		22		23		24		25		26		28		29		30		31	
x_i	15.0738		15.0074		14.3694		17.7123		17.0713		16.7983		28.0565		19.6578		19.1255		21.4034		23.2277		23.2277		23.3098	
y_i	-11.9847		-11.9847		-4.3103		-11.6342		-5.9860		-3.8242		-0.5890		-11.7448		-9.6141		-10.8230		-9.8979		[illegible]		-4.6716	
z_i	0.0000		1.7365		1.7365		1.0648		1.0517		1.0495		1.7655		1.7269		2.0000		1.7437		0.0000		[illegible]		1.7190	
M1	–	–	–	–	–	–	–	–	–	–	–	–	1080	45	–	–	–	–	–	–	–	–	–	–	1287	49
M2	–	–	–	–	105	640	–	–	84	744	138	708	236	975	–	–	–	–	–	–	22	979	–	–	118	893
M3	17	157	13	154	282	120	31	249	228	218	304	205	435	595	29	318	102	295	64	377	103	439	100	440	284	435
M4	–	–	–	–	–	–	112	17	306	1	–	–	474	306	102	60	176	40	130	110	163	196	158	163	339	167
M5	–	–	–	–	–	–	102	14	–	–	–	–	470	360	96	87	174	76	125	144	150	175	154	203	339	208
M6	–	–	–	–	–	–	–	–	–	–	–	–	1054	954	–	–	–	–	–	–	–	–	–	–	–	–
M7	–	–	–	–	–	–	–	–	–	–	–	–	1152	1020	–	–	–	–	–	–	–	–	–	–	–	–
M8	–	–	–	–	–	–	–	–	–	–	–	–	316	172	–	–	–	–	–	–	–	–	–	–	427	20
M9	–	–	1013	26	1031	266	911	23	933	200	942	266	531	376	839	33	858	105	773	62	707	56	706	90	703	252
M10	1257	640	1240	629	1030	667	1237	559	1083	592	1024	606	934	301	1234	499	1172	518	1208	453	1200	419	1184	405	1042	418
M11	–	–	–	–	–	–	–	–	–	–	–	–	1128	126	–	–	–	–	–	–	–	–	–	–	1293	190
M12	–	–	–	–	109	430	–	–	104	479	151	472	317	729	–	–	29	539	26	584	84	595	63	623	182	625
M13	29	368	21	343	230	319	30	416	184	395	244	386	318	649	20	454	78	435	43	496	(74	562)	66	540	210	536
M14	157	217	127	315	233	201	143	273	221	260	251	254	283	503	130	315	155	304	142	353	(184	392)	155	393	227	397
M15	530	942	[illegible]	924	506	754	597	923	570	798	559	751	819	675	647	918	631	868	687	898	720	895	728	877	718	763
M16	678	64	689	42	740	139	643	53	683	125	699	153	533	182	615	42	636	66	591	53	(556	86)	566	64	591	131
M17	1187	597	1184	611	1040	634	1173	559	1066	579	1026	587	935	396	1169	529	1129	544	1148	499	(1129	456)	1126	468	1025	475
M18	105	330	88	335	209	326	105	382	194	372	228	368	292	577	100	419	130	412	118	451	(152	477)	136	484	219	488
M19	–	–	–	–	–	–	–	–	1320	331	1282	383	1021	151	–	–	–	–	–	–	(1265	92)	1299	86	1193	192
M20	–	–	–	–	1224	440	1305	167	1196	326	1155	387	776	202	1248	122	1212	193	1180	104	1127	67	1110	82	992	214
M21	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
M22	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
M23	–	–	–	–	–	–	–	–	–	–	–	–	451	20	–	–	–	–	–	–	–	–	–	–	–	–
M24	1247	803	1249	832	984	842	1242	737	1045	750	970	756	875	406	1251	687	1176	705	1222	631	(1191	547)	1194	572	1011	563
M25	1238	775	1211	786	960	784	1212	684	1027	690	958	693	852	287	1208	617	1133	631	1179	551	1177	477	1150	483	980	466
M26	324	1033	–	–	293	931	–	–	367	957	359	918	650	917	–	–	–	–	–	–	(531	1027)	–	–	525	971
M27	8	184	8	148	215	169	–	–	153	224	213	229	228	439	–	–	42	227	–	–	(7	342)	6	307	147	331
M28	–	–	–	–	780	936	–	–	866	940	835	892	1081	695	–	–	960	1011	1030	1012	(1063	954)	1065	970	1008	845
M29	–	–	–	–	–	–	–	–	–	–	–	–	459	35	–	–	–	–	–	–	–	–	–	–	–	–
M30	–	–	–	–	800	91	–	–	734	50	743	99	479	186	–	–	–	–	–	–	–	–	–	–	585	90
M31	(49	387)	24	389	156	382	41	447	137	439	174	435	225	690	30	490	62	482	46	529	87	566	63	570	153	579
M32	–	–	–	–	314	880	356	1034	367	915	370	870	585	765	383	1026	378	978	420	1002	(481	993)	458	978	480	865
M33	1158	642	1175	653	1057	685	1166	604	1078	630	1044	641	1008	457	1176	574	1145	591	1163	547	(1133	508)	1150	518	1069	530
M34	207	316	201	306	343	290	209	358	314	344	354	337	410	538	203	390	242	378	220	421	(242	462)	237	454	334	453
M35	(566	999)	587	1007	576	871	616	986	605	887	601	849	762	748	651	982	647	947	674	958	676	926	698	933	699	840
M36	929	17	–	–	1005	192	882	14	924	157	941	211	692	292	843	5	869	56	805	28	(745	69)	765	51	788	185
M37	–	–	–	–	–	–	–	–	–	–	–	–	113	590	–	–	–	–	–	–	–	–	–	–	–	–
M38	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
M39	–	–	–	–	912	154	–	–	846	118	852	163	594	269	790	6	804	54	749	31	696	38	706	55	704	169
M40	145	532	150	554	301	522	158	583	268	557	310	547	379	705	160	620	202	610	179	643	193	642	199	665	299	651

Table 4: Correspondences from given world coordinates (cont.)

i	32	33	34	35	37	38	36	39	41	47	49	52	53
x_i	38.9054	27.2603	26.7440	30.2677	30.2677	29.5155	21.4034	38.9054	30.1444	44.3461	42.3092	38.9054	40.2720
y_i	10.9691	-10.8638	-4.4274	-5.7648	-5.7648	-4.7444	-10.8230	4.3656	-10.0722	4.7606	8.6541	4.3656	8.5080
z_i	0.0000	1.7644	1.7224	1.7608	0.0000	1.7764	0.0000	0.0000	0.0000	1.0948	1.1593	1.1240	1.1324
M1	548 118	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –	566 18
M2	– –	– –	133 988	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –
M3	855 952	72 583	298 555	256 681	259 678	291 653	68 377	625 962	108 680	– –	– –	626 966	773 1007
M4	(875 671)	116 280	342 267	289 371	293 404	326 348	135 143	639 669	143 398	646 820	789 756	638 649	785 693
M5	825 664	112 330	340 317	285 426	(279 397)	322 404	121 117	(603 662)	(130 391)	606 840	743 783	608 681	744 722
M6	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –
M7	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –
M8	95 621	– –	327 73	213 94	(235 131)	242 104	– –	49 471	(209 41)	– –	– –	34 448	21 560
M9	159 674	555 59	577 259	447 217	450 182	475 249	774 27	150 483	451 49	– –	32 630	147 506	104 626
M10	662 67	1210 291	1036 325	1072 225	(1090) 240	1045 249	1226 467	829 45	(1206 230)	– –	– –	817 34	712 13
M11	694 6	– –	1242 107	1229 1	– –	1214 34	– –	– –	– –	– –	– –	– –	– –
M12	696 933	77 708	218 699	218 773	(239 744)	235 758	47 556	542 928	(141 740)	– –	657 1021	530 947	637 979
M13	644 938	33 640	213 621	171 710	179 733	200 690	51 519	(455 944)	59 734	– –	572 1015	451 930	569 962
M14	482 757	141 479	230 472	210 548	(240 547)	224 532	172 353	383 748	181 541	370 871	427 832	365 749	425 786
M15	1025 458	826 898	798 758	883 786	(874 805)	863 764	679 915	1043 592	(882 898)	1170 573	1114 493	1050 580	1069 495
M16	393 363	494 49	535 133	469 113	460 136	487 127	581 75	360 270	(441 81)	271 260	327 314	366 255	363 313
M17	(691 207)	1135 394	1012 413	1029 347	1033 336	1011 363	1150 487	816 195	1116 331	791 101	722 147	813 202	730 184
M18	513 774	127 556	228 550	212 613	(229 605)	227 600	(134 444)	405 767	(161 599)	410 869	470 838	395 773	465 801
M19	526 108	– –	1124 106	– –	– –	1080 24	1316 121	– –	– –	– –	– –	– –	575 12
M20	276 215	– –	894 137	828 18	(848 3)	826 62	1197 88	408 52	– –	– –	219 88	394 62	275 132
M21	264 806	– –	– –	– –	– –	– –	– –	27 805	– –	34 958	177 892	25 785	173 828
M22	316 601	– –	– –	– –	– –	– –	– –	92 606	– –	105 785	239 723	95 625	236 662
M23	162 460	– –	– –	– –	– –	– –	– –	185 268	– –	23 302	59 400	163 252	112 382
M24	478 12	1236 445	1008 453	1062 341	1060 317	1024 365	1220 605	714 21	1211 326	– –	– –	714 35	– –
M25	– –	1185 339	975 342	1021 219	1048 214	986 244	1205 545	– –	1189 228	– –	– –	– –	– –
M26	959 710	– –	615 979	706 1017	714 982	687 996	– –	(951 833)	– –	1095 869	1045 790	947 856	989 784
M27	483 745	– –	131 397	70 458	71 494	103 448	– –	(295 713)	– –	271 807	397 781	296 691	405 738
M28	1223 277	1177 948	1092 801	1196 794	1192 779	1165 778	1028 996	1298 439	1237 882	– –	– –	1300 447	1289 330
M29	259 363	– –	– –	– –	– –	– –	– –	255 185	– –	123 201	167 303	237 185	210 298
M30	255 431	– –	503 98	417 71	(422 57)	437 94	– –	241 288	– –	112 310	168 394	237 298	215 389
M31	459 946	47 656	158 655	136 729	(160 725)	153 715	(71 525)	340 932	(86 715)	– –	402 1021	325 936	400 975
M32	(857 507)	532 989	546 851	609 870	632 887	598 850	443 1018	837 645	615 981	934 608	904 531	823 634	863 542
M33	808 279	1169 445	1068 470	1092 405	1075 396	1076 421	1146 537	(918 264)	1142 387	925 170	859 216	928 270	860 254
M34	632 743	217 528	338 516	311 581	317 589	331 567	226 430	(506 745)	236 589	510 844	585 804	503 741	583 765
M35	879 498	752 933	745 821	791 831	(768 824)	782 816	(652 950)	878 614	767 899	960 592	936 532	893 618	911 541
M36	450 624	663 26	705 192	612 158	592 176	636 184	785 46	(420 446)	575 65	292 450	364 554	433 435	416 548
M37	(616 860)	– –	– –	– –	– –	– –	– –	399 916	– –	– –	593 968	407 898	562 906
M38	1161 722	– –	– –	– –	– –	– –	– –	(1163 961)	– –	1293 947	1231 804	(1138 961)	1174 810
M39	(339 518)	610 45	623 183	540 164	531 145	558 184	739 13	334 382	532 51	210 418	262 494	338 395	310 485
M40	585 791	183 725	306 698	282 750	(276 727)	301 737	173 619	466 811	195 738	(478 893)	548 857	470 825	545 831

Table 4: Correspondences from given world coordinates (cont.)

i	54	55	56	65	66	70	71	73	74	75	76	77	79
x_i	38.9054	42.3186	37.5356	48.0902	48.0902	41.5488	39.7476	33.6162	45.5432	45.5432	44.9778	44.9778	35.5943
y_i	10.9691	10.5488	26.3974	33.2218	33.2218	28.5136	28.3761	16.5866	32.3703	32.3703	37.3366	37.3366	24.6626
z_i	1.1130	1.0787	0.0000	0.0000	0.7535	0.7320	0.7300	0.7009	0.1343	0.0000	0.3773	0.0000	0.6440
M1	554 98	469 24	(267 492)	– –	– –	113 447	166 480	583 334	– –	– –	– –	– –	361 480
M2	– –	– –	1064 912	– –	– –	1133 997	1120 943	747 921	– –	– –	– –	– –	991 868
M3	857 955	– –	– –	– –	– –	– –	– –	1044 761	– –	– –	– –	– –	– –
M4	875 651	858 758	– –	– –	– –	– –	– –	1078 499	– –	– –	– –	– –	– –
M5	830 682	805 782	– –	– –	– –	– –	– –	1035 520	– –	– –	– –	– –	1296 581
M6	– –	– –	– –	– –	– –	– –	– –	1267 347	– –	– –	– –	– –	– –
M7	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –
M8	80 598	– –	(251 972)	– –	– –	– –	– –	290 660	– –	– –	– –	– –	291 886
M9	156 697	35 683	– –	– –	– –	– –	– –	348 847	– –	– –	– –	– –	– –
M10	649 57	– –	(282 151)	– –	– –	227 50	228 95	510 215	– –	– –	– –	– –	314 189
M11	694 7	– –	330 244	20 72	17 73	222 174	250 217	624 213	75 124	75 124	– –	– –	398 271
M12	685 952	702 1022	(1062 915)	– –	– –	1137 1015	1121 977	782 837	– –	– –	– –	– –	998 863
M13	641 924	627 1015	1088 888	– –	– –	1151 982	1146 935	804 787	– –	– –	– –	– –	1036 831
M14	463 758	457 835	(727 748)	840 1012	827 1014	749 847	747 804	558 644	824 949	826 949	906 947	912 946	688 700
M15	1031 445	1108 455	(955 155)	1161 33	1166 24	1039 109	1001 110	897 336	1111 46	1110 48	– –	– –	920 180
M16	399 348	336 342	(500 590)	338 702	342 692	441 614	474 611	521 434	383 685	382 686	423 761	(420 766)	530 554
M17	687 214	686 150	408 259	247 79	243 84	352 195	361 227	599 317	271 125	271 125	180 146	182 144	443 295
M18	504 779	502 840	(774 765)	906 967	901 971	810 845	805 812	594 687	887 920	888 919	974 917	976 915	736 731
M19	547 104	– –	199 491	– –	– –	77 424	118 472	519 376	– –	– –	– –	– –	292 504
M20	261 225	182 134	11 612	– –	– –	– –	– –	291 480	– –	– –	– –	– –	85 623
M21	263 786	246 894	815 764	– –	– –	895 877	889 820	466 631	1037 1014	1037 1016	1217 992	1216 999	753 693
M22	321 620	302 720	837 548	(1037 849)	1042 861	902 677	902 624	517 450	1016 779	1016 777	1181 763	(1179 756)	786 503
M23	139 445	53 456	(141 918)	– –	– –	11 998	59 984	266 583	– –	– –	– –	– –	186 844
M24	477 27	– –	– –	– –	– –	– –	– –	274 190	– –	– –	– –	– –	– –
M25	– –	– –	– –	– –	– –	– –	– –	343 55	– –	– –	– –	– –	– –
M26	955 732	1048 752	(939 398)	1257 294	1254 309	1055 383	1002 379	816 595	1180 305	1180 302	1172 200	(1173 192)	879 439
M27	484 722	451 792	933 790	– –	– –	977 871	982 832	674 649	1071 989	1071 991	1222 996	(1221 1004)	894 729
M28	1225 285	1318 255	– –	– –	– –	– –	– –	1024 204	– –	– –	– –	– –	– –
M29	241 363	169 354	(298 778)	82 956	69 958	202 835	240 832	365 512	131 935	(134 935)	– –	– –	328 732
M30	252 441	172 434	(319 752)	99 894	96 902	231 804	271 801	386 553	153 878	153 876	175 978	177 974	357 722
M31	444 950	438 1025	(752 950)	– –	– –	– –	782 1008	553 842	– –	– –	– –	– –	709 904
M32	843 495	911 492	(877 201)	1103 24	1094 15	952 135	917 145	763 403	1048 50	1050 52	– –	– –	827 235
M33	818 285	827 220	541 342	418 155	424 159	510 275	512 309	717 394	435 203	434 203	345 227	342 225	577 378
M34	630 738	622 803	928 711	1064 910	1064 907	970 783	966 749	738 638	1047 860	1047 861	1143 847	1143 848	893 672
M35	893 502	935 498	(862 225)	995 76	1005 78	923 178	900 185	820 417	965 98	963 98	962 9	957 8	845 264
M36	463 614	371 606	– –	– –	– –	– –	– –	621 765	– –	– –	– –	– –	608 991
M37	625 842	655 954	(1088 692)	– –	– –	1214 785	1186 732	735 640	– –	– –	– –	– –	1015 637
M38	1135 722	1233 735	1116 170	– –	– –	1211 83	1160 92	996 525	– –	– –	– –	– –	1048 235
M39	343 531	263 531	(381 816)	151 975	153 985	295 877	336 869	467 623	207 954	207 952	– –	– –	426 765
M40	590 806	581 850	(860 729)	967 843	971 852	897 784	896 761	691 715	956 815	(956 814)	1043 797	1041 793	834 717

Table 4: Correspondences from given world coordinates (cont.)

i	84	85	87	88	89	90	92	93	94	95	96	97	98
x_i	39.6868	39.6868	36.0484	30.1898	32.2561	30.1898	32.2561	22.7888	22.7888	24.1515	25.0249	25.6880	27.2072
y_i	33.0488	33.0488	20.1998	18.2284	17.7971	18.2284	17.7971	22.4633	22.4633	25.2913	27.1229	28.4460	31.5884
z_i	0.7064	0.0000	0.5545	0.0000	0.6763	0.6150	0.0000	0.0000	0.6720	0.6916	0.6400	0.6745	0.6510
M1	66 592	63 604	441 370	642 451	596 389	645 441	593 400	763 699	767 688	671 727	607 754	561 771	451 816
M2	1262 874	1276 883	864 946	794 805	777 863	782 797	791 872	883 529	869 520	955 518	1013 518	1054 518	1154 516
M3	– –	– –	1174 840	1095 638	1084 711	1096 638	1083 710	1231 373	1233 373	– –	– –	– –	– –
M4	– –	– –	1207 577	(1136 409)	1121 458	1136 397	1121 471	(1283 192)	1284 180	– –	– –	– –	– –
M5	– –	– –	1146 592	(1098 406)	1080 479	1102 416	(1076 468)	1270 182	1275 192	– –	– –	– –	– –
M6	– –	– –	– –	(1159 290)	1222 304	1152 289	1229 305	914 141	906 140	954 40	– –	– –	– –
M7	– –	– –	– –	1225 378	1298 398	1226 384	1297 392	966 237	966 243	1014 146	1044 82	1068 37	– –
M8	– –	– –	243 785	419 663	342 669	411 650	351 683	689 654	681 640	662 729	651 788	641 830	620 933
M9	– –	– –	268 942	(469 879)	396 881	468 892	398 866	(725 996)	725 1011	– –	– –	– –	– –
M10	118 111	(127 117)	424 165	(474 315)	479 254	467 310	(487 260)	(363 520)	355 515	286 487	243 469	212 455	138 424
M11	135 280	137 278	502 200	629 318	612 263	629 320	613 261	622 556	622 558	534 562	477 564	436 565	338 569
M12	1240 979	(1246 966)	891 889	803 751	802 808	797 762	809 796	850 588	843 600	925 631	980 651	1018 667	1112 702
M13	1282 929	1283 938	908 848	(853 707)	839 751	851 699	840 760	971 518	970 509	1050 539	1102 560	1140 575	1230 610
M14	827 811	839 810	617 705	596 566	577 614	586 567	589 614	665 397	654 397	700 430	731 452	752 469	805 508
M15	988 23	984 32	941 268	(814 310)	864 312	817 303	(860 320)	(640 224)	643 216	667 161	683 127	695 102	721 43
M16	501 685	497 694	496 489	(587 466)	552 452	591 458	548 460	740 527	745 518	738 560	733 588	729 608	721 656
M17	275 236	279 232	525 279	581 377	580 344	579 381	583 339	(526 513)	523 518	467 499	432 486	405 477	344 455
M18	888 816	894 812	660 734	623 622	613 663	618 625	618 660	686 488	681 490	732 518	766 535	790 549	848 580
M19	6 583	– –	385 390	538 509	519 440	550 507	506 442	592 806	605 805	514 833	454 852	410 865	305 898
M20	– –	– –	160 507	(355 591)	302 540	347 597	311 533	(459 861)	451 868	360 901	303 922	260 937	162 972
M21	1056 819	1055 832	594 708	(524 540)	509 590	524 529	509 602	671 323	671 310	770 349	834 376	881 395	991 441
M22	1056 618	(1052 607)	635 518	(575 333)	560 407	578 343	(557 396)	733 102	737 112	831 154	892 179	935 199	1038 244
M23	– –	– –	192 711	368 620	299 611	355 612	313 621	559 703	545 694	497 792	468 857	444 903	391 1013
M24	– –	– –	147 103	(216 290)	231 232	215 298	232 223	67 521	65 531	– –	– –	– –	– –
M25	– –	– –	– –	(304 169)	306 100	294 170	318 99	(177 412)	165 414	83 362	32 328	– –	– –
M26	1007 279	1010 265	888 529	725 535	779 565	722 547	782 552	521 422	517 435	556 383	581 347	600 323	644 262
M27	1118 854	1116 868	764 718	(738 602)	716 628	738 589	715 641	892 476	893 463	965 502	1012 528	1047 546	1129 591
M28	– –	– –	1045 84	916 199	975 190	917 203	974 185	678 180	678 184	679 97	679 41	– –	– –
M29	245 958	256 956	317 611	(452 556)	396 545	442 556	(407 545)	621 672	610 672	580 751	563 802	548 838	516 925
M30	282 895	284 887	338 629	(469 579)	419 578	467 585	421 572	(645 664)	643 670	617 730	600 767	587 795	558 859
M31	875 1018	(884 1016)	624 904	593 765	576 813	585 767	585 811	673 602	665 604	720 640	757 664	783 681	846 722
M32	929 55	937 63	823 323	(711 389)	741 384	704 383	749 391	(588 333)	580 326	614 266	637 228	653 200	691 135
M33	427 321	421 318	654 357	(681 458)	695 422	687 461	689 419	607 604	612 608	564 592	532 581	509 572	453 553
M34	1057 746	1057 749	808 683	770 577	761 612	769 574	762 615	849 438	849 435	903 459	938 474	963 485	1024 512
M35	899 100	890 98	850 344	(766 395)	802 399	774 397	(793 397)	665 339	674 341	693 284	704 247	713 220	734 157
M36	– –	– –	574 868	(708 811)	661 796	715 806	653 802	913 915	922 910	902 989	– –	– –	– –
M37	– –	– –	880 688	(738 536)	756 590	743 526	751 601	(778 286)	784 275	888 294	955 307	1004 316	1119 337
M38	– –	– –	1065 393	(920 472)	960 484	906 471	975 485	725 336	710 335	743 235	766 170	782 122	821 10
M39	341 956	338 947	413 699	(542 637)	498 643	545 645	495 635	(707 699)	711 707	681 765	662 801	648 828	615 891
M40	978 747	(974 738)	754 735	(719 656)	713 693	722 664	(710 [illegible])	801 542	805 551	855 562	888 568	911 574	966 586

Table 4: Correspondences from given world coordinates (cont.)

i	100	101	102	107	108	113	114	115	120	121	122	132	135
x_i	30.7266	29.4689	29.4689	23.5984	16.8101	23.5984	21.6740	21.1030	16.8101	14.6612	37.5356	14.6612	12.5749
y_i	34.0511	36.1745	36.1745	38.5563	24.5408	38.5563	34.4459	33.5997	24.5408	34.6592	26.3974	34.6592	34.5680
z_i	0.6714	0.0000	0.6710	0.0000	0.0000	1.0592	1.0621	1.0713	1.0225	0.0000	3.2536	0.6857	0.6688
M1	298 802	285 892	287 881	– –	892 872	– –	551 996	586 987	898 856	– –	280 435	– –	– –
M2	1245 585	1315 525	1302 515	– –	916 330	– –	1202 306	1174 303	895 315	1193 120	997 868	1180 110	1167 54
M3	– –	– –	– –	– –	1295 161	– –	– –	– –	1298 161	– –	– –	– –	– –
M4	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –
M5	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –
M6	– –	– –	– –	– –	(718 70)	– –	– –	– –	707 67	– –	– –	– –	– –
M7	– –	– –	– –	– –	755 169	– –	– –	– –	755 177	– –	– –	– –	– –
M8	– –	– –	– –	– –	900 615	– –	824 913	835 882	888 593	1072 833	206 905	1064 818	1133 784
M9	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –
M10	84 341	41 384	32 379	– –	305 683	– –	58 572	77 585	293 676	57 763	242 123	48 759	48 814
M11	230 514	196 571	195 573	216 745	650 729	214 749	341 743	369 746	649 733	430 912	324 251	429 915	459 965
M12	1202 782	1255 743	1248 755	1276 609	855 451	1265 628	1141 582	1114 569	844 468	1104 398	1030 973	1097 410	1078 359
M13	1304 696	– –	– –	– –	(1028 370)	– –	1307 463	1283 450	1026 356	1308 307	1083 847	1307 297	1303 248
M14	847 595	896 566	884 567	939 424	700 256	921 424	849 375	834 361	683 256	872 204	671 752	861 204	860 153
M15	– –	– –	– –	– –	502 180	– –	– –	– –	506 168	– –	972 117	– –	– –
M16	672 696	(703 737)	708 729	824 771	855 555	831 758	840 692	844 679	862 542	956 705	519 547	961 697	996 694
M17	288 397	257 418	254 423	(234 523)	(506 619)	229 531	309 558	326 566	502 627	(332 670)	394 280	329 675	338 710
M18	896 649	938 623	933 627	975 513	715 376	967 518	890 478	874 467	706 380	893 338	748 782	888 341	884 300
M19	173 859	(137 946)	149 945	– –	(665 1014)	– –	– –	– –	685 1013	– –	259 483	– –	– –
M20	28 947	28 1015	19 1023	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –
M21	1082 545	(1153 520)	1155 508	1228 348	(740 151)	1230 328	1086 271	1056 255	741 132	1083 92	819	1084 79	1079 21
M22	1112 350	1182 300	1187 311	1279 119	– –	1286 135	1152 77	1125 60	– –	– –	854 602	– –	– –
M23	– –	– –	– –	– –	724 730	– –	– –	– –	703 716	– –	72 876	– –	808 1023
M24	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –
M25	– –	– –	– –	– –	114 608	– –	– –	– –	96 612	– –	– –	– –	– –
M26	747 222	715 158	712 171	545 83	354 357	539 104	483 187	467 203	348 377	288 135	925 462	283 148	224 142
M27	1184 673	(1248 671)	1250 657	– –	976 373	– –	1231 488	1210 473	978 352	1260 377	940 725	1262 363	1267 325
M28	– –	– –	– –	– –	501 198	– –	– –	– –	500 204	– –	– –	– –	– –
M29	439 990	– –	– –	– –	760 730	– –	636 1008	649 985	744 731	(817 1017)	244 783	806 1019	856 1017
M30	483 911	517 945	515 952	(651 990)	784 704	649 1002	686 919	697 902	783 714	(846 909)	308 785	845 916	892 913
M31	896 810	947 783	939 785	997 647	713 466	984 650	899 597	881 583	700 469	916 428	710 962	908 429	906 379
M32	763 74	754 46	746 39	(655 26)	488 314	643 15	595 98	582 116	475 303	485 133	838 165	477 126	440 136
M33	409 494	(365 520)	371 524	322 637	(570 718)	331 643	405 666	421 674	579 724	393 788	569 360	399 793	400 830
M34	1073 577	1114 555	1113 552	1157 445	(887 328)	1157 440	1077 406	1060 396	886 323	1078 284	925 697	1078 280	1075 242
M35	781 102	755 63	764 65	674 31	581 318	689 33	662 117	654 135	595 321	547 128	906 234	557 130	526 137
M36	– –	– –	– –	– –	(1070 962)	– –	– –	– –	1083 954	– –	597 1023	– –	– –
M37	1240 421	1281 380	1287 369	(1276 195)	768 100	1286 177	1138 151	1106 141	777 82	– –	1119 640	– –	– –
M38	– –	– –	– –	– –	(574 275)	– –	– –	– –	551 274	– –	1041 164	– –	– –
M39	538 947	563 974	567 983	(690 1000)	838 721	696 1013	737 933	750 916	844 733	(882 903)	395 856	886 912	930 904
M40	1006 627	1042 595	1045 604	1092 509	844 452	1098 522	1027 507	1012 501	850 465	1034 393	876 771	1038 401	1039 371

Table 4: Correspondences from given world coordinates (cont.)

i	136	142	143	147	145	146	150	151	152	153	154	157	158
x_i	9.5818	21.1030	0.1605	4.5016	0.1605	1.3866	9.2771	8.6065	8.5694	10.9738	8.6065	17.7123	16.7983
y_i	37.2152	33.5997	12.0458	14.2713	12.0458	15.4392	20.0131	12.0308	16.9416	12.0266	12.0308	-11.6342	-3.8242
z_i	0.6978	0.0000	0.0000	0.6976	0.9530	0.9604	1.0086	0.8000	0.7526	0.7752	0.0000	0.0000	0.0000
M1	– –	580 1004	– –	– –	– –	– –	1206 903	– –	1285 847	1311 681	– –	– –	– –
M2	– –	1195 318	518 62	576 133	499 48	591 31	740 174	529 273	660 203	538 337	545 284	– –	160 722
M3	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –	34 249	305 205
M4	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –	(115 38)	382 14
M5	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –
M6	– –	– –	177 497	306 421	164 496	205 380	459 225	434 500	437 331	510 501	444 500	– –	– –
M7	– –	– –	184 595	331 526	182 605	221 491	493 334	475 601	471 436	558 600	476 593	– –	– –
M8	1262 804	848 905	1304 78	1183 176	1294 57	1291 150	1086 372	1030 185	1080 296	957 221	1039 203	– –	– –
M9	– –	– –	– –	– –	– –	– –	1192 956	1223 730	1219 866	1140 728	1222 713	911 3	942 245
M10	– –	(91 592)	– –	544 991	– –	– –	398 869	604 872	479 883	606 807	613 877	1248 567	1035 613
M11	– –	371 742	(1175 973)	1064 900	1176 978	1076 992	860 859	1067 771	945 835	1036 713	1066 767	– –	– –
M12	1126 285	1125 550	394 61	481 176	383 76	479 103	665 292	460 276	584 272	482 331	470 263	– –	164 455
M13	– –	1283 465	– –	744 82	– –	776 5	899 182	682 178	816 172	681 234	(684 189)	35 431	248 400
M14	907 78	853 361	– –	– –	– –	– –	611 79	489 67	567 64	489 122	(503 68)	161 274	269 254
M15	– –	581 10	138 424	239 372	141 413	166 344	343 254	339 418	330 318	393 419	336 427	593 934	(555 762)
M16	1064 733	837 693	(1052 378)	1001 400	1058 366	1060 412	961 476	922 368	952 436	883 368	(916 378)	637 67	693 166
M17	301 763	(330 559)	778 883	724 819	775 891	711 874	606 749	753 748	665 755	747 708	(756 742)	1174 551	(1028 579)
M18	929 242	882 462	474 56	513 141	466 59	526 82	617 234	480 219	564 219	483 264	(487 217)	(115 378)	238 364
M19	– –	– –	– –	– –	– –	– –	– –	1106 942	– –	1061 881	(1090 943)	– –	1260 386
M20	– –	– –	– –	– –	– –	– –	– –	1029 962	– –	968 905	1038 952	1315 157	1165 376
M21	– –	1054 275	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –
M22	– –	1118 43	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –
M23	– –	– –	1252 211	1101 304	1234 196	1189 314	938 518	982 264	972 419	912 282	997 276	– –	– –
M24	– –	– –	– –	– –	– –	– –	146 957	418 983	251 978	420 911	419 971	1240 721	970 740
M25	– –	– –	– –	– –	– –	– –	224 878	464 926	319 911	466 845	478 922	1229 680	(974 689)
M26	135 73	473 182	– –	22 530	– –	– –	143 440	134 591	128 494	198 599	(139 575)	– –	365 898
M27	– –	1207 494	(721 25)	759 95	723 6	805 41	889 196	680 155	810 176	668 197	(679 171)	–	213 250
M28	– –	– –	(234 691)	314 595	233 699	222 604	364 403	444 603	384 484	503 576	(444 597)	– –	834 882
M29	– –	666 983	1156 365	1038 433	1142 365	1113 466	920 601	935 371	940 513	878 373	947 371	– –	– –
M30	962 966	699 891	1158 430	1057 487	1157 439	1131 513	951 615	958 442	964 547	903 443	959 435	– –	745 90
M31	960 310	895 580	471 70	504 172	458 71	522 101	611 288	462 266	555 268	462 320	472 264	(55 445)	189 433
M32	398 99	(595 128)	154 606	227 544	141 597	175 528	327 416	288 574	306 479	330 567	298 582	– –	(384 880)
M33	354 892	412 668	(773 966)	744 905	781 973	726 963	654 843	784 828	702 844	784 787	(776 822)	1155 598	1034 635
M34	1124 187	1061 402	(652 39)	692 110	650 33	713 52	800 189	651 183	742 180	651 226	(652 188)	212 363	357 343
M35	481 92	640 132	339 615	413 558	351 619	369 548	486 432	476 586	474 492	511 578	466 583	(602 981)	588 845
M36	– –	– –	– –	1316 670	– –	– –	1239 822	1211 615	1237 743	1155 617	(1200 622)	869 25	928 222
M37	– –	(1096 159)	– –	– –	– –	– –	– –	– –	– –	315 6	– –	– –	– –
M38	– –	– –	(183 709)	266 633	161 709	188 597	369 437	361 704	359 541	418 703	(379 704)	– –	– –
M39	993 943	(744 903)	1209 434	1117 497	1215 445	1185 515	1010 626	1028 465	1026 561	976 471	1023 456	– –	(845 152)
M40	1093 320	1006 488	615 236	660 307	619 249	686 258	770 367	616 377	710 362	615 412	612 367	154 569	306 533

Table 4: Correspondences from given world coordinates (cont.)

i	160	161	162	163	201	202	205	209	210	211	213	214	217
x_i	23.3098	27.2603	33.6162	26.7440	6.0049	6.1086	4.5016	9.2771	17.8310	17.8310	22.9810	32.1439	14.4702
y_i	-4.6716	-10.8638	16.5866	-4.4274	-9.7407	-3.2590	14.2713	20.0131	-1.3484	1.3484	-0.3870	-1.6528	-4.8741
z_i	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.8077	0.0000	0.0000	0.0000	2.0486
M1	1274 80	– –	580 346	1176 20	– –	– –	– –	1199 919	– –	– –	1204 178	– –	– –
M2	155 914	– –	762 930	170 1009	– –	148 427	590 143	760 189	188 694	226 717	265 849	– –	85 646
M3	286 434	76 581	1043 759	300 553	– –	– –	– –	– –	392 237	393 237	435 415	405 737	262 124
M4	343 200	121 313	1078 512	346 300	– –	– –	– –	– –	462 9	(465 44)	491 192	434 461	– –
M5	333 181	108 301	1031 509	334 289	– –	– –	– –	– –	471 37	464 9	(484 173)	417 457	– –
M6	– –	– –	1274 349	– –	– –	352 1023	315 423	472 227	711 970	732 969	903 941	1210 995	– –
M7	– –	– –	– –	– –	– –	– –	332 520	494 325	– –	800 1028	976 995	(1289 1033)	– –
M8	448 56	– –	299 675	348 110	– –	– –	1190 192	1097 394	616 9	637 48	491 142	(209 244)	– –
M9	704 218	557 24	(350 833)	579 224	– –	– –	– –	1191 934	903 357	903 320	717 347	385 307	1028 255
M10	1060 432	1227 306	518 221	1054 340	1195 903	1017 914	553 995	(411 875)	950 578	969 591	(946 453)	982 203	1042 660
M11	1290 187	– –	625 212	1239 105	– –	– –	(1064 896)	860 854	1282 371	(1278 366)	1187 252	1099 7	– –
M12	203 597	(98 679)	790 825	238 670	– –	78 214	489 164	676 275	209 507	231 478	298 590	348 785	94 437
M13	(217 559)	42 662	806 796	220 643	(102 154)	275 149	(746 92)	901 196	307 398	(314 422)	336 546	291 776	213 318
M14	256 396	171 478	569 644	259 471	(188 21)	278 20	– –	628 80	273 277	(304 277)	317 391	(298 591)	220 203
M15	710 781	817 916	(893 344)	[illegible]89 776	307 892	300 750	237 380	(339 266)	582 690	574 709	(692 689)	907 717	510 763
M16	582 153	485 72	517 443	526 155	(833 96)	867 177	(996 408)	954 489	700 175	(690 198)	610 209	448 190	737 128
M17	(1029 463)	1138 382	601 313	(1016 402)	1164 758	(1043 764)	726 814	810 741	975 578	979 565	948 475	949 308	1050 633
M18	235 481	(144 548)	600 683	244 543	(125 166)	229 168	519 139	625 231	262 391	279 384	303 478	(297 641)	198 329
M19	1158 198	– –	506 378	1090 112	– –	– –	– –	– –	1228 407	(1192 411)	1077 296	931 25	– –
M20	(1010 198)	– –	(300 473)	913 121	– –	– –	– –	– –	1065 431	1083 412	926 313	710 62	1231 427
M21	– –	– –	466 644	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –
M22	– –	– –	514 438	– –	– –	– –	– –	– –	– –	– –	– –	– –	– –
M23	– –	– –	281 593	– –	– –	– –	1115 315	958 533	– –	– –	622 13	(381 47)	– –
M24	(1010 538)	1233 420	276 180	1007 428	– –	– –	– –	149 941	887 732	(886 705)	860 543	918 251	1003 844
M25	(1007 460)	1213 334	(355 54)	1002 337	– –	– –	– –	242 874	867 652	896 645	870 461	917 139	973 783
M26	534 937	– –	819 581	623 946	(94 961)	89 848	(26 517)	149 421	381 893	(391 858)	525 860	(766 917)	294 947
M27	(148 366)	– –	673 663	132 432	(127 36)	293 64	(758 108)	888 216	272 243	(272 279)	266 379	171 549	199 162
M28	1006 830	1174 932	1023 199	1089 787	– –	563 986	(315 589)	365 396	832 828	(831 813)	949 733	1194 660	789 951
M29	– –	– –	376 512	– –	– –	– –	1049 433	(936 601)	– –	– –	602 35	(397 13)	– –
M30	589 76	– –	(388 547)	508 84	– –	1002 94	1057 481	952 605	721 162	724 147	(604 172)	386 151	796 81
M31	176 575	(72 652)	563 840	182 650	83 191	195 198	514 172	624 286	208 462	233 459	251 574	(232 772)	142 385
M32	(502 881)	555 1005	(771 410)	(569 867)	166 1029	(194 894)	236 551	340 426	388 808	413 825	512 792	683 793	310 888
M33	1052 520	1152 436	710 391	1051 461	(1117 800)	1018 818	(738 900)	645 836	1013 635	(996 624)	984 537	1011 372	1069 684
M34	(339 461)	223 536	739 641	343 524	(251 154)	370 152	(693 114)	801 195	398 [illegible]	(403 360)	419 453	394 622	332 291
M35	677 833	(730 925)	811 415	722 814	436 1001	435 881	404 556	473 428	625 803	601 796	671 758	(792 744)	582 883
M36	(768 202)	643 45	613 771	685 209	(1150 72)	1180 233	1307 675	1226 830	938 268	(916 285)	796 312	564 283	1003 175
M37	– –	– –	(730 652)	– –	– –	– –	– –	– –	– –	– –	49 463	116 755	– –
M38	– –	– –	1013 526	– –	– –	– –	(282 633)	393 436	– –	– –	– –	– –	– –
M39	694 151	601 27	(463 615)	(614 165)	– –	(1089 137)	1112 489	1004 614	832 227	821 207	702 241	488 238	912 146
M40	293 628	178 701	688 706	299 675	181 390	311 372	657 298	(765 354)	361 564	354 541	(374 611)	354 740	292 529

Table 4: Correspondences from given world coordinates (cont.)

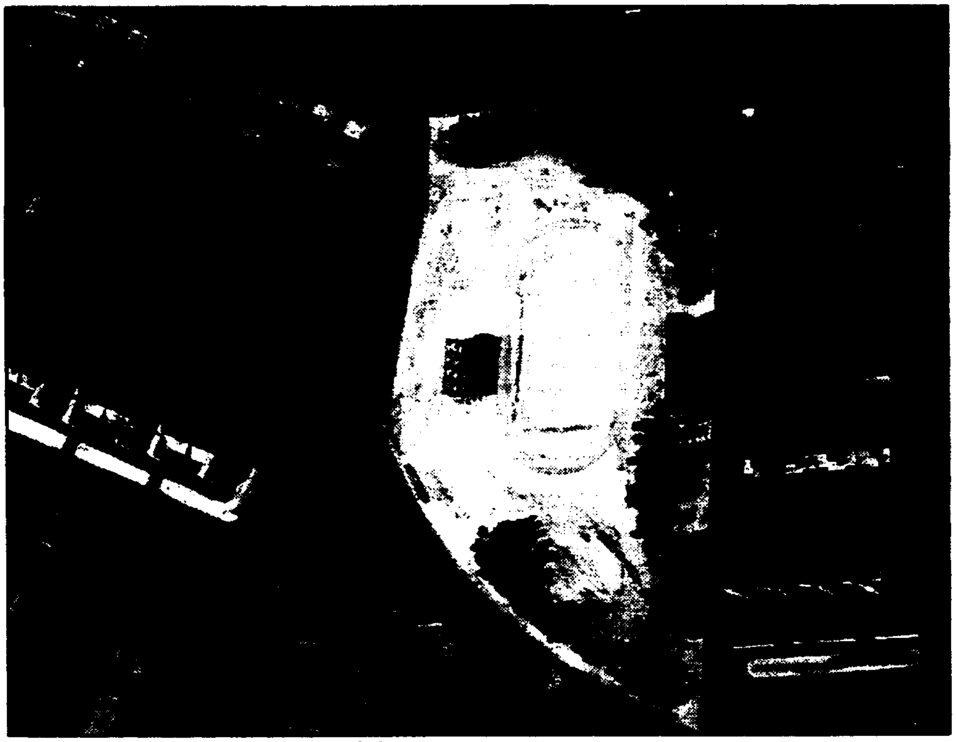
i	220	221	222	224	225	226	227	229	230	231	234	235	237
x_i	26.8210	29.4949	28.6772	19.7810	6.0049	22.0448	22.0448	43.1951	43.1951	24.6340	20.2364	36.0484	20.9319
y_i	-6.9744	-9.4680	-1.0188	-1.0234	-9.7407	20.9441	20.9441	34.5410	34.5410	38.0176	28.9903	20.1998	31.1615
z_i	1.9669	2.0311	2.0427	1.8190	0.6930	0.6900	0.0000	0.5663	0.0000	1.3138	1.2112	0.0000	0.0000
M1	– –	– –	1073 19	1317 194	– –	820 667	815 [illegible]	– –	– –	390 1019	709 889	438 380	637 948
M2	62 1020	– –	220 995	201 744	– –	823 520	83[illegible] [illegible]	– –	– –	1319 335	1033 344	876 953	1123 350
M3	208 562	125 660	420 618	406 305	– –	1179 350	117[illegible]	– –	– –	– –	– –	1172 838	– –
M4	252 264	159 341	458 319	471 64	– –	1231 157	(123[illegible]	– –	– –	– –	– –	1206 587	– –
M5	251 322	158 405	455 384	477 100	– –	1226 168	1222 157	– –	– –	– –	– –	1142 582	– –
M6	– –	– –	1072 970	775 961	– –	880 193	(888 195)	– –	– –	– –	– –	– –	– –
M7	– –	– –	– –	– –	– –	940 295	940 289	– –	– –	– –	875 24	– –	– –
M8	303 16	206 1	291 166	560 43	– –	691 592	699 607	– –	– –	– –	817 750	251 797	830 840
M9	573 185	473 108	508 368	832 366	– –	750 970	751 [illegible]	– –	– –	– –	– –	269 931	– –
M10	1102 313	1169 232	943 280	942 525	1188 898	391 531	400 [illegible]	89 30	(96 35)	– –	185 595	431 169	149 591
M11	1306 72	– –	1131 105	1248 327	– –	669 557	670 [illegible]	51 212	53 211	213 718	494 708	503 199	433 714
M12	158 704	125 760	310 746	234 551	– –	798 583	806 57[illegible]	– –	– –	1256 656	985 551	897 879	1059 546
M13	141 622	67 691	304 662	314 445	99 144	927 492	(929 501)	– –	– –	– –	1152 432	909 855	1215 463
M14	191 472	155 528	273 517	277 320	176 21	629 379	641 378	854 898	864 897	907 450	753 338	626 704	811 356
M15	807 810	876 864	836 681	627 683	310 885	629 245	626 253	– –	1054 6	– –	575 82	938 274	582 56
M16	522 96	465 62	522 173	670 178	837 87	749 496	(744 504)	442 712	439 720	811 747	832 607	(493 496)	825 655
M17	1061 409	1102 357	941 386	964 544	1163 763	553 529	(555 524)	237 175	239 171	234 514	411 576	527 275	374 558
M18	186 551	150 597	283 589	271 427	118 168	653 476	659 473	920 882	924 879	957 539	788 448	(665 731)	838 457
M19	1180 50	– –	1024 124	1186 362	– –	654 790	(641 791)	– –	– –	– –	519 1026	375 391	– –
M20	945 72	– –	765 179	1005 392	– –	500 851	508 843	– –	– –	– –	– –	167 502	– –
M21	– –	– –	– –	– –	– –	618 288	(618 301)	1114 932	1113 942	1214 353	896 226	594 718	970 270
M22	– –	– –	– –	– –	– –	686 90	682 79	1096 717	(1092 707)	1267 171	970 37	632 509	1035 39
M23	– –	– –	430 8	– –	– –	571 641	585 650	– –	– –	– –	587 879	203 718	585 970
M24	1098 457	1191 375	891 391	878 671	– –	118 556	(119 546)	– –	– –	– –	– –	148 96	– –
M25	1054 346	1137 255	861 266	858 582	– –	209 443	221 441	– –	– –	– –	– –	– –	– –
M26	615 1029	– –	665 933	432 895	90 974	496 464	500 450	1113 258	1115 246	568 124	442 301	(890 518)	468 234
M27	60 382	– –	212 443	269 280	126 22	855 442	(854 455)	1147 939	1145 951	– –	1086 433	763 730	1140 480
M28	1123 863	1218 893	1102 701	878 799	– –	677 230	677 226	– –	– –	– –	533 54	1044 81	– –
M29	– –	– –	440 24	646 10	– –	626 630	637 629	174 995	184 994	– –	664 856	326 611	668 915
M30	496 42	– –	463 179	675 171	– –	658 639	659 632	208 924	210 918	624 994	710 809	340 624	700 841
M31	111 653	68 707	214 703	214 506	74 192	635 585	644 583	– –	– –	969 675	786 556	631 902	845 572
M32	535 903	577 950	591 769	426 796	156 1022	561 357	(569 365)	1003 13	1010 20	657 17	549 204	830 329	(584 174)
M33	1110 464	1152 411	1018 446	1010 602	1123 804	638 617	(632 613)	399 256	394 254	343 624	503 677	649 355	455 663
M34	289 517	241 567	401 548	403 386	249 150	820 422	(820 425)	1088 813	1088 815	1147 458	972 382	809 685	(1014 400)
M35	749 867	785 900	774 754	652 789	445 1004	665 372	655 370	942 64	934 63	706 42	645 226	842 342	638 180
M36	694 124	617 59	677 278	893 277	1159 65	933 867	(924 873)	– –	– –	– –	– –	567 873	– –
M37	– –	– –	108 608	1 340	– –	727 264	722 276	– –	– –	1286 206	957 146	875 697	(1021 171)
M38	– –	– –	– –	– –	– –	691 388	707 389	– –	– –	– –	630 115	1078 394	(675 41)
M39	622 131	558 84	580 265	787 239	1102 6	727 675	724 667	263 992	261 985	675 1010	770 829	410 693	(748 857)
M40	258 710	213 754	372 718	368 591	183 399	777 545	774 536	999 785	(996 778)	1088 541	930 504	751 728	962 492

Table 4: Correspondences from given world coordinates (cont.)

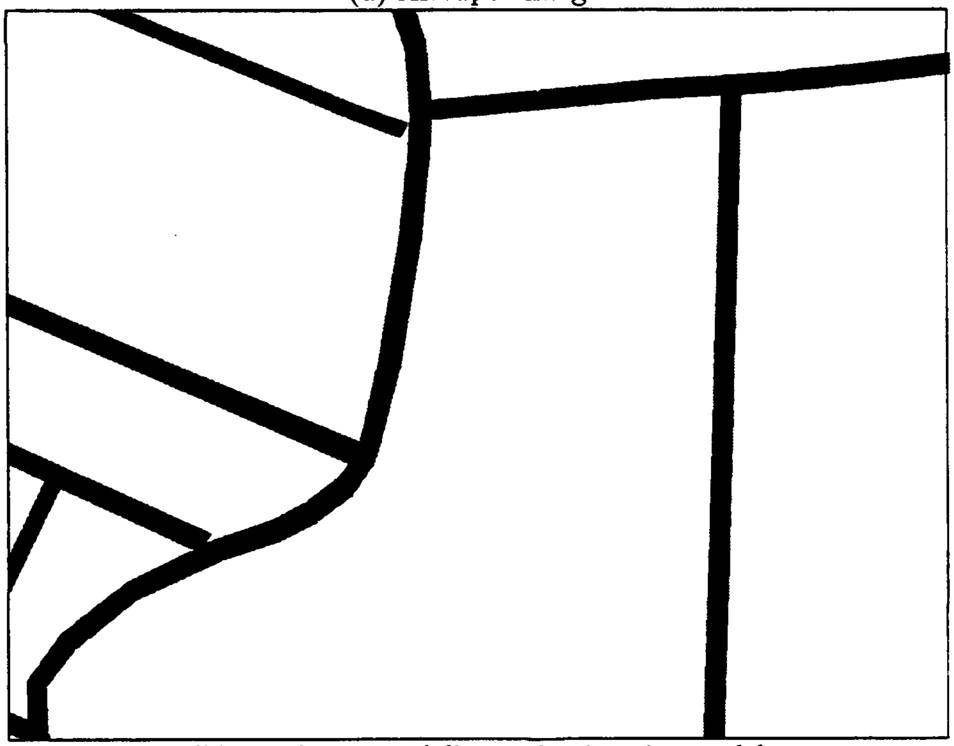
i	238	241	243	244	245	250	251	252	253	254	255	256	257
x_i	19.8450	45.9329	35.7686	36.4857	17.7493	4.4465	5.2451	6.6867	11.3680	12.0491	18.3980	13.7511	18.4662
y_i	28.0938	35.1289	13.7698	13.8304	17.2859	35.1327	35.2569	39.8925	38.4688	38.0839	35.0307	25.9135	16.9150
z_i	1.2367	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
M1	739 875	– –	578 241	557 228	1010 680	– –	– –	– –	– –	– –	– –	952 969	997 657
M2	1005 346	– –	691 1033	– –	719 460	– –	– –	– –	– –	– –	1223 220	941 225	711 486
M3	– –	– –	948 839	951 863	1043 205	– –	– –	– –	– –	– –	– –	– –	1031 231
M4	– –	– –	977 576	979 598	1104 47	– –	– –	– –	– –	– –	– –	– –	1091 67
M5	– –	– –	929 572	928 593	1113 20	– –	– –	– –	– –	– –	– –	– –	1098 42
M6	– –	– –	– –	– –	744 322	– –	– –	– –	– –	– –	– –	620 24	767 335
M7	861 56	– –	– –	– –	791 414	– –	– –	– –	– –	– –	– –	647 123	816 426
M8	821 721	– –	211 640	189 652	802 457	– –	– –	– –	1223 879	1196 880	952 900	1012 602	776 459
M9	– –	– –	272 753	247 755	900 857	– –	– –	– –	– –	– –	– –	– –	875 846
M10	206 603	– –	589 157	588 138	487 641	– –	30 1015	– –	– –	– –	53 666	268 768	497 621
M11	521 706	2 151	666 121	655 104	816 612	– –	– –	– –	– –	380 1019	371 826	656 822	816 590
M12	958 543	– –	739 869	746 884	679 474	1036 147	1046 167	1185 200	1182 317	1176 333	1142 487	866 379	676 490
M13	1126 423	– –	725 854	727 872	827 400	1312 66	1316 84	– –	– –	– –	– –	1065 295	816 418
M14	738 329	873 964	525 690	526 706	583 277	– –	– –	967 6	941 123	934 139	878 295	724 183	577 294
M15	568 99	– –	948 401	964 400	537 324	– –	– –	– –	– –	– –	– –	431 152	554 332
M16	833 593	390 731	464 403	451 403	796 451	1132 707	1119 709	1126 780	1037 762	1023 756	893 713	916 574	782 446
M17	428 581	219 123	647 269	644 256	635 593	356 840	352 827	266 809	276 730	280 718	314 607	491 673	639 580
M18	772 440	938 930	555 720	557 733	590 390	890 138	893 154	981 183	960 275	953 288	905 411	735 318	584 404
M19	548 1015	– –	526 256	510 238	805 822	– –	– –	– –	– –	– –	– –	– –	799 795
M20	– –	– –	300 356	281 340	692 859	– –	– –	– –	– –	– –	– –	– –	681 833
M21	865 215	1138 1029	365 709	367 731	492 177	– –	– –	– –	1207 1	1195 20	1100 198	785 66	479 198
M22	940 25	1104 786	415 505	416 527	– –	– –	– –	– –	– –	– –	– –	– –	– –
M23	600 847	– –	234 522	215 529	718 507	1070 1001	1045 1010	– –	– –	– –	– –	810 754	698 500
M24	– –	– –	377 113	375 89	244 685	– –	– –	– –	– –	– –	– –	– –	257 663
M25	– –	– –	– –	– –	326 596	– –	– –	– –	– –	– –	– –	74 707	338 573
M26	431 319	1197 244	876 644	896 646	383 507	– –	18 87	– –	190 39	210 50	395 141	268 317	402 517
M27	1063 421	1151 1014	582 693	579 708	774 358	1309 194	1310 209	– –	– –	– –	1257 449	1026 322	761 370
M28	534 81	– –	1111 244	1128 234	611 365	57 78	75 66	– –	– –	– –	– –	406 199	634 366
M29	672 831	127 1008	328 437	313 439	733 525	– –	– –	– –	– –	– –	729 1025	833 769	717 515
M30	718 791	151 931	333 489	317 490	753 553	1077 914	1059 917	1031 1011	924 984	908 977	762 918	856 731	736 545
M31	768 545	– –	510 882	512 899	573 479	926 183	928 202	1025 241	995 353	987 369	924 518	740 396	566 495
M32	539 223	– –	804 460	818 456	479 450	313 163	327 158	367 69	441 77	452 81	552 112	439 299	491 455
M33	519 682	383 202	759 344	758 331	693 679	385 971	383 958	301 948	326 860	333 846	386 721	546 776	699 665
M34	955 375	1101 867	685 683	687 696	751 349	1083 100	1086 114	1174 137	1149 222	1142 235	1087 352	913 272	744 362
M35	640 244	969 47	839 458	848 455	596 452	394 145	406 140	426 43	498 61	508 67	601 111	537 300	606 457
M36	– –	– –	544 697	526 699	1010 769	– –	– –	– –	– –	– –	– –	1151 993	991 760
M37	925 141	– –	667 740	678 761	555 178	– –	– –	– –	– –	– –	1105 73	773 4	552 201
M38	620 146	– –	1072 624	1092 621	600 520	– –	– –	– –	– –	– –	– –	499 234	618 531
M39	778 812	201 1004	413 565	396 568	819 584	1098 883	1081 887	1048 975	950 963	936 958	802 921	904 739	803 579
M40	914 501	1002 811	636 743	637 752	708 486	1058 239	1059 251	1144 261	1109 334	1101 345	1036 446	873 404	701 497

Table 4: Correspondences from given world coordinates (cont.)

i	258		262		264		265		266		267		310		311	
x_i	46.3372		1.8304		17.3288		27.6470		35.0110		35.0460		10.1781		10.1781	
y_i	35.9346		-1.5658		3.0217		3.9064		-3.0873		-2.3005		16.1336		16.1336	
z_i	0.0000		0.0000		0.0000		0.0000		0.0000		0.0000		1.1418		0.7570	
M1	–	–	–	–	1296	362	995	182	–	–	–	–	1257	788	1254	795
M2	–	–	180	293	337	645	392	925	–	–	–	–	637	252	644	258
M3	–	–	–	–	545	213	592	571	360	839	387	839	–	–	–	–
M4	–	–	–	–	615	31	635	329	379	547	407	548	–	–	–	–
M5	–	–	–	–	–	–	620	321	361	543	387	544	–	–	–	–
M6	–	–	218	961	719	817	1062	794	–	–	1309	1022	483	358	488	359
M7	–	–	–	–	781	886	1135	854	–	–	–	–	527	466	527	463
M8	–	–	–	–	689	133	384	300	114	254	119	271	1017	293	1021	302
M9	–	–	–	–	920	450	550	473	280	263	279	286	1164	852	1164	843
M10	–	–	–	–	853	616	835	340	1021	122	1000	124	496	835	(501	838)
M11	–	–	–	–	1176	436	1019	193	–	–	–	–	945	787	944	785
M12	–	–	78	113	329	466	438	692	339	843	358	845	573	317	577	311
M13	–	–	324	49	434	406	451	657	248	850	270	850	793	205	794	211
M14	887	976	–	–	367	267	378	497	277	653	288	654	547	101	554	102
M15	–	–	195	712	554	616	790	600	976	747	974	731	370	331	(368	335)
M16	387	744	944	200	722	256	554	266	392	170	395	180	923	420	920	425
M17	203	117	1022	839	899	580	855	399	969	254	954	254	673	729	675	726
M18	953	938	249	89	350	377	380	565	279	691	291	692	549	251	553	250
M19	–	–	–	–	1112	518	900	263	–	–	–	–	997	994	990	994
M20	–	–	–	–	1002	531	712	307	–	–	–	–	898	1023	(902	1019)
M21	–	–	–	–	4	164	23	463	–	–	–	–	–	–	–	–
M22	1130	796	–	–	–	–	84	264	–	–	–	–	–	–	–	–
M23	–	–	–	–	768	69	483	174	311	28	307	50	918	399	926	405
M24	–	–	–	–	735	715	717	389	973	159	946	157	278	936	279	930
M25	–	–	–	–	759	650	738	287	966	39	941	36	337	861	343	859
M26	1211	228	–	–	376	777	650	799	840	953	842	939	170	523	173	516
M27	1173	1027	–	–	392	289	354	487	112	599	134	604	782	193	781	201
M28	–	–	438	994	768	715	1017	578	1282	663	1274	643	434	489	434	486
M29	119	1029	–	–	734	123	500	160	–	–	–	–	894	490	900	490
M30	144	947	1107	129	743	244	502	270	316	121	317	138	926	534	926	531
M31	–	–	224	103	310	453	328	683	207	832	221	834	535	306	540	305
M32	–	–	127	870	420	737	615	689	735	816	738	799	326	486	331	490
M33	368	195	988	894	926	645	918	466	1036	317	1023	319	720	816	716	813
M34	1117	875	402	76	484	349	499	536	366	676	381	676	727	207	727	209
M35	974	31	371	868	593	718	733	663	829	758	829	744	503	504	498	502
M36	–	–	1286	273	949	397	702	426	486	246	488	267	1200	721	1195	724
M37	–	–	–	–	95	268	246	571	102	857	129	851	–	–	–	–
M38	–	–	–	–	594	1006	858	976	–	–	–	–	389	566	398	566
M39	193	1020	1185	162	832	297	595	342	420	215	420	232	994	554	991	550
M40	1016	813	343	301	438	521	456	663	329	782	343	780	696	393	(694	388)

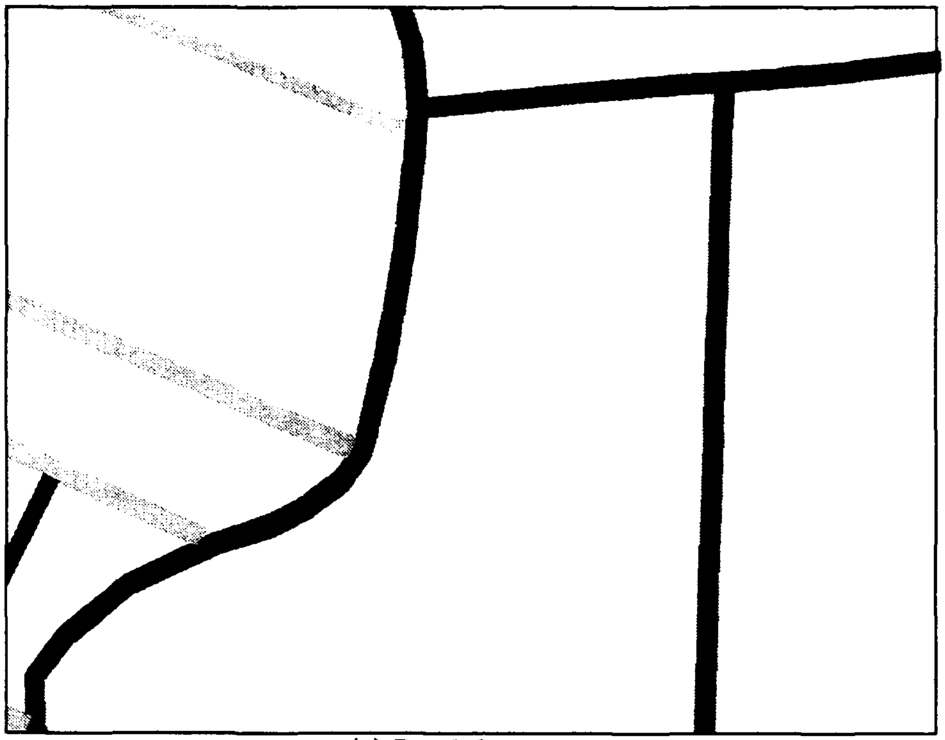


(a) An input image

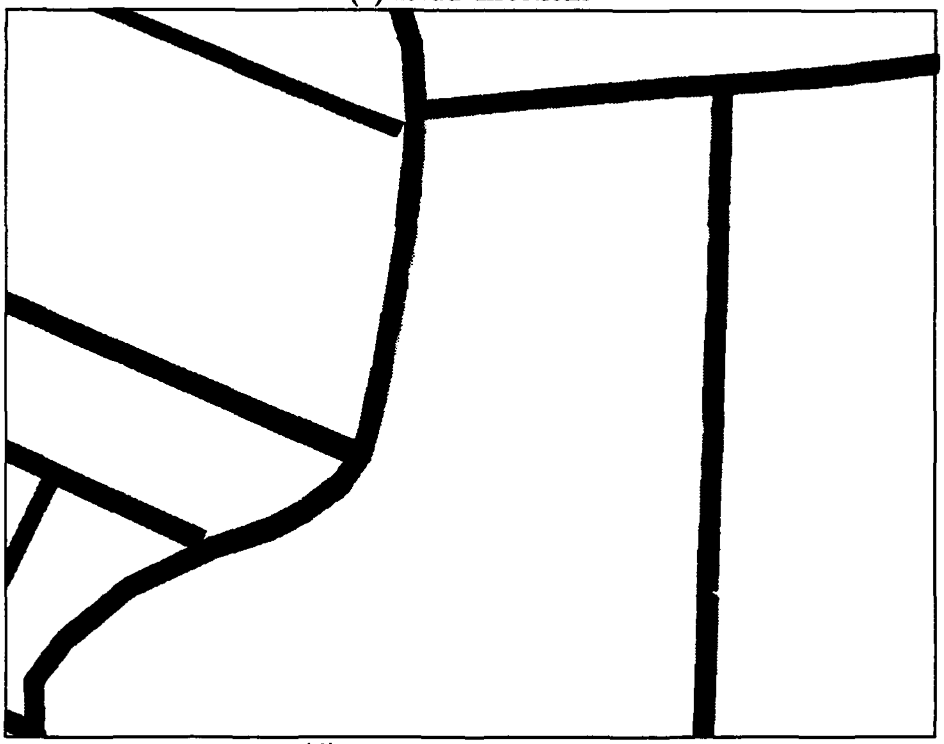


(b) Road regions delineated using site model

Figure 5: Region delineation using a site model

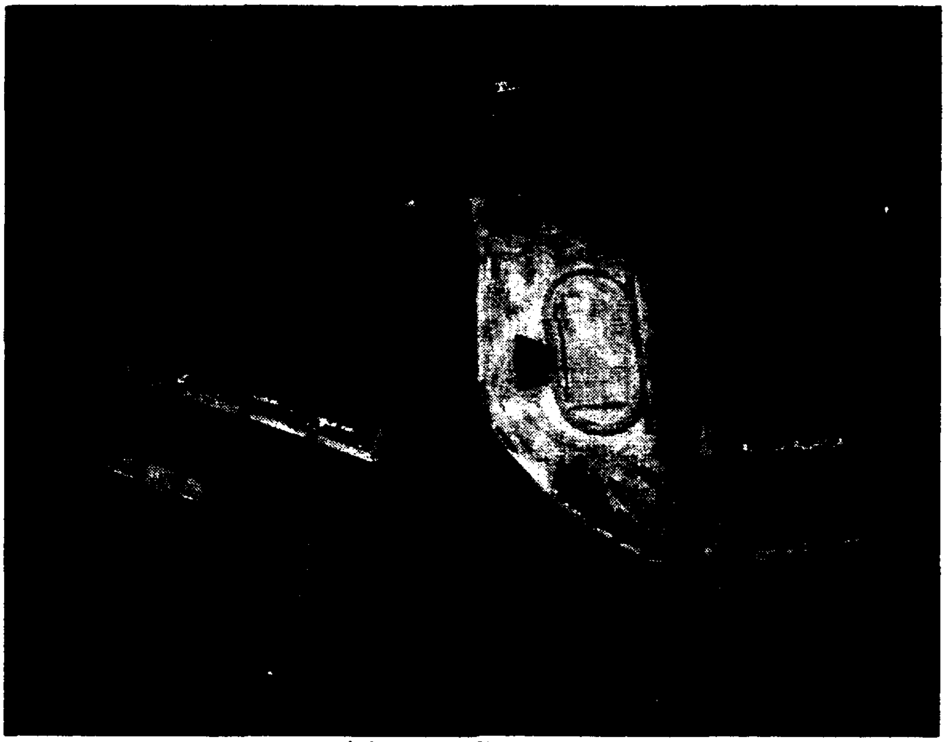


(c) Road directions

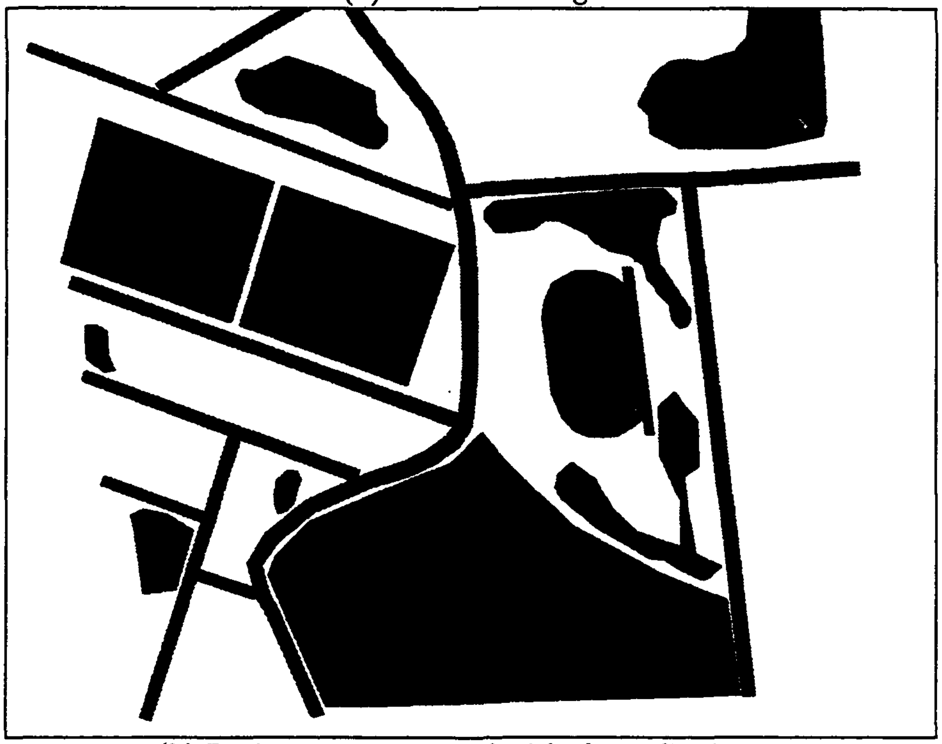


(d) Image to be monitored

Figure 5: (cont.) Region delineation using a site model

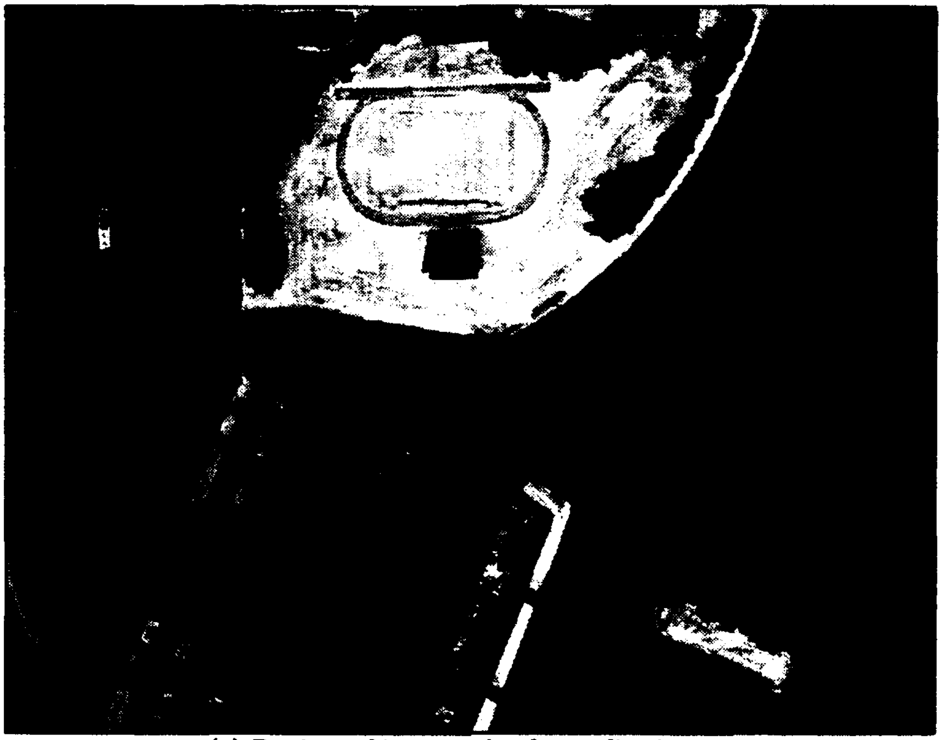


(a) An earlier image



(b) Region map associated with the earlier image

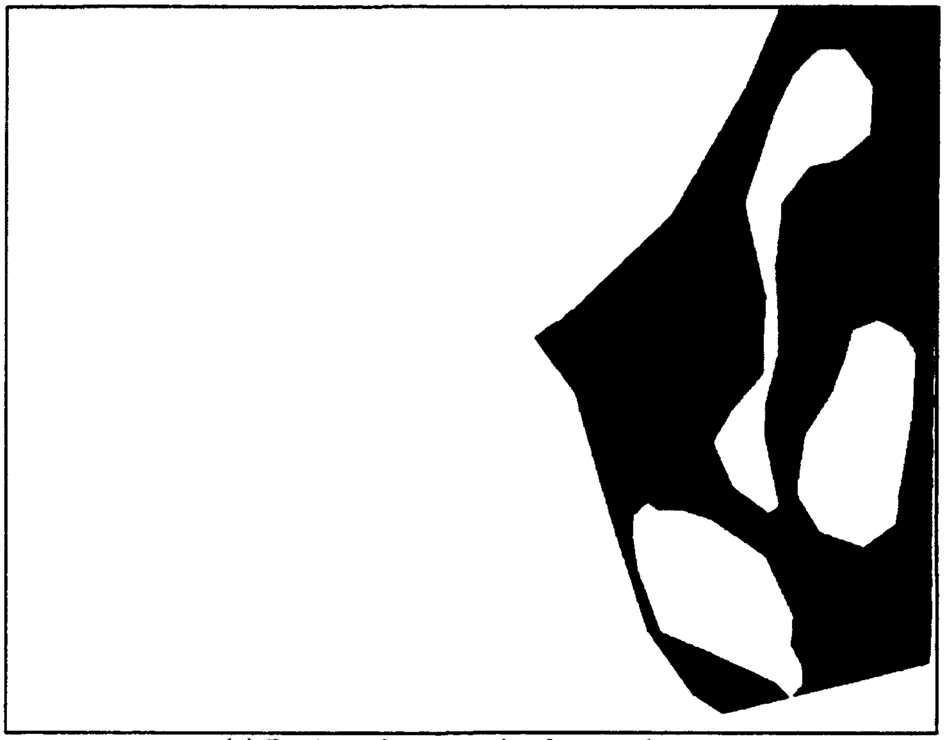
Figure 6: Image delineation using an associated map



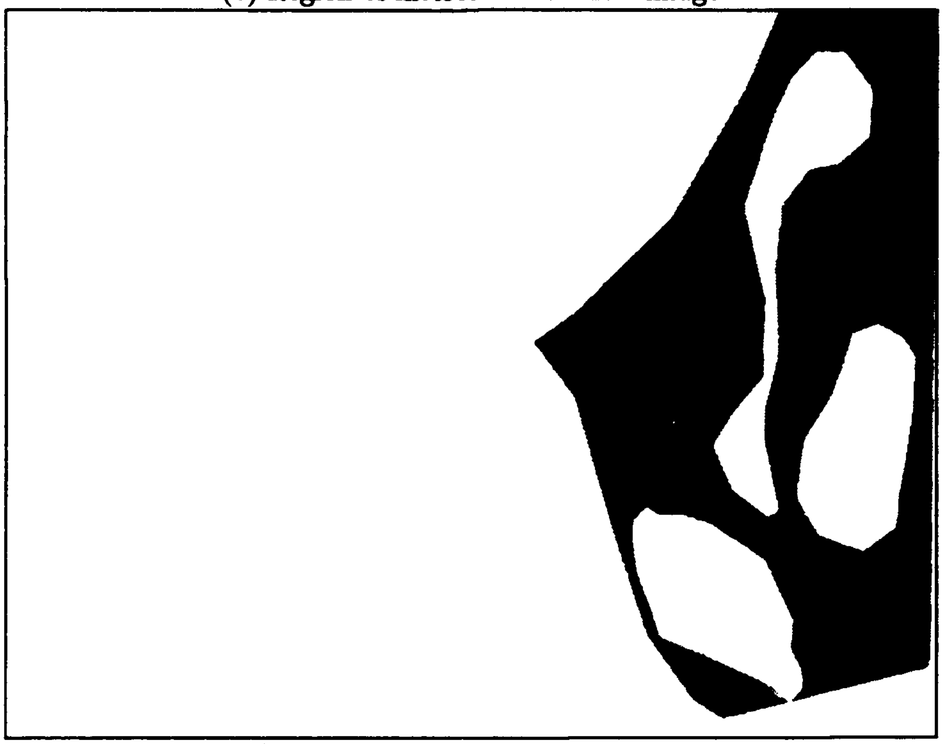
(c) Region of interest in the earlier image

(d) Image to be monitored

Figure 6: (cont.) Image delineation using an associated map



(e) Region of interest in the new image



(f) Image of the region of interest

Figure 6: (cont.) Image delineation using an associated map

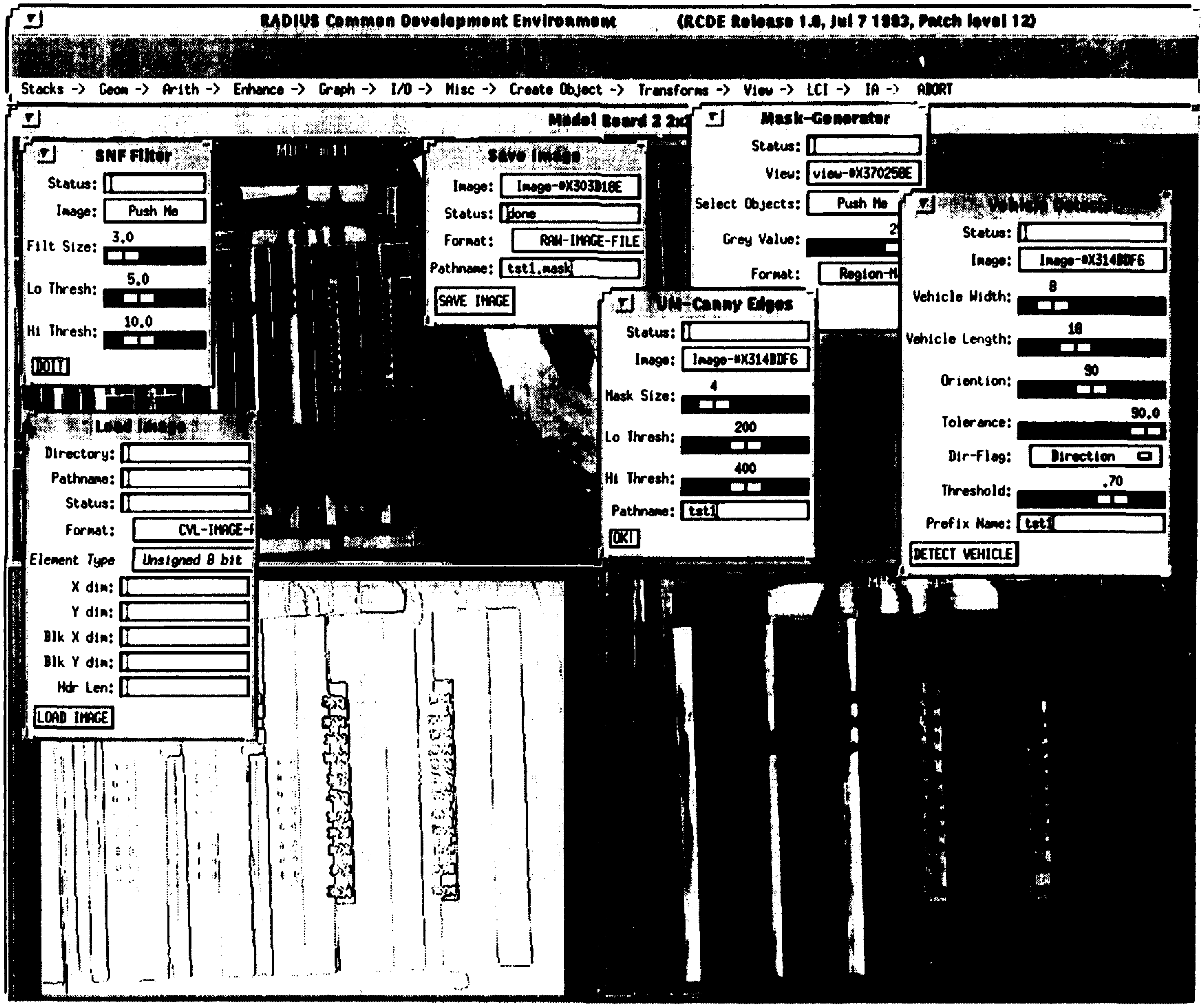


Figure 7: Some functions added to RCDE

- Load raw formatted image,
- Save image in various formats,
- UM-Canny edge detector,
- Vehicle detection

4. We have detected and corrected several minor bugs in the RCDE source code. We have been in contact with the RCDE developers at Martin Marietta, King of Prussia, PA.

3.5. Monitoring Construction Activities

Two site model supported monitoring tasks have been considered in our system. The first is monitoring new construction activities using cylindrical structure as an example. The second is detecting and counting vehicles in a garage area, on roads, and in a training ground. In this section we discuss the subsystem for monitoring construction activities, details of low level features used, the representation of target objects in terms of the low level features, feature extraction scheme, hypothesis generation, and hypothesis verification. An experimental result on monitoring of new cylindrical structure from model board images is presented.

3.5.1. Low Level Feature Extraction:

Edge detection: A Canny edge detector [3] is first used to get an edge map and a gradient direction for each edge pixel. We have found that the edge map is more reliable than region segmentation output, especially when they are used to search for objects in a cluttered image.

Line linking: We apply a line linking program [15] to the edge detector output. In doing this, we first scan the edge map and group the edge pixels according to some predefined templates. We then merge small collinear line fragments into long straight lines.

3.5.2. Object Representation for Cylinders

We use a hierarchically parameterized object model to incorporate knowledge from the site model and the image acquisition conditions into the low level processes. For example, a 3-D model is designed to accommodate prior information about a 3-D cylinder on the ground plane as follows:

(a) height of the cylinder, $h_{3d} \in (h_{\min}, h_{\max})$

(b) radius of its cross-section, $r_{3d} \in (r_{\min}, r_{\max})$

(c) center of its base $(x_w, y_w, 0)$

(d) center of its apex (x_w, y_w, h_{3d})

In the above object model definition, the constraints on a 3-D cylindrical object become part of the object model and independent of camera pose. ($h_{\min}$,$h_{\max}$) is the range of the height of a candidate cylinder, and ($r_{\min}$,$r_{\max}$) is the range of the radius of its cross-section. We further model the contour of a cylinder as an ellipse, a pair of parallel lines, and some geometric relations between them. In doing this, we transfer the 3-D object model onto the following 2-D object model, which depends on the camera parameters, and use it as a working template for detecting cylinders.

1. Ellipse:
   (a) center $c = (x_0, y_0) \in \mathbf{A}$, where **A** is the area of projection of the set of 3-D points of the forms (x_w, y_w, h_{3d}) on the image plane.
   (b) length of the semi-major axis, $a = r_{3d} \times s_c$.
   (c) length of the semi-minor axis, $b = r_{3d} \times s_c \times \cos\alpha$.
   (d) orientation $= \gamma$.
2. Pairs of parallel lines:
   (a) symmetry axis, VL;
   (b) length, $h_{3d} \times s_c \times \sin\alpha$.
   (c) separation, $r_{3d} \times s_c$.
   (d) orientation, $\frac{\pi}{2} + \gamma$.
3. Geometric constraint(s).
   The center of the ellipse should be close to the symmetry axis of the parallel lines.

s_c is a scale factor derived from the camera focal length and image resolution.

3.5.3. Primitive Feature Detection

Primitive features are building blocks used to describe the objects. They are useful for locating possible objects. A robust primitive feature extractor is crucial for successful target detection. The following three primitive feature extractors have been implemented for cylindrical object detection.

Circle detection: A circle is of the form

$$(X - x_0)^2 + (Y - y_0)^2 = r^2$$

where (x_0, y_0) is the center of the circle and r is its radius. A traditional approach to circle direction is the generalized Hough transform [2, 12], which requires a huge amount of memory. We have defined a two-stage template matching scheme for circle detection. In the first stage, edge templates are used to determine possible candidate centers. In the second stage, gradient direction templates are used to re-inspect the selected candidate center points. The details are as follows:

Edge template matching: For each r, we form a search space, QA, by quantizing the angular range $[0°, 360°]$ into $10 \times r$ levels. The radius vector, $V(\psi)$, in the Cartesian coordinate system is

$$V(\psi) = \begin{pmatrix} \Delta x \\ \Delta y \end{pmatrix} = \begin{pmatrix} r \times \cos\psi \\ r \times \sin\psi \end{pmatrix}, \qquad \psi \in QA; \tag{34}$$

it is used as the edge template for a circle. We then apply this template to the edge map and obtain candidate centers which have sufficient numbers of supporting pixels around them.

Gradient direction template matching: The gradient direction of a boundary pixel is the direction from the center to the pixel. We apply both the edge and gradient direction templates to each candidate circle, allowing a three pixel wide tolerance band on the edge template to accommodate slightly misplaced pixels. For an edge pixel to be a supporting pixel, the pixel must fall within the tolerance band and have a gradient direction consistent with the gradient template. We accept those candidates whose consistency scores are above the high threshold for a circle. For candidates whose consistency scores fall between the high and low thresholds, we further apply a radius histogram test: if we plot a histogram of intensity as a function of distance from the candidate center, there should be a steep slope around the radius of the circle. The center and the radius of the k^{th} successful candidate are then stored as Cir_k.

Ellipse detection: An ellipse is of the form

$$\frac{(X - x_0)^2}{a^2} + \frac{(Y - y_0)^2}{b^2} = 1 \tag{35}$$

The scheme we use for detection of an ellipse is similar to the scheme used for circle detection. The difference lies in the way we generate edge templates and gradient direction templates. To simplify the discussion, assuming that the major axis of the ellipse is parallel to the x-axis ($\gamma = 0$), we define the following templates:

Edge templates: The edge pixels for an ellipse satisfy

$$V(\psi) = \begin{pmatrix} \Delta x \\ \Delta y \end{pmatrix} = \begin{pmatrix} a \times \cos\psi \\ b \times \sin\psi \end{pmatrix}, \qquad \psi \in QA. \tag{36}$$

Note that the definition of ψ in (36), shown in Figure 8, is different from the definition of ψ in (34).

Gradient direction template: As shown in Figure 8, for an ellipse, $V(\psi)$ corresponds to point n (instead of m). The gradient orientation is determined by

$$\tan\theta = \frac{\Delta y}{\Delta x} = \frac{a}{b} \times \tan\psi. \tag{37}$$

We define the gradient direction template as

$$G(\psi) = \arctan\left(\frac{a}{b} \times \tan\psi\right) \qquad \psi \in QA. \tag{38}$$

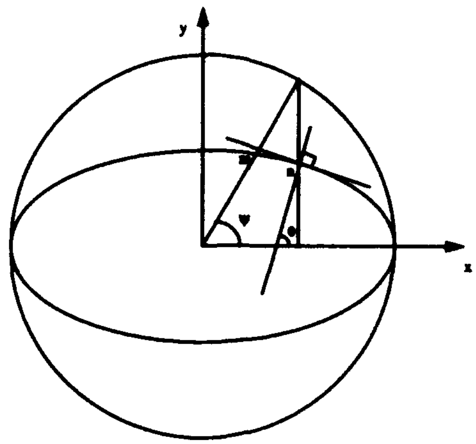


Figure 8: Ellipse

When the camera roll angle is non-zero ($\gamma \neq 0$), (36) and (38) become

$$V(\psi) = \begin{pmatrix} \Delta x \\ \Delta y \end{pmatrix} = \begin{pmatrix} \cos\gamma & -\sin\gamma \\ \sin\gamma & \cos\gamma \end{pmatrix} \begin{pmatrix} a \times \cos\psi \\ b \times \sin\psi \end{pmatrix} \qquad \psi \in QA, \tag{39}$$

$$G(\psi) = \arctan\left(\frac{a}{b} \times \tan\psi\right) + \gamma \qquad \psi \in QA. \tag{40}$$

Line grouping: For line grouping, we use the constraints from the camera model to check candidate parallel lines. Since the silhouette of a cylinder is always projected along the camera viewing direction, we ignore lines which are oriented far away from the expected direction. As shown in Figure 9, two lines, L_i and L_j, form a parallel line pair, $Para_{i,j}$, if they satisfy the following constraints:

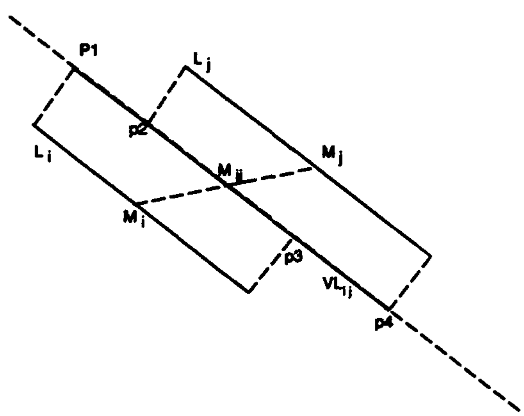


Figure 9: Line grouping

1. parallelism: $|\theta_i - \theta_j| \leq \epsilon_\theta$

2. distance: $dist(L_i, L_j) \in (r_{\min}, r_{\max})$

where

$$dist(L_i, L_j) = \frac{1}{2}\left[pl\_dist(M_i, L_j) + pl\_dist(M_j, L_i)\right]$$

θ_i is the orientation of the i^{th} line, ϵ_θ is the angle deviation threshold, and $pl\_dist(M_p, L_q)$ is the distance from point M_p to line L_q. For each pair of parallel lines, we further compute their axis of symmetry $VL(\theta_{ij}, M_{ij})$, which satisfies

$$\theta_{ij} = \frac{l_i \times \theta_i + l_j \times \theta_j}{l_i + l_j} \tag{41}$$

$$M_{ij} = \frac{1}{2}(M_i + M_j) \tag{42}$$

where $\theta_{i,j}$ is the orientation of the symmetry axis, M_{ij} is a point on the symmetry axis, and l_i is the length of the i^{th} line segr ent. We then define the overlap between L_i and L_j as

$$Overlap(L_i, L_j) = \frac{l_i + l_j}{2.0 \times l_{i,j}} \tag{43}$$

where $l_{i,j}$ is the distance between p_1 and p_4 in Figure 9. In our implementation, only line pairs with overlaps greater than 0.5 are retained as valid parallel line pairs.

3.5.4. Perceptual Grouping and Hypothesis Generation

With primitive features extracted, we detect possible locations of the target. For each primary feature (ellipse candidate, C_k), the following constraints are used to search for supporting secondary features (parallel line pairs, $Para_{i,j}$'s):

1. $\max(pp\_dist(O_k, P_s), pp\_dist(O_k, P_e)) \in (\sin\alpha \times s_c \times h_{\min}, \sin\alpha \times s_c \times h_{\max})$?
2. $\mathrm{mod}(|\theta(\overline{M_{ij}O_k}) - (\frac{\pi}{2} + \gamma)|, 2\pi) < \epsilon_\theta$?

where $\theta(\overline{M_{ij}O_k})$ is the direction from M_{ij} in (42) to the center of C_k, as shown in Figure 10, and $pp\_dist(P_1, P_2)$ is the distance between points P_1 and P_2. If a grouping passes the tests, we evaluate the quality of the grouping by computing

$$H(C_k, Para_{i,j}) = \sum_{l=1}^{3} w_l \times H_l(C_k, Para_{i,j}) \tag{44}$$

where

$$w_1 = w_2 = w_3 = \frac{1}{3} \tag{45}$$

$$H_1(C_k, Para_{i,j}) = \frac{pl\_dist(O_k, VL_{i,j})}{R_k} \tag{46}$$

$$H_2(C_k, Para_{i,j}) = \frac{|R_k - dist(L_i, L_j)/2|}{\epsilon_{width}} \tag{47}$$

$$H_3(C_k, Para_{i,j}) = \frac{\min(pp\_dist(O_k, P_e), pp\_dist(O_k, P_s))}{\max(pp\_dist(O_k, P_e), pp\_dist(O_k, P_s))} \tag{48}$$

If $H(C_k, Para_{i,j})$ is less than a threshold, an hypothesis is formed that there is a cylindrical object located at the corresponding position.

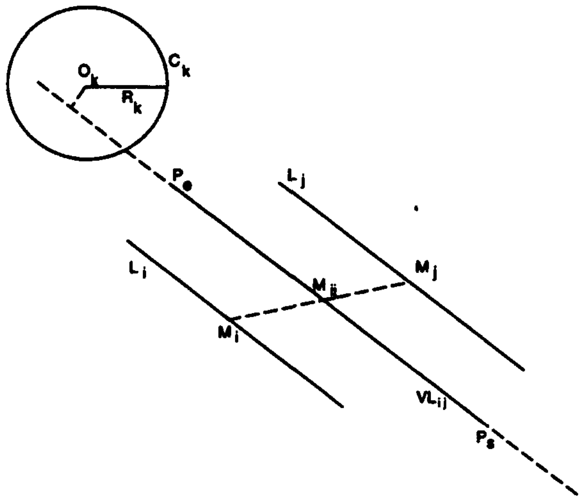


Figure 10: Perceptual grouping

3.5.5. Hypothesis Verification

The hypotheses are then verified by checking for more support from the original edge map, shadow information, and intensity distribution. The following three tests are used in cylindrical object detection.

1. **Model inversion test:** For each candidate cylinder, we fit a model and check its consistency with the original edge map. If the support is above a threshold, we accept it as a valid cylinder. Otherwise, we continue with additional tests.

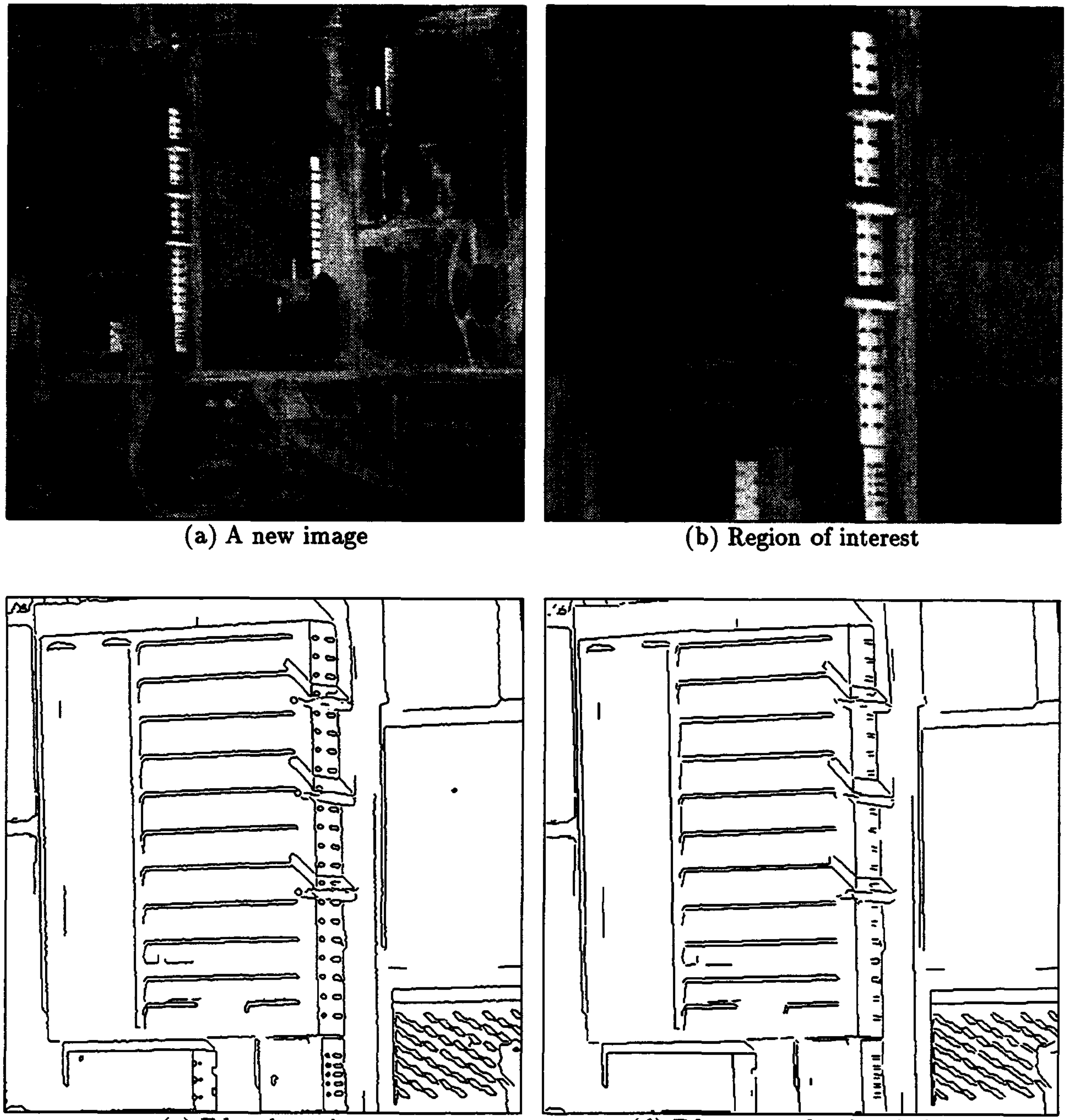
2. **Shadow test:** Since the illumination direction is available from the site model, we delineate a region where the shadow of the proposed cylinder might appear. If we find a supporting shadow (a homogeneously dark region) bounded by a pair of parallel lines within the region, the hypothesis is accepted.

3. **Homogeneity test:** We can also check the intensity variations within the ellipse and the region bounded by the parallel lines. If these variations are much smaller than the intensity variation in the image, we accept the hypothesis.

Once a hypothesis passes the above tests, the detected cylinder is reported to the IA.

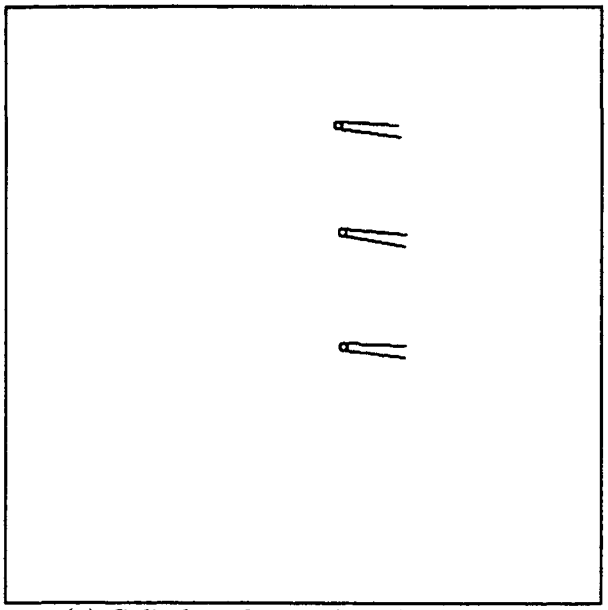
3.5.6. An Example: Chimney Detection

In Figure 11 an example of cylindrical object detection is shown: (a) a new image; (b) the region of interest delineated using the site model; (c) the results of edge detection; (d) the edge map after line linking; (e) cylinders detected in the new image; (f) an earlier image of the same site (the old image has been registered to the coordinates of the new image); (g) the results of cylindrical object detection when the same procedure is applied to the earlier image; and (h) the registration of the cylindrical objects detected in both images.

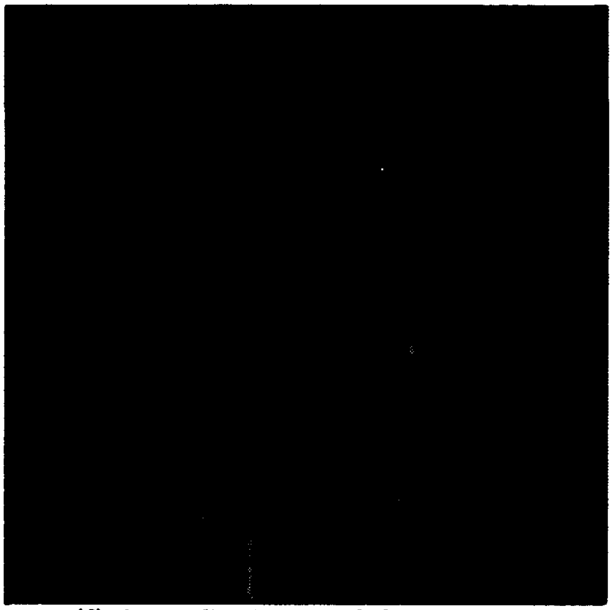


(a) A new image
(b) Region of interest
(c) Edge detection
(d) Edge map after line linking

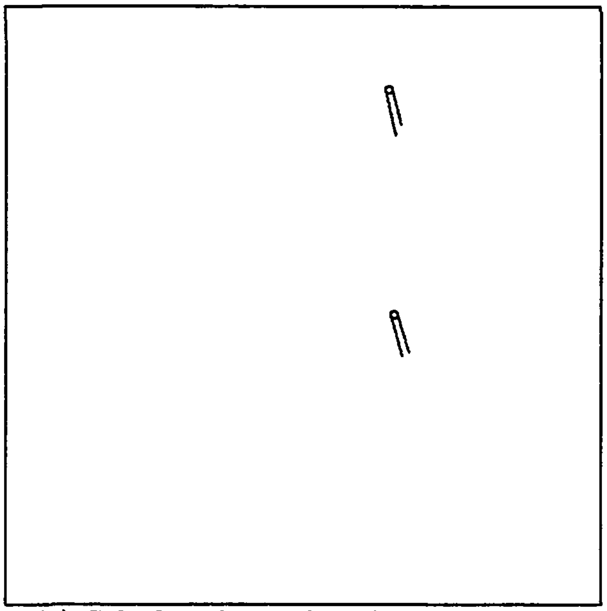
Figure 11: New construction detection



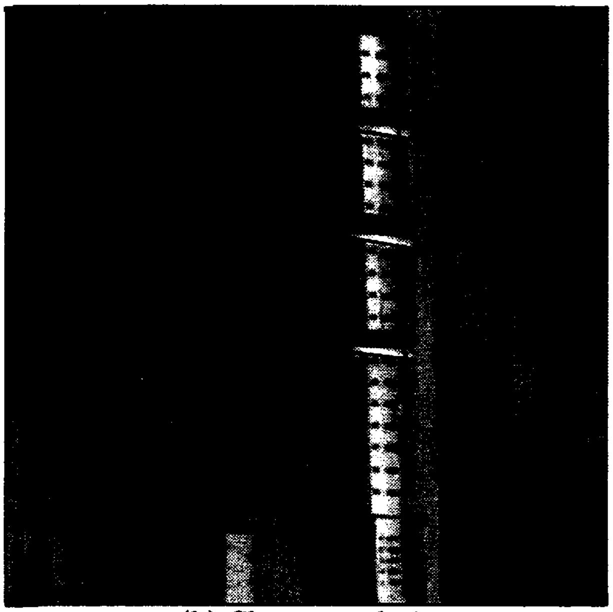
(e) Cylinders detected in the new image



(f) An earlier image of the same site



(g) Cylinders detected in the earlier image



(h) Change analysis

Figure 11: (cont.) New construction detection

Three cylinders are detected in the new image and two in the earlier image. A report that *there is a new cylindrical object (the middle one in (e)) built since the last time the site was investigated* is sent to the IA. The location of the new cylindrical object can be highlighted (not shown here).

3.6. Detecting and Counting Vehicles

The subsystem for carrying out another site model supported image monitoring task, detecting and counting vehicles, is reported in this section. In our implementation, vehicles are modeled as 3-D boxes with width, length and height specifications. Figure 12 shows the block diagram of the subsystem for detecting and counting vehicles. As shown in Figure 12, 3-D object model and site information (camera model, illuminant, etc.) are used through out the procedure. Details of the implementation are discussed as follows.

3.6.1. Edge Detection:

The modified Canny edge detector [15] is used for detection of edges and their gradient directions. Let $\mathbf{H}$ be a mask and define its inner product with an image $\mathbf{U}$ at location (m, n) as

$$< \mathbf{U}, \mathbf{H} >= \sum_i \sum_j h(i,j)u(i+m, j+n) = u(m,n) \bigotimes h(-m,-n)$$

Two mutually orthogonal masks, $\mathbf{H}_1$ and $\mathbf{H}_2$, are used in our implementation. Let

$$\begin{aligned} g_1(m,n) &= < \mathbf{U}, \mathbf{H}_1 > \\ g_2(m,n) &= < \mathbf{U}, \mathbf{H}_2 > \end{aligned}$$

then the magnitude and direction of the gradient vector are

$$\begin{aligned} g(m,n) &= \sqrt{g_1^2(m,n) + g_2^2(m,n)} \\ \theta_g(m,n) &= \arctan \frac{g_2(m,n)}{g_1(m,n)} \end{aligned}$$

3.6.2. Vehicle Representation

A vehicle is modeled as a 3-D box characterized by the following parameters:

- Width: w_{3d};
- Length: l_{3d};
- Height: h_{3d};
- Center: (x_c, y_c, z_c);
- Rotation Matrix: a 3×3 matrix describes the orientation of the local coordinate frame.

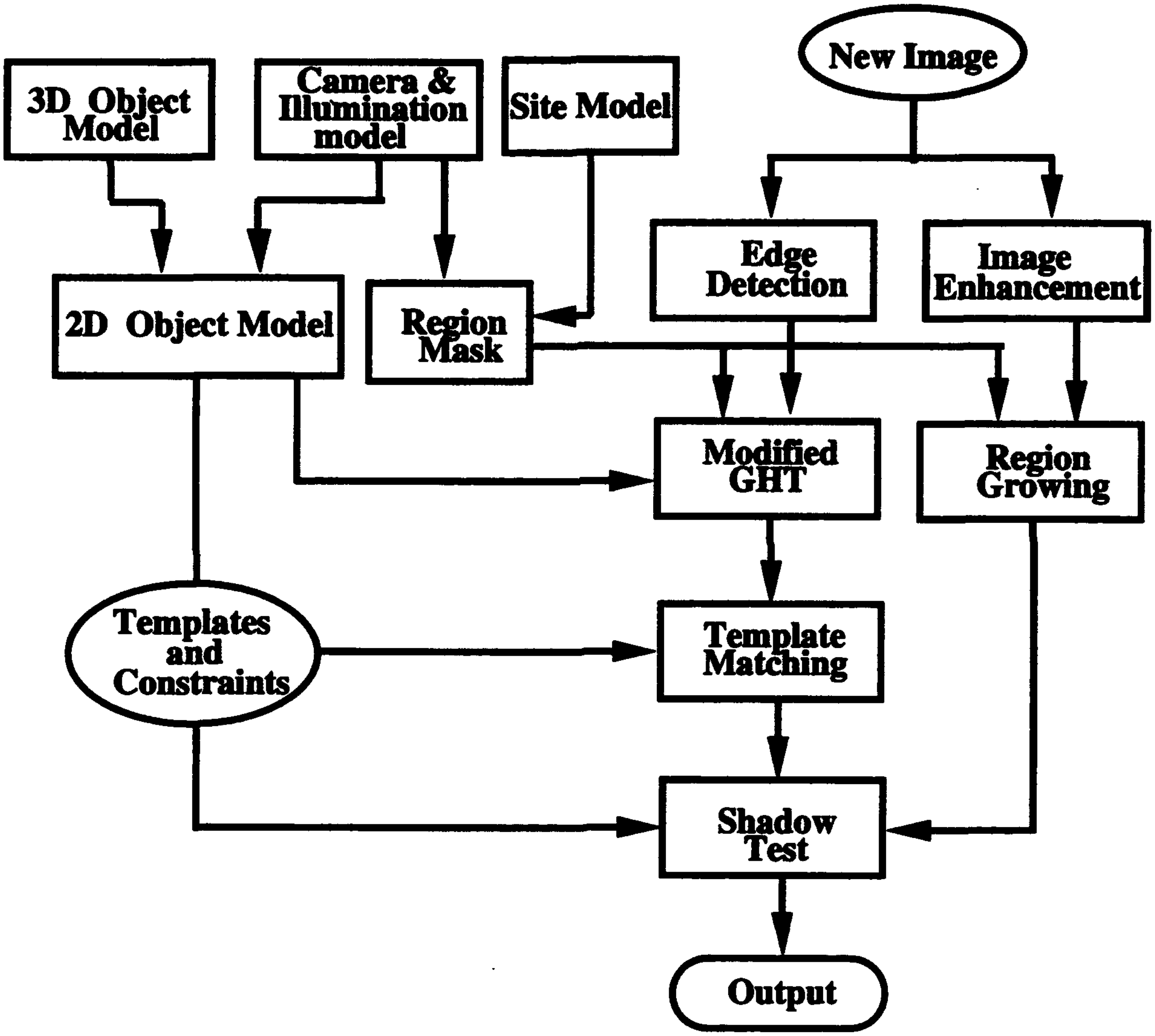


Figure 12: Flowchart for vehicle detection

With the camera model given, we compute the 2-D projection of the 3-D model. Since the height of a vehicle is normally shorter than its length and width, in addition, images are taken from above, the vertical contour of the vehicle is neglegible and the vehicle can be very well approximated by a 2-D rectangle. In our implementation, we search for rectangles to locate candidate vehicles.

3.6.3. Search Scheme

A modified Generalized Hough transform (GHT) is used to locate possible vehicles (by extracting the centers of candidate rectangles). The basic idea is to vote the possible loci of reference points from the detected edge points.

In our case, the reference point is the center of the rectangle. For each edge point, we also computed its gradient direction. The location of the reference point is represented as a function of the gradient direction. All such locations, indexed by gradient direction, are precomputed to form a table, see Table 5. The relevant geometry used to form the table is showed in Figure 13. The searching algorithm is described as follows.

Table 5: Indexed table for reference points

Gradient direction of edge points	Set of radii r^k where r = (r, α)
ϕ_1	$r_1^1, r_2^1, \ldots, r_{m_1}^1$
ϕ_2	$r_1^2, r_2^2, \ldots, r_{m_2}^2$
ϕ_3	$r_1^3, r_2^3, \ldots, r_{m_3}^3$
.	.
.	.
.	.
ϕ_{n-1}	$r_1^{n-1}, r_2^{n-1}, \ldots, r_{m_{n-1}}^{n-1}$
ϕ_n	$r_1^n, r_2^n, \ldots, r_{m_n}^n$

Step 1 – Make a table for the rectangle to be located.

Step 2 – Create an accumulator array of possible reference points, $A(x_{\min} : x_{\max}; y_{\min} : y_{\max})$, and initialize it to zero.

Step 3 – Compute $\phi(x)$ for each edge pixel and vote for possible center of an associated rectangle at

$$
\begin{aligned}
x_c &= x + r(\phi)\cos[\alpha(\phi)] \\
y_c &= y + r(\phi)\cos[\alpha(\phi)]
\end{aligned}
$$

Step 4 – A candidate vehicle is formed for each candidate center whose vote is above a threshold.

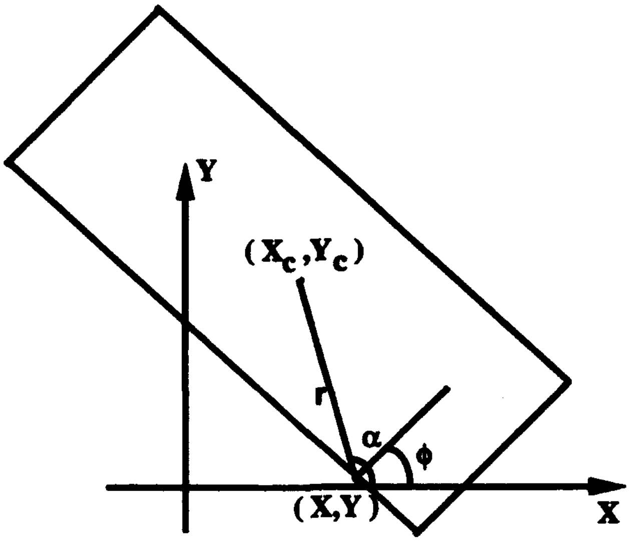


Figure 13: Geometry used to computed reference point

3.6.4. Hypothesis Generation

For each candidate rectangle obtained, we generate a 3-D vehicle and compute its contours in the image. We then compare the contours with the local edge map. If the match is above a threshold, a hypothesis that there is a vehicle at the corresponding location is formed. In our implementation, a rubber-band rectangle template is used to evaluate the matching. A rubber-band rectangle template is similar to a rectangle template with a tolerance band but guarantees that each pixel on the template can get no more than one vote from the pixels on the edge map along the perpendicular direction. We check not only the overall matching, but also the degree of matching on the boundaries in directions along and perpendicular to the vehicle direction. Therefore, to be qualified as a vehicle, the candidate rectangle has to have (almost) complete boundaries on both parallel sides.

3.6.5. Hypothesis Verification

The image of a 3-D vehicle is different from a 2-D rectangle in that there should be a shadow associated with the detected rectangle. Using the illuminant model available from the site model, we can form a hypothesis about the associated shadow region, which includes constraints on the position, size, intensity, and shape of the shadow region. We then detect a shadow next to the candidate vehicle. If the detected shadow is consistent with the prediction from the illuminant model, the candidate vehicle is confirmed. This hypothesis verification method is still under development. With improvements in image resolution and availability of more accurate vehicle models, we plan to develop a more sophisticated verification mechanism. The vehicle detection results presented in this section are obtained before verification.

3.6.6. Experiments

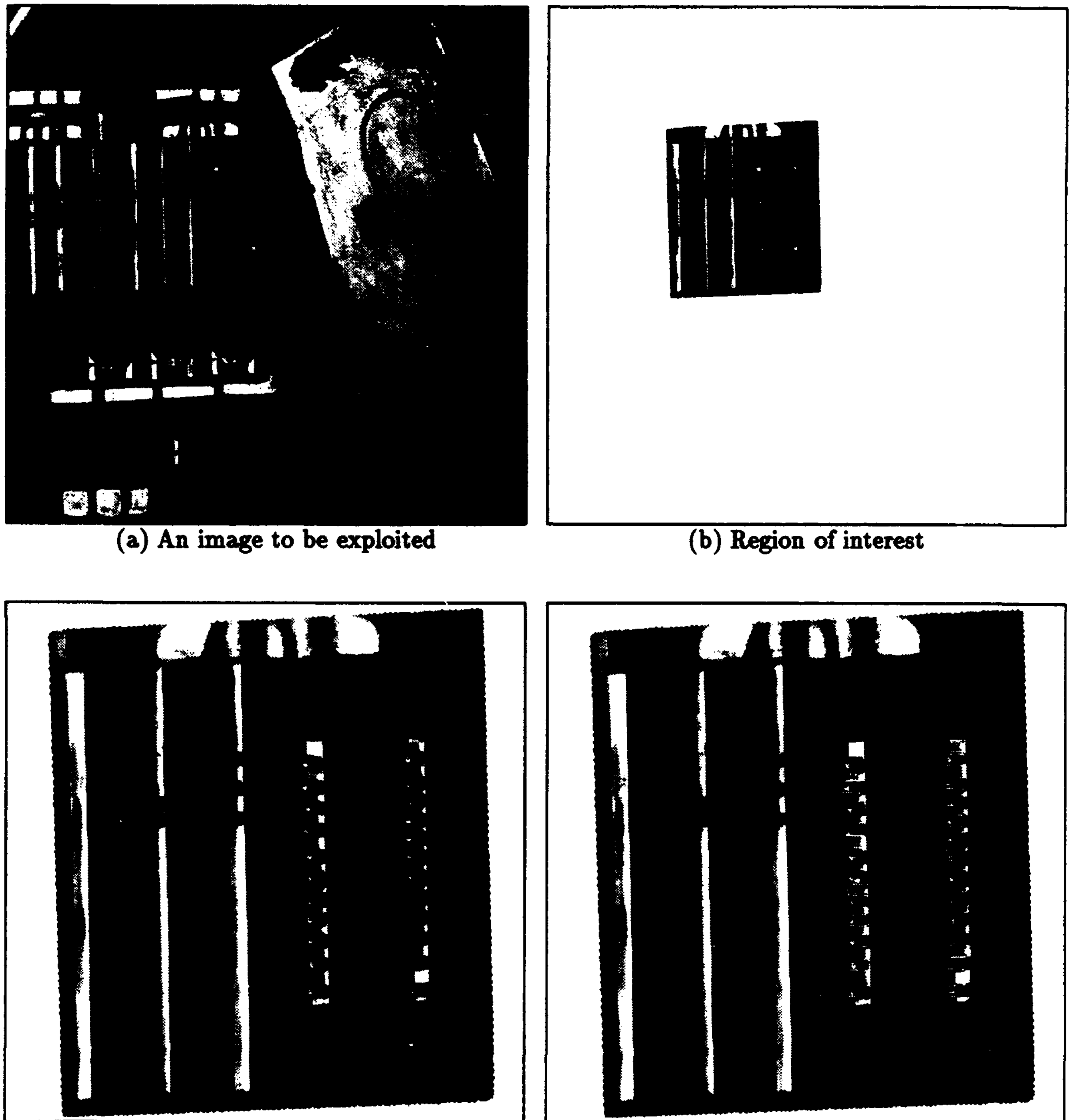
Vehicle detection in a parking area: In Figure 14, an example of vehicle detection in a parking area is shown: (a) an image to be exploited; (b) the area corresponding to the garage of interest, delineated from the region information in the site model; (c) a zoom-in view of the region of interest; (d) the detected vehicles. For vehicle detection in the parking area, we used information about the garage orientation to constrain the possible vehicle parking direction. In this case, a report of "*the garage is about half full*" was sent to the IA.

Vehicle detection on roads: In Figure 15, an example of monitoring vehicles on roads specified by the IA (through a QL profile) is shown: (a) an input image, (b) a window to be monitored, (c) the area corresponding to roads of interest. Since vehicles drive along the road direction, the directions of the roads are also generated and used as an additional constraint for vehicle detection. In the algorithm, only candidates whose orientations are approximately along the road direction are considered to be valid vehicles on the roads. Finally, in (d) the detected vehicles are shown.

Vehicle detection on a training ground: In Figure 16, an example of vehicle detection in a training ground is shown: (a) an input image; (b) a window to be monitored; (c) the area corresponding to the training ground which is of intelligent interest; and (d) the detected vehicles. For vehicle detection in a training ground (since vehicles can be oriented in any direction), we have to detect possible vehicles in all directions.

3.7. Ground Plane Image-to-Image Registration

Image-to-image registration is required in the following situations: (1) It is important for setup of an initial site model, especially when no ground control points are available. Image-to-image registration can provide 3-D coordinates of some control points through triangulation. (2) It is critical for automatic registration of new images into an existing site model. From the site model we may have 3-D coordinates of some feature points; to locate the image plane positions of these feature points we need an image-to-image registration algorithm. (3) For generating 2-D region delineations corresponding to arbitrary viewing directions. In [18], Zheng and Chellappa developed an automatic image-to-image registration technique for nadir images. The work was later extended to automatic registration of oblique images [4]. On the RADIUS project, partial knowledge about the cameras is available. We have developed algorithms which use the partially known camera parameters to perform image registration efficiently. Details of the image-to-image registration technique are reported in this section.



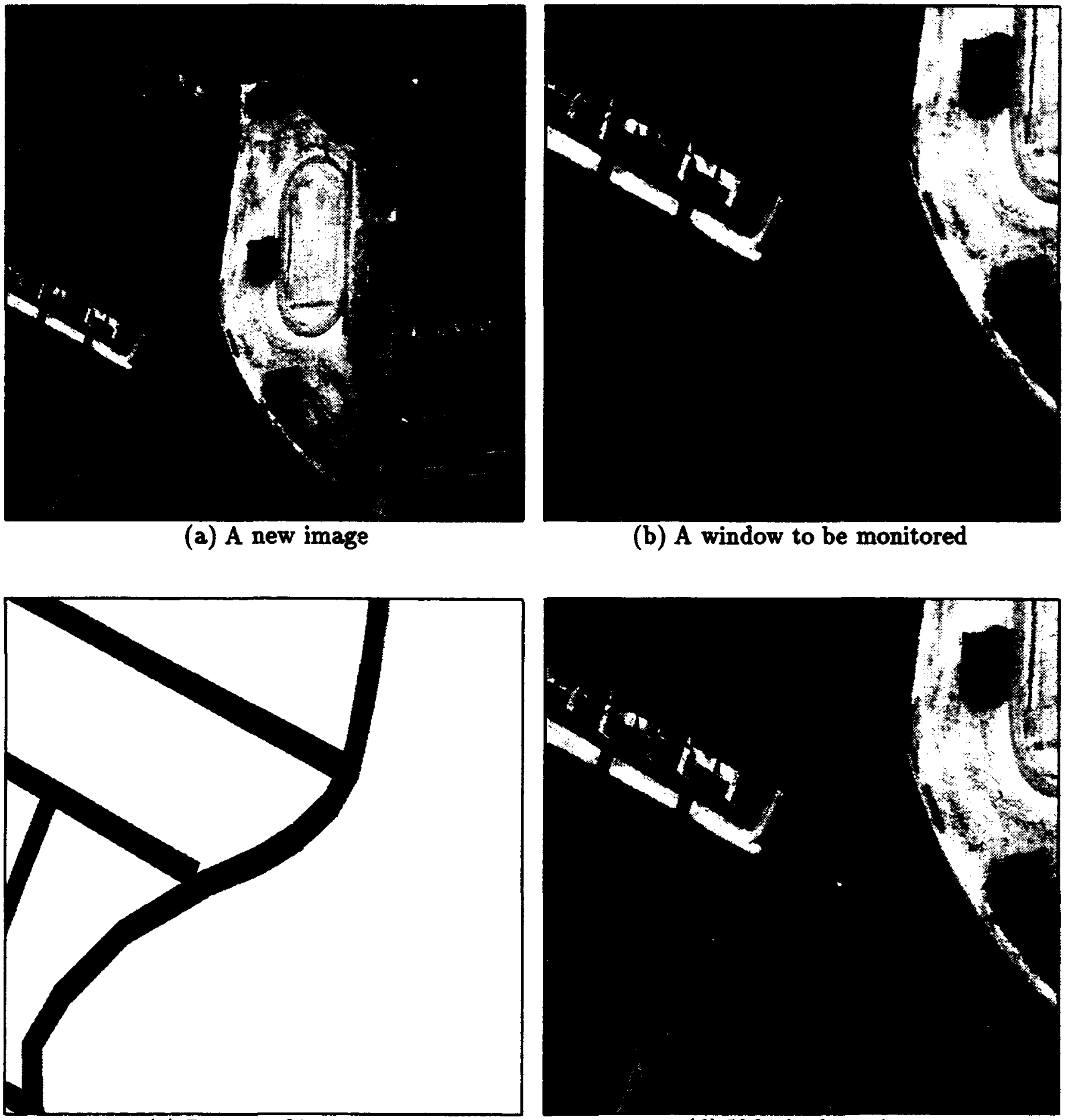
(a) An image to be exploited

(b) Region of interest

(c) A zoom-in view of the region of interest

(d) Vehicles detected in the parking area

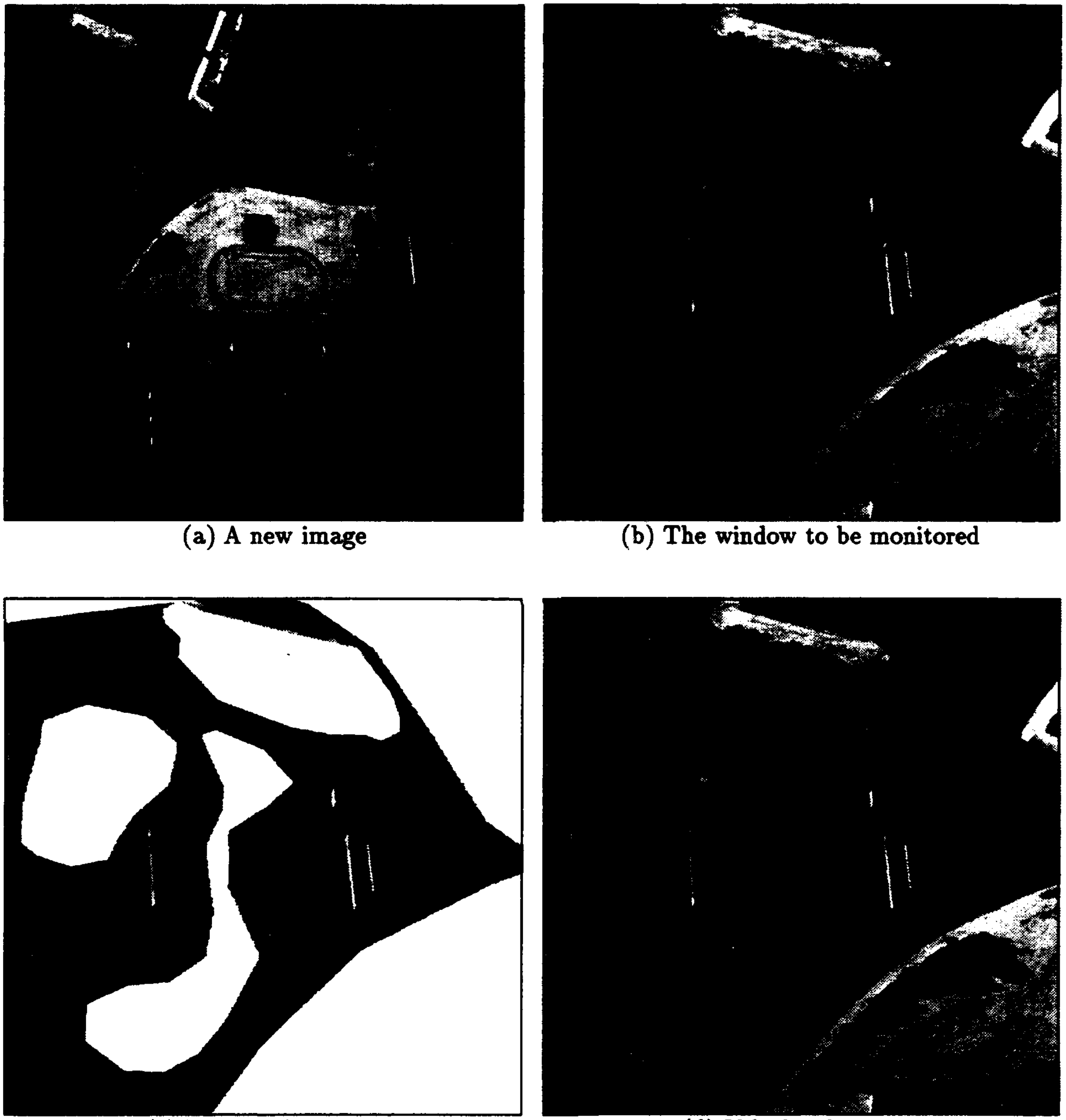
Figure 14: Vehicle detection in a parking area



(a) A new image (b) A window to be monitored

(c) Region of interest (d) Vehicle detection

Figure 15: Vehicle detection on communication roads



(a) A new image (b) The window to be monitored

(c) Region of interest (d) Vehicles detection

Figure 16: Vehicle detection in a training ground

3.7.1. Relationship between Two Images

Assuming we have two sets of camera parameters, a point $(x_w, y_w, z_w)^t$ in the world coordinates is represented in the two camera centered coordinate systems by

$$\begin{pmatrix} x_1 \\ y_1 \\ z_1 \end{pmatrix} = \mathbf{R}_1 \begin{pmatrix} x_w - x_{1o} \\ y_w - y_{1o} \\ z_w - z_{1o} \end{pmatrix} = \begin{pmatrix} r^1_{11} & r^1_{12} & r^1_{13} \\ r^1_{21} & r^1_{22} & r^1_{23} \\ r^1_{31} & r^1_{32} & r^1_{33} \end{pmatrix} \begin{pmatrix} x_w - x_{1o} \\ y_w - y_{1o} \\ z_w - z_{1o} \end{pmatrix} \quad (49)$$

and

$$\begin{pmatrix} x_2 \\ y_2 \\ z_2 \end{pmatrix} = \mathbf{R}_2 \begin{pmatrix} x_w - x_{2o} \\ y_w - y_{2o} \\ z_w - z_{2o} \end{pmatrix} = \begin{pmatrix} r^2_{11} & r^2_{12} & r^2_{13} \\ r^2_{21} & r^2_{22} & r^2_{23} \\ r^2_{31} & r^2_{32} & r^2_{33} \end{pmatrix} \begin{pmatrix} x_w - x_{2o} \\ y_w - y_{2o} \\ z_w - z_{2o} \end{pmatrix} \quad (50)$$

respectively. So the transform from $(x_1, y_1, z_1)^t$ to $(x_2, y_2, z_2)^t$ is

$$\begin{aligned} \begin{pmatrix} x_2 \\ y_2 \\ z_2 \end{pmatrix} &= \mathbf{R}_2 \begin{pmatrix} x_w - x_{2o} \\ y_w - y_{2o} \\ z_w - z_{2o} \end{pmatrix} \\ &= \mathbf{R}_2\mathbf{R}_1^t \begin{pmatrix} x_1 \\ y_1 \\ z_1 \end{pmatrix} + \mathbf{R}_2 \begin{pmatrix} x_{1o} - x_{2o} \\ y_{1o} - y_{2o} \\ z_{1o} - z_{2o} \end{pmatrix} \\ &= \mathbf{R}_{21} \begin{pmatrix} x_1 \\ y_1 \\ z_1 \end{pmatrix} + \begin{pmatrix} \delta x_o \\ \delta y_o \\ \delta z_o \end{pmatrix} \quad (51) \end{aligned}$$

where

$$\begin{aligned} \mathbf{R}_{21} &= \mathbf{R}_2\mathbf{R}_1^t \quad (52) \\ &= \begin{pmatrix} r^2_{11} & r^2_{12} & r^2_{13} \\ r^2_{21} & r^2_{22} & r^2_{23} \\ r^2_{31} & r^2_{32} & r^2_{33} \end{pmatrix} \begin{pmatrix} r^1_{11} & r^1_{21} & r^1_{31} \\ r^1_{12} & r^1_{22} & r^1_{32} \\ r^1_{13} & r^1_{23} & r^1_{33} \end{pmatrix} \quad (53) \\ &= \begin{pmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{pmatrix} \quad (54) \end{aligned}$$

$$\begin{pmatrix} \delta x_o \\ \delta y_o \\ \delta z_o \end{pmatrix} = \mathbf{R}_2 \begin{pmatrix} x_{1o} - x_{2o} \\ y_{1o} - y_{2o} \\ z_{1o} - z_{2o} \end{pmatrix} \quad (55)$$

Note that for points on the ground plane we have $z_w = D$, a constant, so that

$$\begin{pmatrix} x_w \\ y_w \\ D \end{pmatrix} = \mathbf{R}_1^t \begin{pmatrix} x_1 \\ y_1 \\ z_1 \end{pmatrix} + \begin{pmatrix} x_{1o} \\ y_{1o} \\ z_{1o} \end{pmatrix} \quad (56)$$

$$D = r^1_{13}x_1 + r^1_{23}y_1 + r^1_{33}z_1 + z_{1o}$$

$$\frac{D - z_{1o}}{z_1} = r^1_{13}\frac{x_1}{z_1} + r^1_{23}\frac{y_1}{z_1} + r^1_{33}$$

Assume f_1 and f_2 are the focal lengths of camera-1 and camera-2, ϵ_1 and ϵ_2 are the pixel spacings for image-1 and image-2, and $(X_1, Y_1)^t$ and $(X_2, Y_2)^t$ are the image plane coordinates for image-1 and image-2, respectively; then using central projection we have

$$\frac{D - z_{1o}}{z_1} = r^1_{13}\frac{\epsilon_1}{f_1}X_1 + r^1_{23}\frac{\epsilon_1}{f_1}Y_1 + r^1_{33}$$

or

$$\frac{f_1}{\epsilon_1 z_1}(D - z_{1o}) = r^1_{13}X_1 + r^1_{23}Y_1 + r^1_{33}\frac{f_1}{\epsilon_1} \tag{57}$$

For image-2 we have

$$\begin{aligned}
\frac{\epsilon_2}{f_2}X_2 &= \frac{x_2}{z_2} \\
&= \frac{r_{11}x_1 + r_{12}y_1 + r_{13}z_1 + \delta x_o}{r_{31}x_1 + r_{32}y_1 + r_{33}z_1 + \delta z_o} \\
&= \frac{r_{11}\frac{\epsilon_1}{f_1}X_1 + r_{12}\frac{\epsilon_1}{f_1}Y_1 + r_{13} + \frac{\delta x_o}{z_1}}{r_{31}\frac{\epsilon_1}{f_1}X_1 + r_{32}\frac{\epsilon_1}{f_1}Y_1 + r_{33} + \frac{\delta z_o}{z_1}}
\end{aligned}$$

$$X_2 = \frac{f_2}{\epsilon_2}\cdot\frac{r_{11}X_1 + r_{12}Y_1 + \frac{f_1}{\epsilon_1}r_{13} + \frac{\delta x_o}{D - z_{1o}}(r^1_{13}X_1 + r^1_{23}Y_1 + r^1_{33}\frac{f_1}{\epsilon_1})}{r_{31}X_1 + r_{32}Y_1 + \frac{f_1}{\epsilon_1}r_{33} + \frac{\delta z_o}{D - z_{1o}}(r^1_{13}X_1 + r^1_{23}Y_1 + r^1_{33}\frac{f_1}{\epsilon_1})} \tag{58}$$

Similarly,

$$Y_2 = \frac{f_2}{\epsilon_2}\cdot\frac{r_{21}X_1 + r_{22}Y_1 + \frac{f_1}{\epsilon_1}r_{23} + \frac{\delta y_o}{D - z_{1o}}(r^1_{13}X_1 + r^1_{23}Y_1 + r^1_{33}\frac{f_1}{\epsilon_1})}{r_{31}X_1 + r_{32}Y_1 + \frac{f_1}{\epsilon_1}r_{33} + \frac{\delta z_o}{D - z_{1o}}(r^1_{13}X_1 + r^1_{23}Y_1 + r^1_{33}\frac{f_1}{\epsilon_1})} \tag{59}$$

So the ground plane transform from a pixel (X_1, Y_1) in image-1 to image-2 is given by

$$X_2 = \frac{A\,X_1 + B\,Y_1 + C}{E\,X_1 + F\,Y_1 + G} \tag{60}$$

$$Y_2 = \frac{H\,X_1 + I\,Y_1 + J}{E\,X_1 + F\,Y_1 + G} \tag{61}$$

where A, B, C, E, F, G, H, I and J are constants:

$$A = \frac{f_2}{\epsilon_2}\left(r_{11} + \frac{\delta x_o}{D - z_{1o}}r^1_{13}\right) \tag{62}$$

$$B = \frac{f_2}{\epsilon_2}\left(r_{12} + \frac{\delta x_o}{D - z_{1o}}r^1_{23}\right) \tag{63}$$

$$C = \frac{f_2 f_1}{\epsilon_2 \epsilon_1}\left(r_{13} + \frac{\delta x_o}{D - z_{1o}}r^1_{33}\right) \tag{64}$$

$$E = r_{31} + \frac{\delta z_o}{D - z_{1o}}r^1_{13} \tag{65}$$

$$F = r_{32} + \frac{\delta z_o}{D - z_{1o}}r^1_{23} \tag{66}$$

$$G = \frac{f_1}{\epsilon_1}\left(r_{33} + \frac{\delta z_o}{D - z_{1o}} r^1_{33}\right) \quad (67)$$

$$H = \frac{f_2}{\epsilon_2}\left(r_{21} + \frac{\delta y_o}{D - z_{1o}} r^1_{13}\right) \quad (68)$$

$$I = \frac{f_2}{\epsilon_2}\left(r_{22} + \frac{\delta y_o}{D - z_{1o}} r^1_{23}\right) \quad (69)$$

$$J = \frac{f_2 f_1}{\epsilon_2 \epsilon_1}\left(r_{23} + \frac{\delta y_o}{D - z_{1o}} r^1_{33}\right) \quad (70)$$

3.7.2. Registration Using Known Camera Parameters

When the camera parameters are available, we can register the ground planes of any two images using (60–70). In Figure 17, the registration of two oblique images of different resolution is shown: (a) a high resolution model board image, M1, with Ground Space Distance (GSD) equal to 15 inch, $\alpha = 30.7°$, $\beta = 339.2°$, and $\gamma = 197.6°$; (b) a low resolution model board image, M40, with GSD=26 inch, $\alpha = 44.7°$, $\beta = 254.8°$, and $\gamma = 340.4°$; (c) the registration of M40 to M1; and (d) the registration of M1 to M40.

3.7.3. Registration with Unknown Camera Parameters

When no information about the camera is available, we still can register two oblique images by automatically matching (at least) four corresponding points and solving for the transform parameters in (60–61). For the principal point of image-1 we have $(X_1, Y_1) = (0, 0)$; its corresponding location in the coordinates of image-2 is $(X_2, Y_2) = (\frac{C}{G}, \frac{J}{G})$. As long as the two cameras are well above the ground, the principal point of image-1 must be a well-defined point (finite) in the coordinates of image-2. Hence $G \neq 0$. The ground plane transformation of image-1 to image-2 can be determined in terms of eight parameters a_i, $i = 1, \ldots, 8$ as

$$X_2 = \frac{a_3 X_1 + a_5 Y_1 + a_1}{-a_7 X_1 - a_8 Y_1 + 1} \quad (71)$$

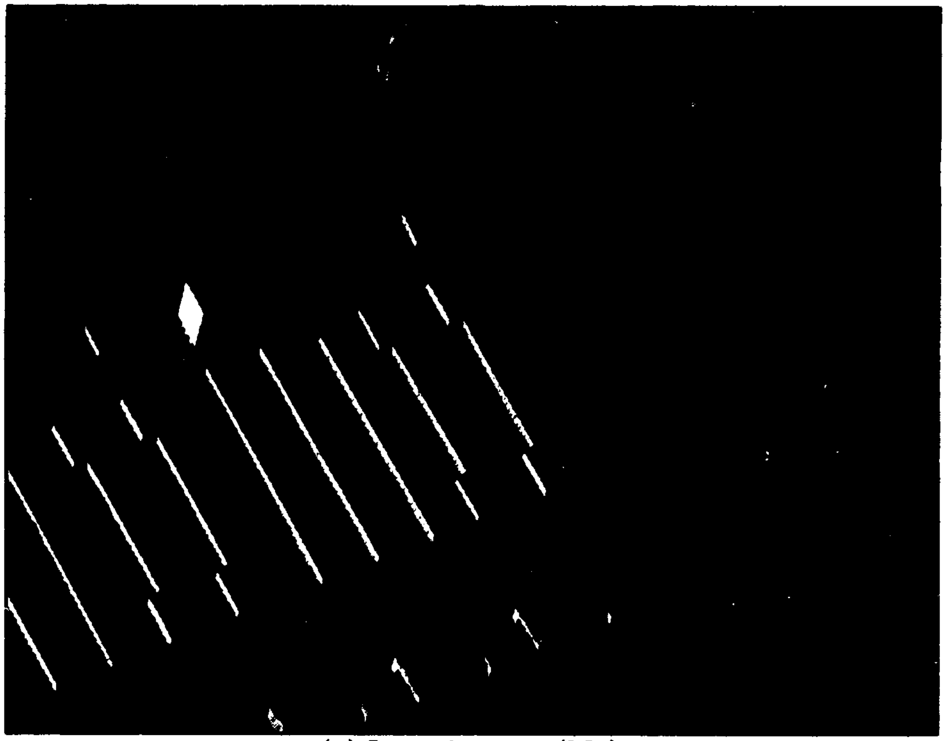
$$Y_2 = \frac{a_4 X_1 + a_6 Y_1 + a_2}{-a_7 X_1 - a_8 Y_1 + 1} \quad (72)$$

where

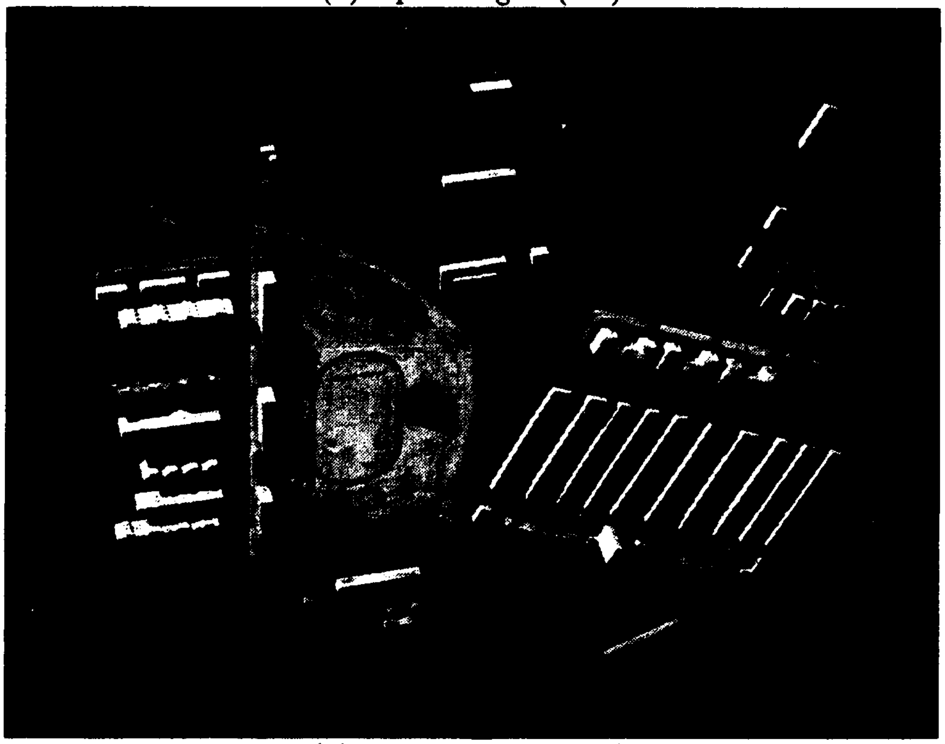
$$a_1 = \frac{C}{G} = \frac{f_2}{\epsilon_2} \frac{\left(r_{13} + \frac{\delta x_o}{D - z_{1o}} r^1_{33}\right)}{\left(r_{33} + \frac{\delta z_o}{D - z_{1o}} r^1_{33}\right)}$$

$$a_2 = \frac{J}{G} = \frac{f_2}{\epsilon_2} \frac{\left(r_{23} + \frac{\delta y_o}{D - z_{1o}} r^1_{33}\right)}{\left(r_{33} + \frac{\delta z_o}{D - z_{1o}} r^1_{33}\right)}$$

$$a_3 = \frac{A}{G} = \frac{f_2 \epsilon_1}{f_1 \epsilon_2} \frac{\left(r_{11} + \frac{\delta x_o}{D - z_{1o}} r^1_{13}\right)}{\left(r_{33} + \frac{\delta z_o}{D - z_{1o}} r^1_{33}\right)}$$

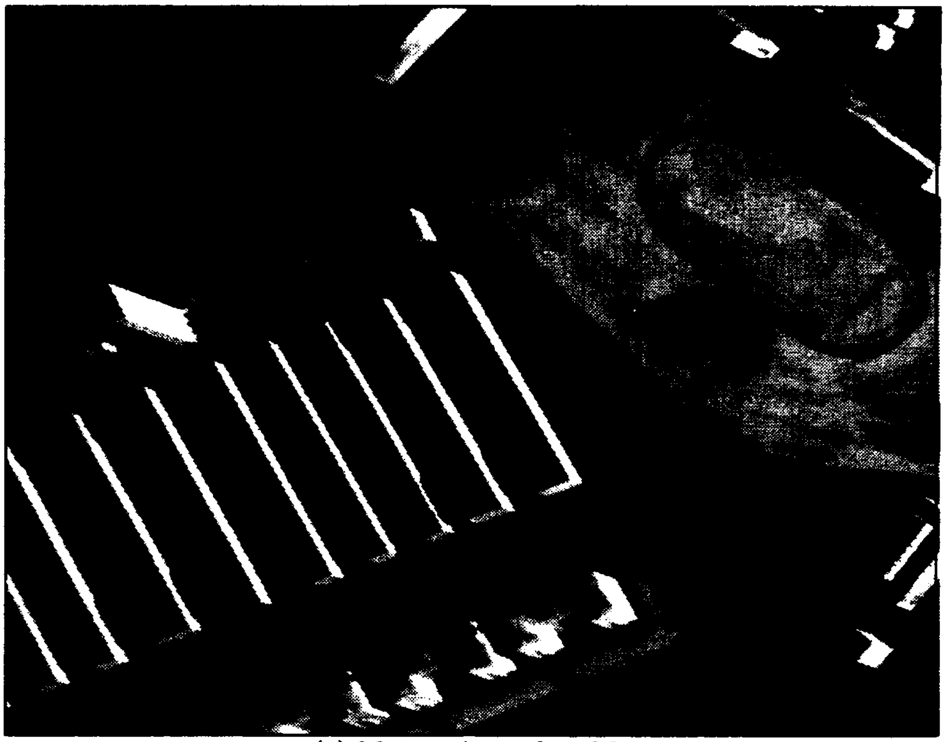


(a) Input image-1 (M1)

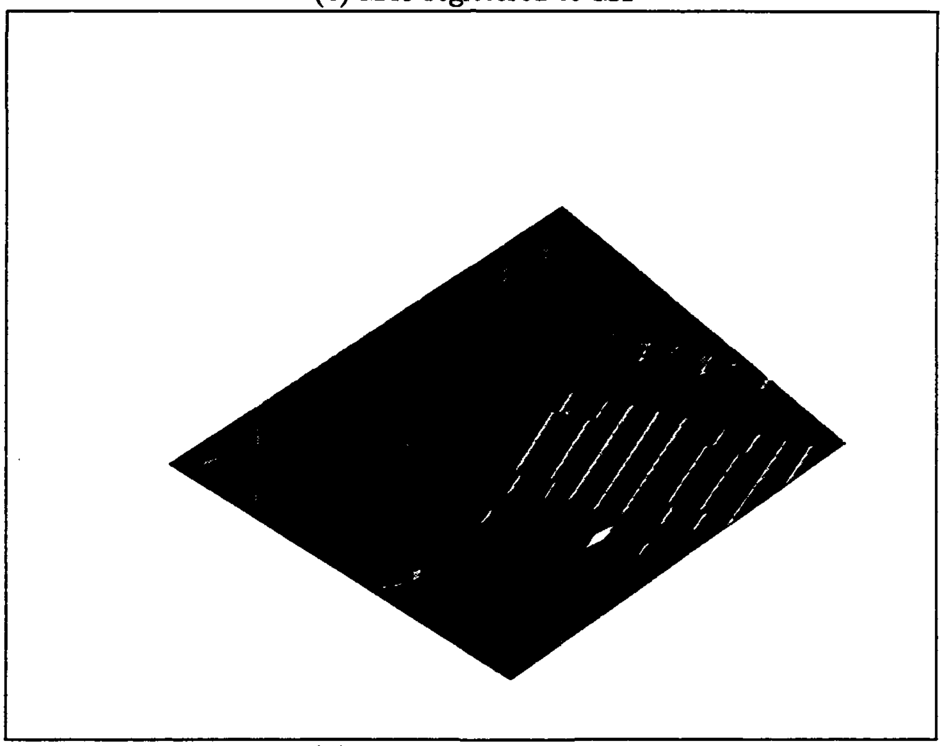


(b) Input image-2 (M40)

Figure 17: Registration of two oblique images (camera parameters are known).



(c) M40 registered to M1



(d) M1 registered to M40

Figure 17: (cont.) Registration of two oblique images (camera parameters are known).

$$a_4 = \frac{H}{G} = \frac{f_2\epsilon_1\left(r_{21} + \frac{\delta y_o}{D-z_{1o}} r^1_{13}\right)}{f_1\epsilon_2\left(r_{33} + \frac{\delta z_o}{D-z_{1o}} r^1_{33}\right)}$$

$$a_5 = \frac{B}{G} = \frac{f_2\epsilon_1\left(r_{12} + \frac{\delta x_o}{D-z_{1o}} r^1_{23}\right)}{f_1\epsilon_2\left(r_{33} + \frac{\delta z_o}{D-z_{1o}} r^1_{33}\right)}$$

$$a_6 = \frac{I}{G} = \frac{f_2\epsilon_1\left(r_{22} + \frac{\delta y_o}{D-z_{1o}} r^1_{23}\right)}{f_1\epsilon_2\left(r_{33} + \frac{\delta z_o}{D-z_{1o}} r^1_{33}\right)}$$

$$a_7 = -\frac{E}{G} = -\frac{\epsilon_1}{f_1}\frac{r_{31} + \frac{\delta z_o}{D-z_{1o}} r^1_{13}}{\left(r_{33} + \frac{\delta z_o}{D-z_{1o}} r^1_{33}\right)}$$

$$a_8 = -\frac{F}{G} = -\frac{\epsilon_1}{f_1}\frac{r_{32} + \frac{\delta z_o}{D-z_{1o}} r^1_{23}}{\left(r_{33} + \frac{\delta z_o}{D-z_{1o}} r^1_{33}\right)}$$

When camera parameters are not available, the eight parameters are obtained by solving the linear equations

$$a_1 + X_{1i}a_3 + Y_{1i}a_5 + X_{1i}X_{2i}a_7 + Y_{1i}X_{2i}a_8 = X_{2i} \quad (73)$$

$$a_2 + X_{1i}a_4 + Y_{1i}a_6 + X_{1i}Y_{2i}a_7 + Y_{1i}Y_{2i}a_8 = Y_{2i} \quad (74)$$

for $i = 1, \ldots, N$, where N is the number of matched points.

Overview of the registration algorithm: Figure 18 illustrates the image registration algorithm. Given two images, we first use an illuminant direction estimator [17, 18] to get an initial estimate of the camera orientation change. A small number of feature points are then located using a Gabor wavelet model for detecting local curvature discontinuities [9]. The feature points extracted from different frames are matched using area correlation. Three match verification tests are used to exclude false matches. After the initial matching is achieved, a multiresolution transform-and-correct matching is implemented to obtain high accuracy registration. At each resolution, image-2 is first transformed to the coordinates of image-1 using the estimated matching parameters and then match refinement is performed on the feature points extracted in image-1.

Feature point detection: For feature point extraction we use a Gabor wavelet decomposition and the local scale interaction based algorithm reported in [9]. The basic wavelet function used in the decomposition is of the form

$$\Phi(X, Y, \vartheta) = e^{-(X'^2+Y'^2)+i\pi X'} \quad (75)$$

$$X' = X\cos\vartheta + Y\sin\vartheta$$

$$Y' = -X\sin\vartheta + Y\cos\vartheta$$

where ϑ is the preferred spatial orientation. In our experiments ϑ is discretized into four orientations. The feature points are extracted as the local maxima of the energy measure

$$I(X, Y) = \max_{\vartheta}\{||W_{j_1}(X, Y, \vartheta) - \gamma W_{j_2}(X, Y, \vartheta)||\} \quad (76)$$

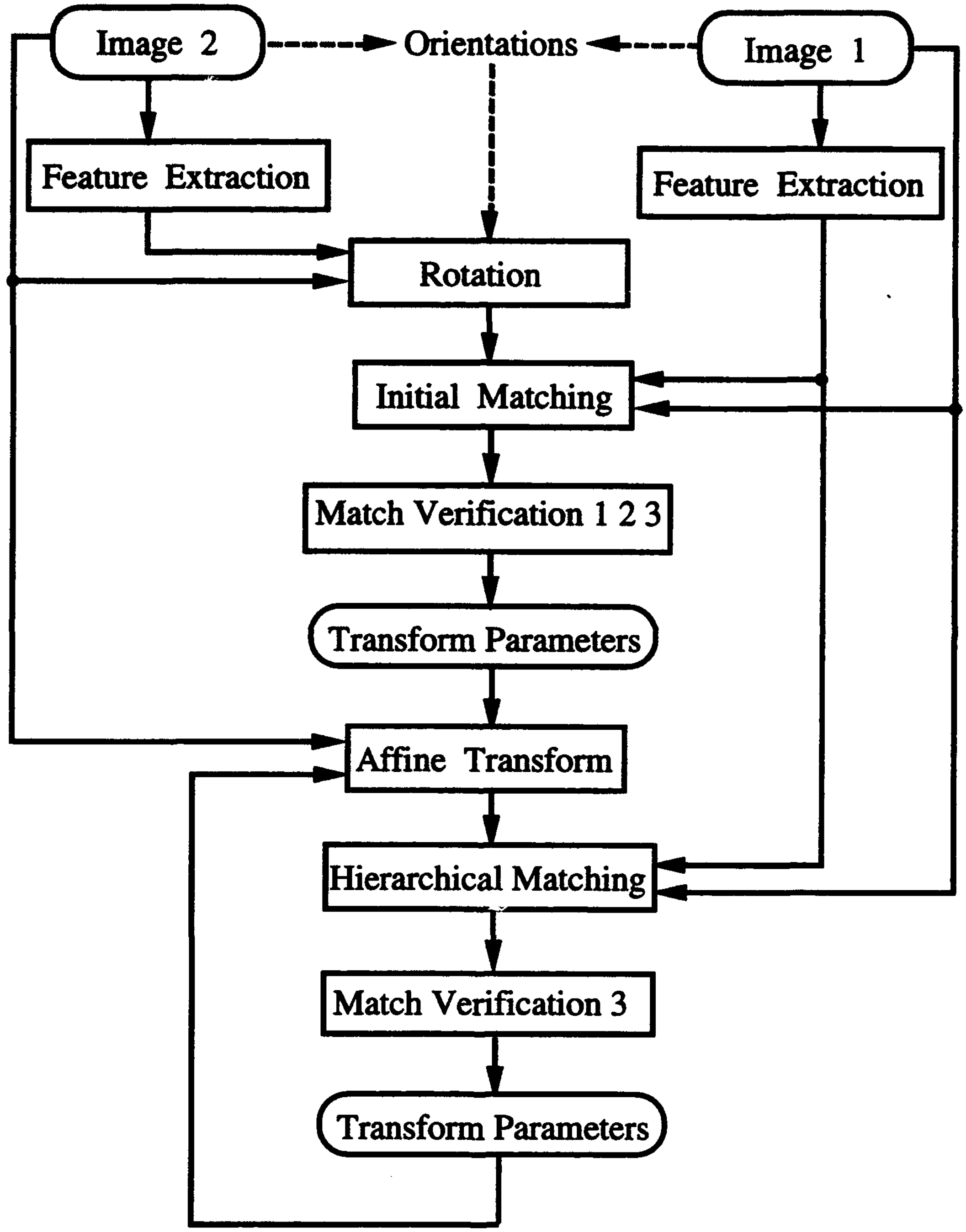


Figure 18: Block diagram of a general image-to-image registration algorithm

where

$$W_j(X,Y,\vartheta) = \mathbf{f} \otimes \Phi(2^{-\frac{j}{2}}X, 2^{-\frac{j}{2}}Y, \vartheta), \; j = \{j_1, j_2\}.$$

Here j_1 and j_2 are two dilation parameters, and $\gamma = 2^{(j_1 - j_2)}$ is a normalizing factor. In implementing the above algorithm, we further require the energy measure for a feature point to be the maximum in a neighborhood with radius equal to 10 and above a threshold.

Match verification: In our algorithm, the initial matching is implemented on 2-D rotation compensated images. Since no further knowledge about the camera parameters is used in the initial matching, false matches due to perspective deformation and similarities between similar objects are inevitable. Automatic exclusion of these false matches is a key to success in image registration. We have used three tests to exclude less reliable matches.

1. **Distance test:** The translation between the rotation-compensated images should not be larger than a certain fraction of the image size. A ··· lid matching pair, (X_r, Y_r) and (X_l, Y_l), should satisfy

$$\begin{cases} d_x = |X_r - X_l| & \le \lambda L_x \\ d_y = |Y_r - Y_l| & \le \lambda L_y \\ |X_r - X_l| + |Y_r - Y_l| & \le \kappa \max\{L_x, L_y\} \end{cases} \tag{77}$$

   For example, $\lambda = \frac{1}{2}$ and $\kappa = \frac{3}{2}\lambda$. L_x and L_y are image size along x and y directions respectively.

2. **Variation test:** The translations used in the correct matches should support each other, i.e.

$$|d_i - \overline{d}| \le \mu\sigma \tag{78}$$

   where d_i is the distance between the i^{th} matching pair, $\overline{d}$ and σ are the mean and standard deviation of the distances for all the matched feature pairs, and μ is a threshold, for example $\mu = \sqrt{3}$ for the uniform distribution.

3. **Outlier exclusion:** The matched feature pairs should satisfy the image transform model. Candidate matching pairs with large residual errors should be excluded. This test also helps to exclude matches on building roofs, etc.

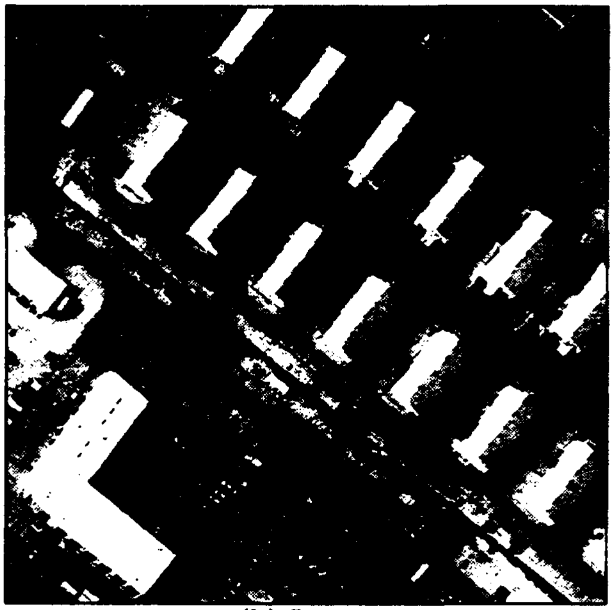
Experimental results: In Figures 19 and 20, the registration of two aerial images is shown: (a) the image taken by the first camera; (b) the image taken by the second camera; (c) the registration of (b) to (a); and (d) the difference between (a) and (c).

4. Ongoing and Future Work

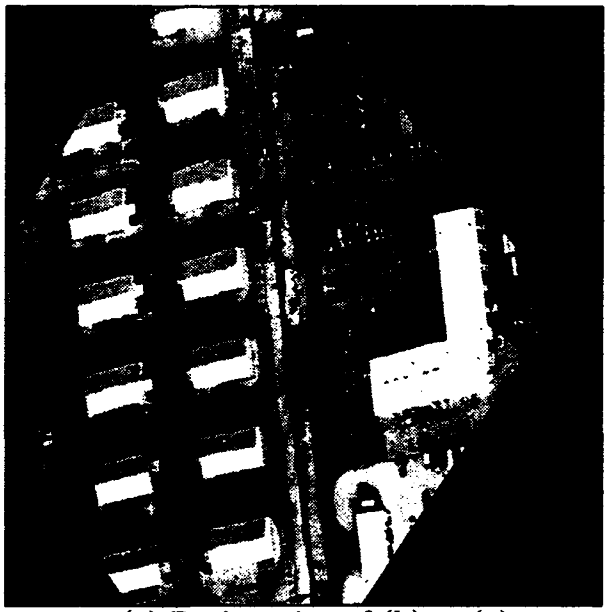
4.1. Hierarchical Model-Based Segmentation

We are developing a general model-based procedure for image segmentation based on a hierarchical connected component analysis. This method will be useful for detection and

(a) Image-1



(b) Image-2



(c) Registration of (b) to (a)

(d) Difference between (a) and (c)

Figure 19: Registration of two aerial images (Example-1)


(a) Image-1


(b) Image-2


(c) Registration of (b) to (a)


(d) Difference between (a) and (c)

Figure 20: Registration of two aerial images (Example-2)

counting as well as for change detection based on comparing the components of the segmentation algorithm. This multi-level segmentation is used as a search space for various complex objects. The hierarchical connected component analysis procedure consists of a multi-stage, region-growing type of segmentation. The initial stage is the result of an initial segmentation of the image into connected components. In our implementation, two adjacent pixels are considered to be connected if the difference in their gray level values is less than a threshold ε. Each successive stage merges adjacent components (or regions) of the previous stage. The selection of the regions to be merged is based on local analysis of region properties. Currently, only average boundary contrast is used. The new stage represents a coarser segmentation of the image. A complete hierarchy is built, i.e., the merging process ends when there are no more regions to merge.

The hierarchy is used as a search space for diverse objects. Currently, a model for an object of interest is interactively created. The model includes various distinctive elements of the object and geometric and topological relations among them. The search process tests for the presence of these elements at several levels of the hierarchy. It is expected that intact or nearly intact elements of the object appear at coarser levels, thus allowing us to find the object using minimal search.

The general paradigm for extracting candidate objects from an image is the following:

Locale specification: A **locale** in the image is selected to start the process of finding object candidates. A locale is defined with respect to known objects or it can correspond to the whole image. The set of basic connected components that lie within a locale is called the basis B_0.

Segmentation: As previously mentioned, the segmentation is a simple gray-level connected components algorithm. The result is a labeled image, in which each connected component is assigned a unique value. These connected components will be referred to as basic components. A characteristic of this set is that any boundary between any two components has an average contrast greater than the threshold ε.

Next, a region adjacency graph (RAG) is constructed from the basic component set. This RAG will be referred to as RAG(0). The parameter 0 indicates that it is the initial RAG of the hierarchy, which is computed next. Several properties of each region are computed; they include area, perimeter, boundary average intensity contrast, etc. In parallel, a list of boundaries is computed from RAG(0), and it is sorted in increasing order of the boundaries' average intensity contrast.

Hierarchy of segments: Starting from RAG(0), the hierarchy consists of the sequence of adjacency graphs RAG(0), RAG(1), ..., RAG(i), ..., RAG(n). Each RAG(i) is formed by merging the regions whose common boundary has minimum average contrast (CONT(i)) in RAG($i-1$). Therefore, any boundary in RAG(i) has average contrast greater than CONT(i). The minimum contrast boundaries at each stage are located through the precomputed boundary list. This list is updated after merging regions (i.e., after creating RAG(i)), since new edges are created, some become redundant and the ones with contrast equal to CONT(i) disappear.

A unique symbolic representation is maintained for each region at each level of the hierarchy. Let $r_{i,j}$ be the j^{th} region (in some arbitrary order) in RAG(i). Each region $r_{i,j}$ in RAG(i) has two kinds of link: (1) a link to each of the regions $\{p_{k,l} \mid k < i \text{ and } p_{k,l} \text{ is a component of } r_{i,j}\}$ and (2) a link to region $t_{m,n}$, where $m > i$ and $r_{i,j}$ is a component of $t_{m,n}$. For all j, the first link of $r_{0,j}$ is NULL; if some $r_{i,j}$ is not a component of any region, its second link is NULL.

The hierarchy can be viewed abstractly as a tree, where each node in the tree is a region (a basic component or a multiple basic component region). The lowest level corresponds to the basic components. Given that there are n basic components obtained from a segmentation using threshold ε, there will be at most $2n$ nodes in the tree.

Search: The hierarchy is the basis for extracting information during the **search** process. The search elements are regions (2D structures). The search procedure initially looks for a basis (a level in the hierarchy) that includes at least a seed. A seed is a region that satisfies necessary conditions, specified by the model. A seed is preferably chosen high in the hierarchy since it is desired that complete objects be found as early in the search as possible (top-down approach).

Next, search looks for combinations of regions that satisfy the conditions expressed by the model. It is guided by predefined search heuristics. The purpose of the heuristic search is to systematically order the search space in order to attain a complete, yet efficient, search. The final output is a list of object candidates.

4.2. Automatic Image-to-Site-Model Registration

Currently, image-to-site-model registration requires that the IA manually select and adjust several control points whose 3-D coordinates in the world coordinate system are known. On the RADIUS project, it is assumed that approximate camera parameters are available. We are developing two automatic image-to-site-model registration algorithms. When approximate 3-D coordinates of the camera stare point are available, we will use an image-to-image registration algorithm to automatically search for the image domain locations of control points whose 3-D coordinates are available from the site model and perform camera resection to get an accurate camera model for the newly acquired image. When the camera stare point is unknown, even with given approximate camera orientation information, the displacement between the new image and the projected world coordinates can be quite large. We will first perform automatic feature detection to select a small set of feature points and then do image-to-image registration based on these feature points. We will do another image-to-image registration to get the image domain locations of a set of control points whose 3-D world coordinates are known. Camera resection can then be performed and an accurate camera model for the new image can be obtained.

4.3. Automatic Optimum Image Selection

Given a change monitoring task in a specific region, several images are usually available. How to automatically select the best images for the given monitoring task based on the scene, illuminant and imaging conditions is an interesting research topic. We plan to develop an automatic site analysis algorithm which will analyze visibility, detectability, and unambiguity, and will generate invariance measures for each feature object. These measurements will also be useful for automatic control point selection and model supported optimization. We also plan to develop a shadow detection and correction algorithm.

4.4. QL Interfaces

Based on our progress in using RCDE in connection with vehicle detection and construction monitoring, we are working with members of TASC team and with some RADIUS users to develop more sophisticated QL profiles. This will include more sophisticated model based object detection algorithms and user friendly menu and query driven image exploitation recipes.

4.5. Integration of Collateral Information

An advantage of model supported image analysis is that collateral information can be used to improve efficiency and accuracy. The more collateral information is used, the easier the monitoring tasks become. Currently, collateral information such as a region map is manually generated for the site model. We plan to develop a semiautomatic region map generation algorithm. The following scenarios will be considered: (1) When collateral information is available on an ordinary map, we will use an automatic curve tracing algorithm to transfer the region curves from the map to the site model. (2) When images taken from different types of sensor are available, we will derive regions from composition of segmentation results using images taken from an appropriate sensor. For example, SAR images are good for segmentation of water, concrete structures, and vegetation. (3) Region information can also be derived from an associated digital terrain map, when it is available. We will integrate the database management facility provided by the THREAD project into our system. We also plan to integrate an image synthesis capability into our system. We will also investigate the incorporation of temporal information into the monitoring algorithm.

5. Other Related Work

5.1. Feature Extraction in SAR images

The RADIUS project will benefit by progress in high resolution SAR imagery analysis tasks such as region segmentation and target detection. Recently, we have developed a constant false alarm rate (CFAR) point target detection algorithm for high resolution SAR imagery [16]. Traditional CFAR detection algorithms produce many false targets when applied to single-look, high-resolution, fully polarimetric SAR images, due to the presence of speckle. We have developed a two-stage CFAR detector followed by conditional dilation for detecting point targets in polarimetric SAR images. In the first stage possible targets are detected, and false targets due to the speckle are removed by using global statistical parameters. In the second stage, the local statistical parameters are used to detect targets in regions adjacent to targets detected in the first stage. Conditional dilation is then performed to recover target pixels lost in second stage CFAR detection. The performance of a CFAR detector is degraded if an incorrect statistical model is adopted and the data are correlated. A goodness-of-fit test is performed to choose the appropriate distribution, and the effects of decorrelation of the data are considered. Good experimental results were obtained when our method was applied to single-look, high-resolution, fully polarimetric SAR images acquired from Dr. Les Novak of MIT Lincoln Laboratory. We have also developed a CFAR detector for non-Gaussian clutter distributions such as the K, Weibull and lognormal distributions. This algorithm has been tested on single look, single polarization SAR images.

5.2. Building Delineation

Building detection is of interest in site model construction and change monitoring. Recently, we have developed an energy function based approach for detection of rectangular shapes in an image. Our building detection algorithm is based on line grouping [8]. The proposed edge-based approach involves extracting straight lines from an edge map of the image. Then a Markov-random field (MRF) is built on these lines, i.e., a suitable neighborhood and an energy function are specified based on the relative orientations and spatial locations of the lines. This energy function can be construed as a measure of the conditional probability of observing the lines given the rectangular shapes (the positions and number of which are unknown) in the image. Minimizing the energy function is equivalent to selecting maximum likelihood estimates of the rectangular shapes in the image from the observed lines. Simulated examples are presented to demonstrate the robustness of the proposed method. This approach, supplemented with some qualitative information about shadows and gradients, has been used to detect rectangular buildings in real aerial images. Due to the poor quality of the real images, only partial shapes are extracted in some cases. A modified deformable contour ("snakes") based approach is then used for completion of the partial shapes.

6. Summary and Conclusions

At the end of the first year of the RADIUS project, we have made considerable progress on mastering RCDE, developed some prototypes of QL profiles for imagery monitoring, and transferred some of our results to Martin Marietta. Based on our experience during the first year, we have made research plans for two up-coming years of the RADIUS project.

References

[1] ARPA, *RADIUS Testbed System Operations Concept*, July 1993.

[2] D. H. Ballard, "Generalizing the Hough transform to detect arbitrary shapes," *Pattern Recognition*, Vol. 13, pp. 111–122, 1981.

[3] J. F. Canny, "A computational approach to edge detection," *IEEE Trans. Pattern Anal. Machine Intell.*, Vol. PAMI-8, pp. 679–698, Nov. 1986.

[4] R. Chellappa, Q. Zheng, L. Davis, D. DeMenthon, and A. Rosenfeld, "Site-model-based change detection and image registration," in *Proc. DARPA Image Understanding Workshop* (Washington, DC), pp. 205–216, Apr. 1993.

[5] D. Climenson, "Model-Supported Exploitation Concepts." RADIUS Project Office, Oct. 1992.

[6] J. D.M. McKeown, "Toward automatic cartographic feature extraction," in *Mapping and Spatial Modeling for Navigation* (L. Pau and L. Kanal, eds.), pp. 149–180, Berlin: Springer-Verlag, 1990.

[7] A. Huertas and R. Nevatia, "Detecting buildings in aerial images," *Comput. Vision, Graphics, Image Processing*, Vol. 41, pp. 131–152, 1988.

[8] S. Krishnamachari and R. Chellappa, "An energy minimization approach to building detection in aerial images," in *Proc. of IEEE Conf. Acoust., Speech, and Signal Proc.* (Adelaide, Australia), 1994.

[9] B. S. Manjunath, R. Chellappa, and C. Malsburg, "A feature based approach to face recognition," in *Proc. IEEE Conf. Comput. Vision Pattern Recognition* (Champaign, Illinois), pp. 373–378, 1992.

[10] B. Methley, *Computational Models in Surveying and Photogrammetry*, Glasgow and London: Blackie, 1986.

[11] J. Mundy, R. Welty, L. Quam, T. Strat, W. Bremner, M. Horwedel, D. Hackett, and A. Hoogs, "The RADIUS Common Development Environment," in *Proc. DARPA Image Understanding Workshop* (San Diego, CA), pp. 215–226, Jan. 1992.

[12] S. D. Shapiro, "Feature space transforms for curve detection," *Pattern Recognition*, Vol. 10, pp. 129–143, 1978.

[13] V. Venkateswar and R. Chellappa, "A hierarchical approach to detection of buildings in aerial images," Tech. Rep. CAR-TR-567, Center for Automation Research, University of Maryland, College Park, Aug. 1991.

[14] V. Venkateswar and R. Chellappa, "Hierarchical stereo matching using feature groupings," Tech. Rep. CAR-TR-556, Center for Automation Research, University of Maryland, College Park, May 1991.

[15] V. Venkateswar and R. Chellappa, "Extraction of straight lines in aerial images," *IEEE Trans. Pattern Anal. Machine Intell.*, Vol. 16, pp. 1111–1116, Nov. 1992.

[16] Y. Wang, R. Chellappa, and Q. Zheng, "Detection of point targets in high resolution SAR images," in *Proc. of IEEE Conf. Acoust., Speech, and Signal Proc.* (Adelaide, Australia), 1994.

[17] Q. Zheng and R. Chellappa, "Estimation of illuminant direction, albedo and shape from shading," *IEEE Trans. Pattern Anal. Machine Intell.*, Vol. PAMI-13, pp. 680–702, 1991.

[18] Q. Zheng and R. Chellappa, "A computational vision approach to image registration," *IEEE Trans. Image Processing*, Vol. 3, pp. 311–326, July 1993.