(2)

ETL-0562

AD-A220 006

Built-Up Area Feature Extraction

Second Year Technical Progress Report

David M. McKeown, Jr.

Carnegie Mellon University
5000 Forbes Avenue
Pittsburgh, Pennsylvania 15213

February 1990

Prepared for:

U.S. Army Corps of Engineers
Engineer Topographic Laboratories
Fort Belvoir, Virginia 22060-5546

Destroy this report when no longer needed.
Do not return it to the originator.

The findings in this report are not to be construed as an official Department of the Army position unless so designated by other authorized documents.

The citation in this report of trade names of commercially available products does not constitute official endorsement or approval of the use of such products.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	February 1990	Annual FROM 88/7 TO 89/6

4. TITLE AND SUBTITLE

Built-Up Area Feature Extraction: Second Year Technical Progress Report

5. FUNDING NUMBERS

DACA72-87-C-0001

6. AUTHOR(S)

McKeown, David M. Jr.

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Carnegie-Mellon University
5000 Forbes Avenue
Pittsburgh, PA 15213

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

U.S. Army Engineer Topographic Laboratories
CEETL-RI-T
Fort Belvoir, Va 22060-5546

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

ETL-0562

11. SUPPLEMENTARY NOTES

Previous report in this series: ETL-0561 Built-Up Area Feature Extraction First Year Report February 1990

12a. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for Public Release;
Distribution is Unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT (Maximum 200 words)

This report describes research performed by the Digital Mapping Laboratory at Carnegie Mellon University on the analysis of aerial images of built-up areas during the second year of Contract DACA72-87-C-0001. During this year we have built on previous research, in road network extraction and in the detection and delineation of buildings using monocular analysis, accomplished during the first year of this resarch contract.

We have expanded our research in monocular analysis to include the detailed analysis of shadows. Our shadow analysis research has resulted in three techniques for the interpretation of monocular imagery: building prediction, grouping of related building hypotheses, and building hypothesis verification. In addition, we have implemented a technique to acquire estimates of building heights using the lengths of cast shadows.

14. SUBJECT TERMS

Feature extraction
Scene interpretation
Monocular analysis
Built-up area

15. NUMBER OF PAGES

37

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

GENERAL INSTRUCTIONS FOR COMPLETING SF 298

The Report Documentation Page (RDP) is used in announcing and cataloging reports. It is important that this information be consistent with the rest of the report, particularly the cover and title page. Instructions for filling in each block of the form follow. It is important to ***stay within the lines*** to meet ***optical scanning requirements.***

Block 1. Agency Use Only *(Leave blank)*.

Block 2. Report Date. Full publication date including day, month, and year, if available (e.g. 1 Jan 88). Must cite at least the year.

Block 3. Type of Report and Dates Covered. State whether report is interim, final, etc. If applicable, enter inclusive report dates (e.g. 10 Jun 87 - 30 Jun 88).

Block 4. Title and Subtitle. A title is taken from the part of the report that provides the most meaningful and complete information. When a report is prepared in more than one volume, repeat the primary title, add volume number, and include subtitle for the specific volume. On classified documents enter the title classification in parentheses.

Block 5. Funding Numbers. To include contract and grant numbers; may include program element number(s), project number(s), task number(s), and work unit number(s). Use the following labels:

C	- Contract	**PR**	- Project
G	- Grant	**TA**	- Task
PE	- Program Element	**WU**	- Work Unit Accession No.

Block 6. Author(s). Name(s) of person(s) responsible for writing the report, performing the research, or credited with the content of the report. If editor or compiler, this should follow the name(s).

Block 7. Performing Organization Name(s) and Address(es). Self-explanatory.

Block 8. Performing Organization Report Number. Enter the unique alphanumeric report number(s) assigned by the organization performing the report.

Block 9. Sponsoring/Monitoring Agency Name(s) and Address(es). Self-explanatory.

Block 10. Sponsoring/Monitoring Agency Report Number. *(If known)*

Block 11. Supplementary Notes. Enter information not included elsewhere such as: Prepared in cooperation with...; Trans. of...; To be published in.... When a report is revised, include a statement whether the new report supersedes or supplements the older report.

Block 12a. Distribution/Availability Statement. Denotes public availability or limitations. Cite any availability to the public. Enter additional limitations or special markings in all capitals (e.g. NOFORN, REL, ITAR).

- **DOD** - See DoDD 5230.24, "Distribution Statements on Technical Documents."
- **DOE** - See authorities.
- **NASA** - See Handbook NHB 2200.2.
- **NTIS** - Leave blank.

Block 12b. Distribution Code.

- **DOD** - Leave blank.
- **DOE** - Enter DOE distribution categories from the Standard Distribution for Unclassified Scientific and Technical Reports.
- **NASA** - Leave blank.
- **NTIS** - Leave blank.

Block 13. Abstract. Include a brief *(Maximum 200 words)* factual summary of the most significant information contained in the report.

Block 14. Subject Terms. Keywords or phrases identifying major subjects in the report.

Block 15. Number of Pages. Enter the total number of pages.

Block 16. Price Code. Enter appropriate price code *(NTIS only)*.

Blocks 17. - 19. Security Classifications. Self-explanatory. Enter U.S. Security Classification in accordance with U.S. Security Regulations (i.e., UNCLASSIFIED). If form contains classified information, stamp classification on the top and bottom of the page.

Block 20. Limitation of Abstract. This block must be completed to assign a limitation to the abstract. Enter either UL (unlimited) or SAR (same as report). An entry in this block is necessary if the abstract is to be limited. If blank, the abstract is assumed to be unlimited.

Preface

This report describes work performed under contract DACA72-87-C-0001, by Carnegie Mellon University, Pittsburgh, Pennsylvania, for the U.S. Army Engineer Topographic Laboratories (ETL), Fort Belvoir, Virginia. The Contracting Officer's Representative at ETL is Mr. George E. Lukes.

Table of Contents

List of Figures

List of Tables

1. Introduction

In July 1988 the Digital Mapping Laboratory at Carnegie Mellon University began work on the second year of a three year contract to explore the detailed analysis of aerial images containing built-up areas as supported under USAETL Contract DACA 72-87-C-001. During this year we have built on previous research, in road network extraction and in the detection and delineation of buildings using monocular analysis, accomplished during the first year of this research contract.

We have expanded our research in monocular analysis to include the detailed analysis of shadows. Our shadow analysis research has resulted in three techniques for the interpretation of monocular imagery: building prediction, grouping of related building hypotheses, and building hypothesis verification. In addition we have implemented an technique to acquire estimates of building heights using the lengths of cast shadows. High estimation of man-made structures can be accomplished even using monocular imagery.

We have continued a small level-of-effort in the extraction of road networks and have experimented with various intelligent search strategies to limit search to areas where a road finder indicates a high probability for the existence of a road segment, or in areas where road tracking has failed.

We have begun a major initiative to explore automatic methods for scene registration in complex aerial imagery and have developed a new feature-based matching algorithm based upon hierarchical waveform analysis. Our work in stereo analysis complements the monocular feature extraction research and provides a basis for the integration of explicit three-dimensional information into built-up area analysis. Our major accomplishments have been to identify a family of techniques for automated scene registration, to improve the state-of-the-art in automated stereo matching using complex urban scenes by applying a combination of area-based and feature-based methods, and to develop tools and techniques for automated performance evaluation using three-dimensional groundtruth.

Finally, we have supported a modest effort to investigate the utility of share-memory multi-processors for high-level vision. Our focus has been the exploration of task-level parallelism for a knowledge-based system that has been used to interpret airport and suburban house scenes. We have achieved near linear speedups on an Encore multimax processor for the most computationally intensive component of the system.

In the following sections we detail the problems and accomplishments in each of these areas achieved over the last year. We believe that progress has been steady and that the work in stereo analysis and shadow analysis has greatly improved our suite of techniques for built-up area analysis. In the final section we discuss our plan for the final year research and detail published reports, technical talks, and other tangible accomplishments funded under this research contract.

2. Detection and Delineation of Buildings Using Monocular Cues

Determining the location, shape, size, and type of man-made structures and generating the description of the underlying terrain are two of the most significant components of any topographic map. Roughly speaking, a two-dimensional image provides two classes of cues from which we can infer the structure of the underlying three-dimensional physical world: pixel intensities and intensity changes within small local neighborhoods. Some objects, such as shadows, may be characterized by their intensity values. Other objects, e.g., vehicles on a road, are manifested in the image mainly by abrupt intensity changes, called intensity edges. Edges carry most of the information in an image, and are relatively robust to changes in image contrast and radiometry. Intensity values are much less stable; it is impossible to find a shadow intensity value that is common to all images, or even many of them. Our approach is to use intensity edges to estimate relevant intensity values on a per image basis.

This section describes how we extract building information, organized as hypotheses, from aerial imagery. Given an image and knowledge of the ground sample distance of a scene, we can utilize intensity and edge cues to generate building hypotheses in the form of polygons that coincide with buildings when superimposed on the image. One method, BABE (Builtup Area Building Extraction), analyzes intensity edges, estimates the shadow intensity and illumination direction, and produces a set of building hypotheses. Other techniques described in Section 2.2 use an estimate of shadow intensity produced by BABE to find shadow regions and to hypothesize the buildings that cast those shadows.

2.1. Building hypothesis generation

BABE examines intensity edges in the image for cues to building locations, generates a large set of plausible building hypotheses, and then uses more detailed analysis of image domain cues to verify each original hypothesis. In our current implementation, we assume the following constraints:

- The camera is (nearly) vertically oriented and can be modeled using a standard frame camera model.
- Most building shapes can be approximated by rectangular boxes.
- Buildings cast shadows, and shadows are darker than the corresponding roofs.
- Most buildings have smooth roofs.
- The minimal building size is given, or can be derived from the ground sample distance and the shortest side is no less than 5 pixels.

In contrast to other building detection programs, BABE does not assume that:

- The buildings are brighter than the average background intensity.
- No building is darker than *any* shadow.
- The shadow intensity is given.
- The illumination angle is given.

Figure 2-1 is the well known Los Angeles Airport scene from the work of Heurtas and Nevatia [7]. We will illustrate several problems that are encountered by any intensity-based analysis program on this relatively simple scene, and then show some results on a more complex scene. There are three types of edge problems that BABE must accommodate:

1. Missing edges: Analysis based on edge detection must distinguish between object edges and texture edges. Usually, the distinction is based on a measure of edge strength. Contrast can be such a measure, but there are often some object edges that are weaker than texture edges. It is desirable both to reject texture edges and to retain object edges. In the balance between these contradicting requirements some object edges are bound to be missed. Therefore analysis procedures cannot assume that all building sides are represented by intensity edges. An example of this problem may be seen on the left side of the large hangar building in the center right of Figure 2-1. There is virtually no left vertical edge to indicate where the building ends and the ground begins.

2. Edge fragmentation: Often a single real-world object is represented by a number of disconnected edge fragments. Unless we are able to associate or aggregate these fragments with each other, they will be interpreted as distinct objects. In many cases such fragmentation prevents analysis from recognizing the correct object properties, resulting in the inability to extract the object. In Figure 2-4 edge fragmentation occurs at the top of the large hangar building, resulting in a hypothesis that covers only half of the actual building.

3. Corner problems: Virtually all edge detectors have an implicit assumption that edges are piecewise linear. This causes distortion in edge detection around corners, resulting either in rounded corners, or in unlinked edges around the corner. Corner problems occur in many of the buildings in our example and are especially clear along the bottom of several of the elongated buildings in the left center of Figure 2-1.

Given intensity edges, BABE attempts to find candidates for building corners. First, edges that contain corners, even rounded ones, are broken. Then every two edges that are within a specified distance (10 meters in the current implementation) are examined to determine whether they approximate a 90° angle. Measuring the angle between two discrete curves that are not perfectly straight is not trivial. In practice, BABE balances local and global curvature to find the angle nearest to 90°. Figure 2-2 shows the corners found for the image in Figure 2-1. Note that there are many corners that have no relationship to buildings or other man-made structures. These accidental alignments occur frequently in high resolution aerial photography and greatly complicate analysis. However, the number of corners found are far less than the number of discrete line segments in the scene, and they form a good abstraction around which BABE can go back into the image to look for meaningful line/corner alignments.

The next step in scene analysis is the formation of boxes consisting of intensity edges linked by corners. Since it is likely that some of the building edges were not found by the edge finding step, BABE works with partial building components, hallucinates the missing edges, and tests whether the hallucinated edges have some support from the underlying imagery. The reason that such an approach can work, and is actually quite powerful, is that the process of verifying an edge uses more context than finding one. Therefore, it is possible to verify the existence of an edge even in places where traditional edge finding programs fail to find them. BABE takes every three edges linked by two corners and completes them to form a box hypothesis. A box hypothesis can be a parallelogram or trapezoid and is evaluated with respect to image intensities. Many competing hypotheses can be generated from sets of

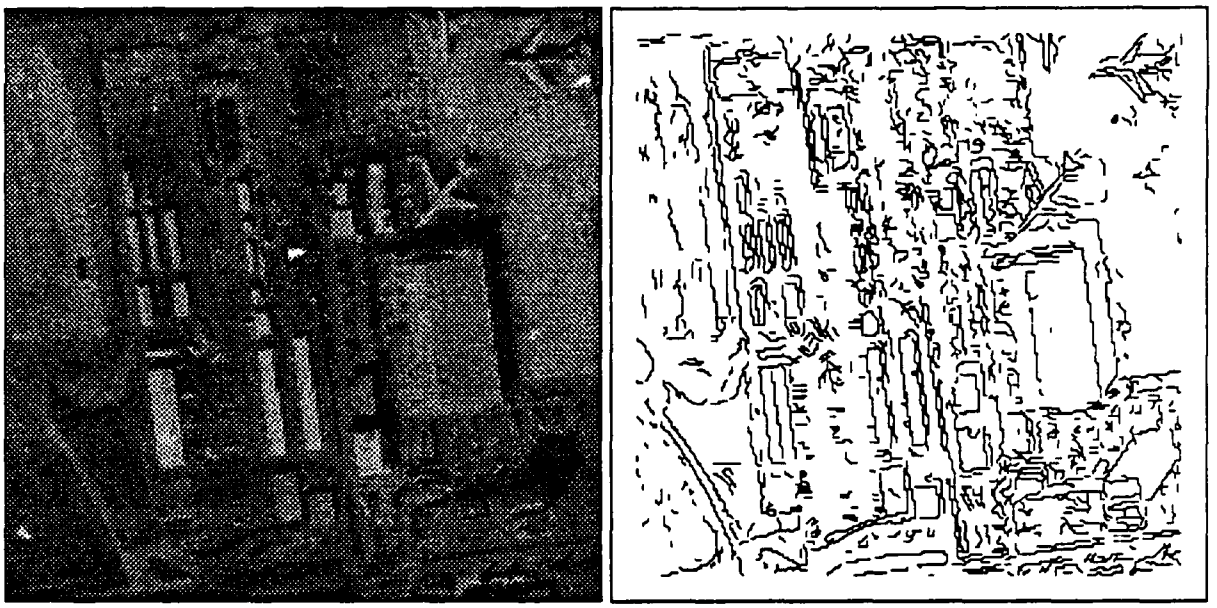

Figure 2-1: LAX Airport Edge Detection

shared corners and edges, and if there are more than two generated, each is evaluated with respect to edge support. At most the best two of each set of competing boxes are passed on to the final shadow verification step. Boxes are considered competing if there is a way to link their edges by corners. Figure 2-3 shows the boxes generated for our example. There are almost 600 boxes generated for this image of 30 buildings.

Figure 2-2: Extracted Corners

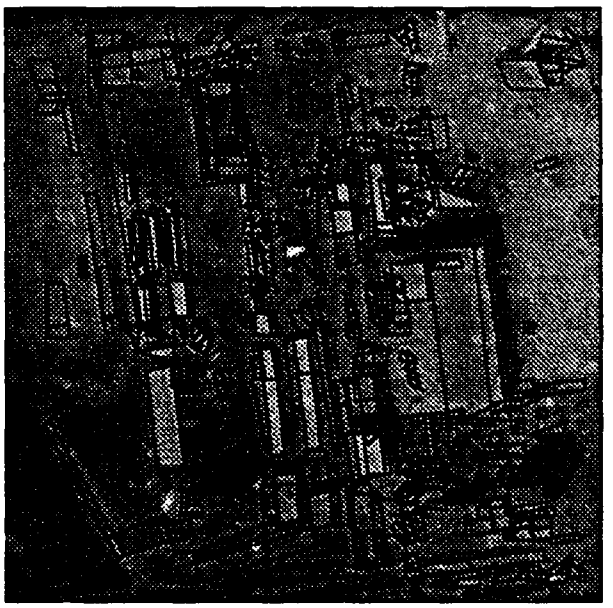

Figure 2-3: All Box Hypotheses

Finally, building hypotheses are selected from the boxes. This process has three steps:

1. Estimate the shadow intensity: For every box the intensities along the inside and outside of each edge are computed. Boxes having an intensity gradient consistent with darker intensities on the outside edge are considered as possible buildings. They vote for a global estimate of the shadow intensity and the non-shadow intensity. A weighted threshold is used to determine a shadow intensity

Figure 2-4: Accepted Building Hypotheses

estimate and standard deviation that optimizes occurrences of shadow intensities outside and inside of these boxes.

2. Estimate the illumination direction: Once the shadow intensity is computed, it is possible to find an illumination direction that best explains the occurrences of shadow intensities outside boxes. The direction that explains the most boxes is used as the illumination direction. The illumination angle estimated for the LAX scene is 151°, 3° off from the 148° direction determined manually. It is possible to provide the illumination angle to BABE, in which case the calculation of the shadow intensity becomes more robust since we can directly determine those building sides that should cast shadows.

3. Select boxes that are consistent with the illumination angle. In this step the strength of the shadow is combined with edge quality and roof uniformity to calculate a score for every box, and an adaptive threshold is used to reject boxes with low scores.

Figure 2-4 shows the final building hypotheses suggested by BABE for the LAX airport scene. Note that while a large number of hypotheses are generated, the local use of shadow evaluation allows us to prune hypotheses that do not appear to be casting a shadow. In the following section we will see how a more elaborate use of shadow region analysis allows us to generate building hypotheses directly from the detected shadows. Figures 2-5 and 2-6 show the results of this technique on a more complex industrial site. This scene violates several of our underlying assumptions in that many roofs are structured, and corners detected on those structures cause fragmentation of building hypotheses. Even so, the sun angle computed by BABE is reasonable, and a significant portion of the buildings are detected. However, it is clear that for this scene, the cast shadows convey a great deal of structural information. The following section presents several methods to utilize shadow information in scene analysis, particularly in combining BABE hypotheses with an independent evaluation

and grouping mechanism.

Figure 2-5: All Box Hypotheses in Industrial Scene

Figure 2-6: Selected Building Hypotheses

2.2. Shadow Analysis

One important property of man-made structures, particularly buildings, is that, because they have height above the ground, when illuminated by the sun, they cast shadows. Therefore, a second strategy for monocular analysis is to utilize the relationship between cast shadows and the associated man-made structures. The use of shadows to perform height determination in aerial imagery has a long tradition in photogrammetry and manual photo-interpretation. Our work has produced four methods for using shadows in monocular analysis. Shadows can be used to predict the location of buildings, to estimate the height of objects, to verify hypotheses produced by other methods, and to group together hypotheses that share a common shadow [8]. The last two techniques, verification and grouping, have been used as an independent source of information to validate BABE hypotheses. In this section we will show how building hypotheses can be generated by the analysis of cast shadows and how height determination can provide us with an estimate of the three-dimensional scene structure without direct stereo analysis.

Figure 2-7 shows a suburban scene containing a variety of building types with many different heights, shapes, orientations, and roof structures. Given an estimate of the shadow intensity produced by BABE based upon analysis of all box hypotheses, we perform shadow extraction through a simple set of image processing techniques including image smoothing, thresholding, and connected region extraction. Figure 2-8 shows the outlines of all shadow regions produced for this scene. The next step is to analyze the geometry of the shadow regions to determine those edges that form the building/shadow boundary. We assume that sun angle and direction are known. This is reasonable given that we know the date, time of day, and latitude/longitude of the photography. In fact, this information is retrieved on a per image basis from the CONCEPTMAP database [12, 14].

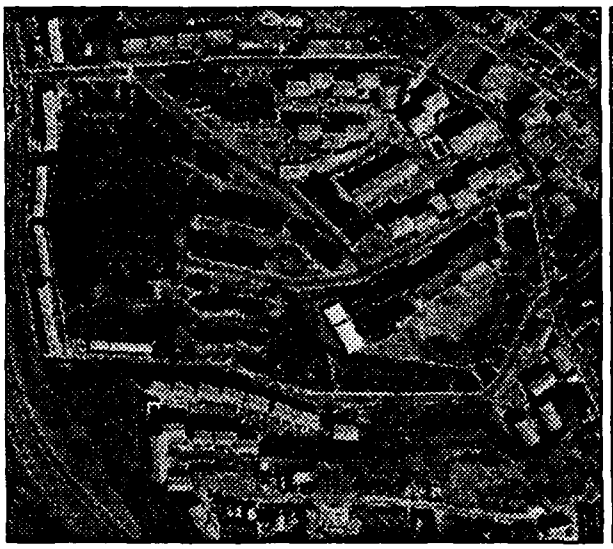

Figure 2-7: Suburban Scene

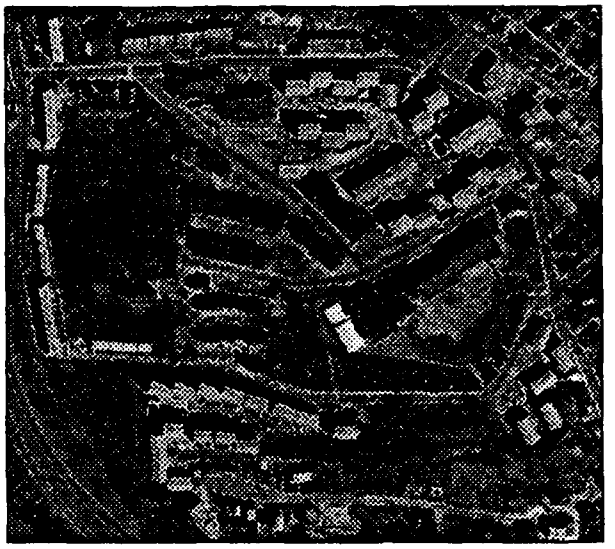

Figure 2-8: Shadow regions extracted

We examine each edge of the smoothed shadow region and test whether or not it is on the sun-side of the shadow. This test involves back projecting the edge's midpoint toward the sun along the sun direction vector a nominal distance, normally one pixel. If the projected point falls outside of the original shadow region, then it is labeled as a sun-side edge. Otherwise, it is labelled as a non-sun-side edge. We then group together sequences of sun-side edges in each shadow region, allowing for some noise in the sequence, into building/shadow edges. Finally we perform analysis on these building/shadow edges to determine the location of a characteristic 'L' shape that coincides with the building corner. This is often complicated when the buildings are oriented in the direction of the sun, or when the original shadow regions are not extracted as a single region. Corners of interest are those whose concave sides are oriented toward the sun. Figure 2-9 shows the building/shadow boundaries that are generated from the shadow regions in Figure 2-8.

Once the corners have been generated from building/shadow analysis, it is a simple matter to extend the corners into parallelograms based on the length of the side of the building/shadow edges. Each of these parallelograms is a hypothesis for the occurrence of a man-made structure. Because buildings have height, they often occlude parts of their own shadows in complex scenes. As a result, the actual delineation may be displaced. The area of coverage can also be adversely effected when buildings share a common boundary, and the shadow is cast on an adjacent building rather than on the ground. Figure 2-10 shows the building hypotheses automatically generated using shadow analysis. While it is evident that further processing is necessary to precisely delineate the buildings and interpret the various three-dimensional roof structures, the use of shadow information to generate building hypotheses provides an important model-independent method for detection and delineation that complements the BABE techniques previously described.

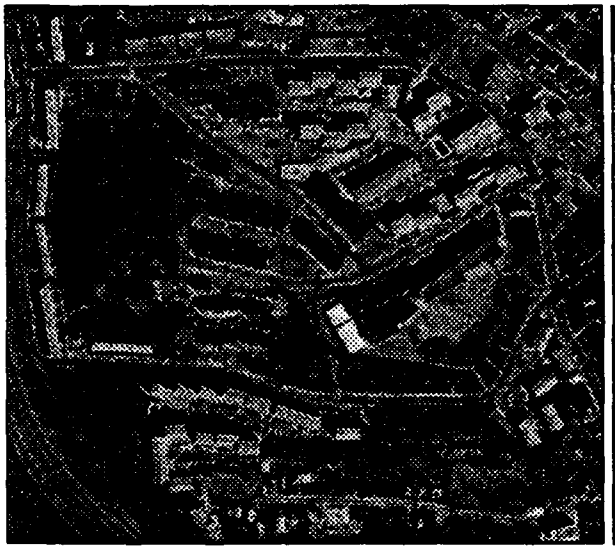

Figure 2-9: Shadow / building boundaries

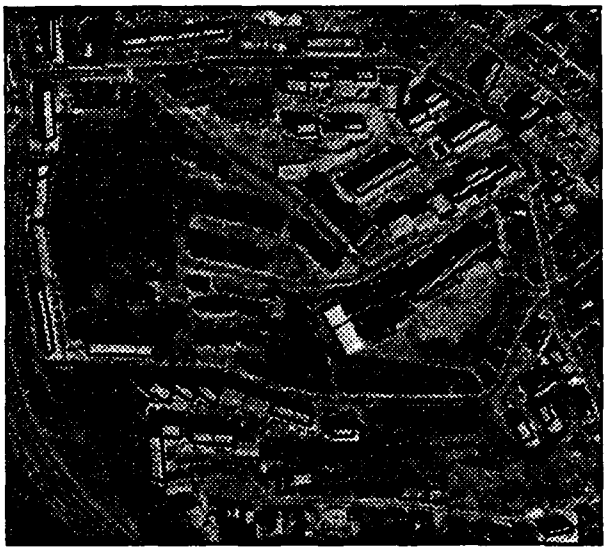

Figure 2-10: Hypothesized buildings from cast shadows

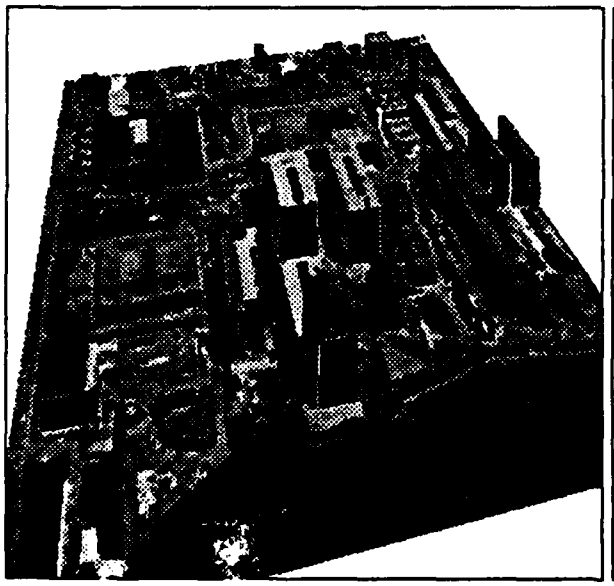

Figure 2-11: Perspective Scene With BABE Delineation

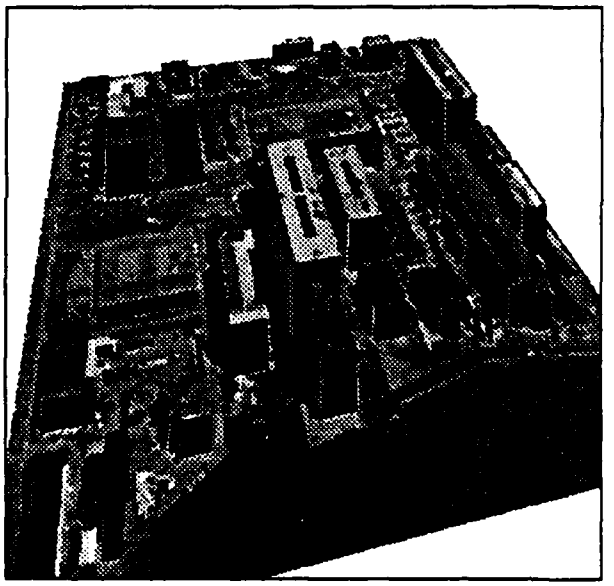

Figure 2-12: Perspective Scene With Manual Delineation

As we have mentioned, given the sun angle, shadow length, and image scale parameters, we can generate a good estimate of building height. Such an estimate provides an alternative to direct methods, such as stereo mensuration. For each building hypothesis, we can estimate the shadow length by walking from the building/shadow boundary along the sun direction vector. We classify each pixel as "dark" or "bright" with respect to the original estimated shadow intensity. A sequence finder is used to find the imperfect sequence of "dark" pixels that correspond to the shadow region. The far end of that sequence is called the shadow terminator. This walk is performed for every point along each building/shadow edge and the length of the shadow at each point is recorded as the number of steps during the walk from the building shadow edge to the shadow terminator. After the shadow region is delineated, a

weighted average length is computed. Figure 2-11 shows a three-dimensional perspective scene generated by taking each of the BABE building hypotheses in Figure 2-6 and performing height estimation using shadow analysis. Figure 2-12 shows the same perspective scene except each of the buildings has been manually delineated while the height is automatically estimated using cast shadows.

As we will see in the subsequent discussion of stereo matching in complex urban scenes, automated stereo analysis is difficult, and methods such as shadow analysis can provide initial estimates of disparity to guide matching or can be used in cases where stereo matching is not possible.

2.3. Discussion

We have shown several methods for obtaining estimates of building detection and delineation using intensity-based structure generation as well as shadow analysis. Our goal is to develop several independent sources of information to allow for redundant and robust hypothesis generation. One should view this strategy as a direct result of our belief that no *single* scene interpretation method is likely to function across a wide range of imaging conditions. Further, efforts to increase the performance level of a single technique generally far outweigh their benefits even assuming such techniques could avoid brittleness encountered in highly developed and specialized algorithms. Our rule of thumb is that each 10% increase in performance requires as much effort as achieving the previous level of performance. Crudely speaking this geometric progression of effort drives us toward system architectures that accommodate failures by utilizing multiple cooperative methods having different failure modalities and lower intrinsic performance capabilities.

Such cooperative methods can be either loosely coupled or closely coupled. In loosely coupled systems the comparison of results occurs after each method has attempted to generate a complete scene description. In closely coupled systems, a finer grain of communication allows for low-level interactions at the point that problems or failures are detected. In the following section on road network extraction, we will see an example of high resolution road tracking using a closely coupled communication mechanism.

3. Road Network Extraction

The automatic extraction of roads from aerial imagery has been an active research subject in computer vision for over a decade. Approaches have varied from multi-spectral analysis, linear feature detection, and structural analysis. Most often the distinction between detection and delineation has not been made explicit. That is, finding roads and accurately describing their properties are treated as the same problem. In our research we have explicitly divided the task of automated road network extraction into three distinct phases:

- *Road Finding:* The whole area-of-interest is processed, and sequences of possible road points are found. Road finding generates road seed hypotheses that are used as initial starting points for road tracking.
- *Road Tracking:* Road tracking takes the center lines defined by the road seeds and attempts to extend the road by tracking the road surface and road boundary.

Specific knowledge about road shape and surface intensity profiles is used to extend them. Typically, only a small neighborhood around the end of the road is processed. The results of road tracking produce a symbolic description of the road centerline, the road width, and the location of various road features, such as overpasses, intersections, surface material and road width changes.

- *Network Construction:* The result of road tracking using multiple starting points results in a set of completely and partially tracked roads. These road hypotheses are examined for overlaps and intersections, and a graph that represents the overall network structure is constructed.

The following sections describe the issues and techniques involved with road finding, road tracking, and road network extraction in more detail.

3.1. Road Finding

The goal of road finding is to provide the road tracker with highly reliable starting points from which the tracker can initialize its road surface and boundary model [1]. In order to perform this initialization the road finder must generate an accurate estimate of the road width and the road center. A distance of 20 to 40 pixels is generally sufficient to support road tracking initialization. Our road finder uses an edge-based algorithm to define road-center-hypotheses (RCH's) as anti-parallel intensity edges and groups such points to produce continuous, smooth road seeds. Figure 3-1 shows a small portion of a suburban house scene containing roads, driveways, and houses. Figure 3-2 shows computed intensity edges and the resulting individual road center hypotheses.

Ideally, roads would have continuous runs of RCH's on them, but in practice, intersections, noise, and shortcomings of the edge finding process cause defects in continuity. Therefore, we look for collections of supporting RCH's that form imperfect sequences [2], to overcome the continuity defects. This results in relatively long runs of RCH's on roads, and mostly short ones outside roads. The left side of Figure 3-3 shows road seeds overlaid on the RCH's, and the right side shows them overlaid on the image.

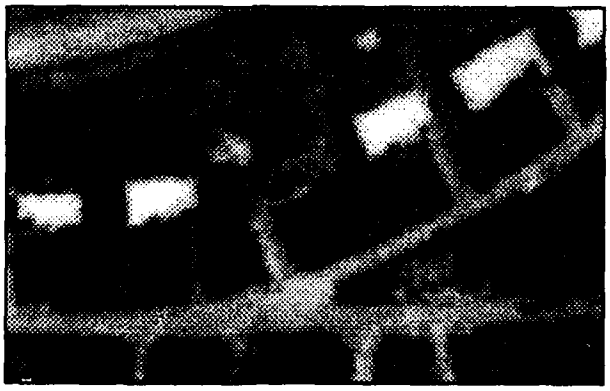

Figure 3-1: Suburban House Scene

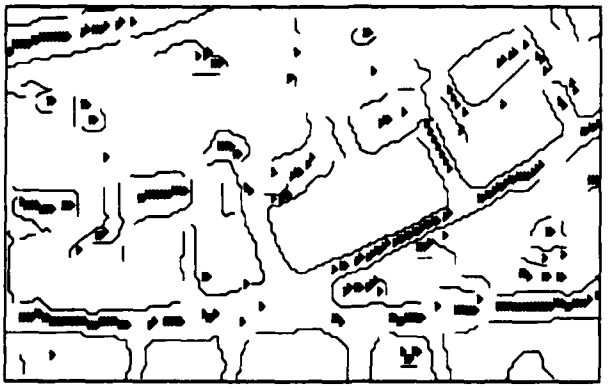

Figure 3-2: Road Center Hypotheses

Road seeds are further analyzed to remove smoothness violations; fragments are linked together, and short seeds are rejected. Figure 3-4 shows the road seeds generated for a more

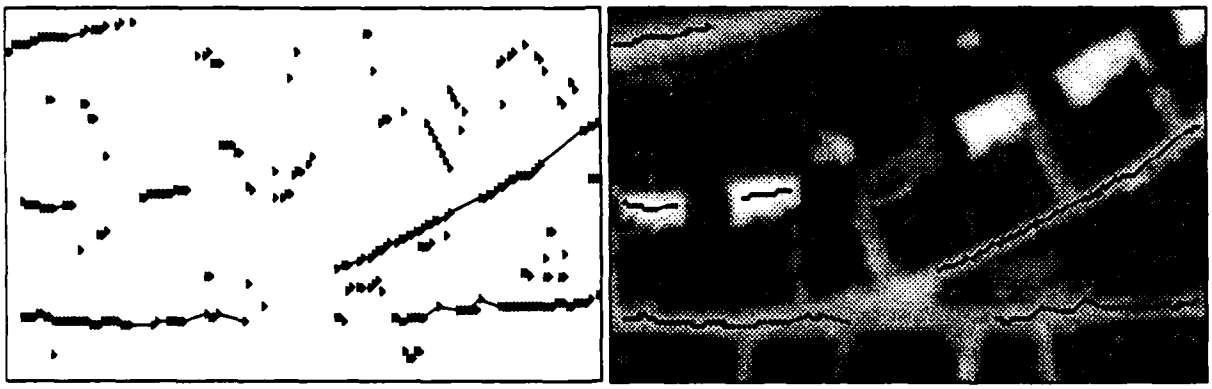

Figure 3-3: Road Starting Hypotheses

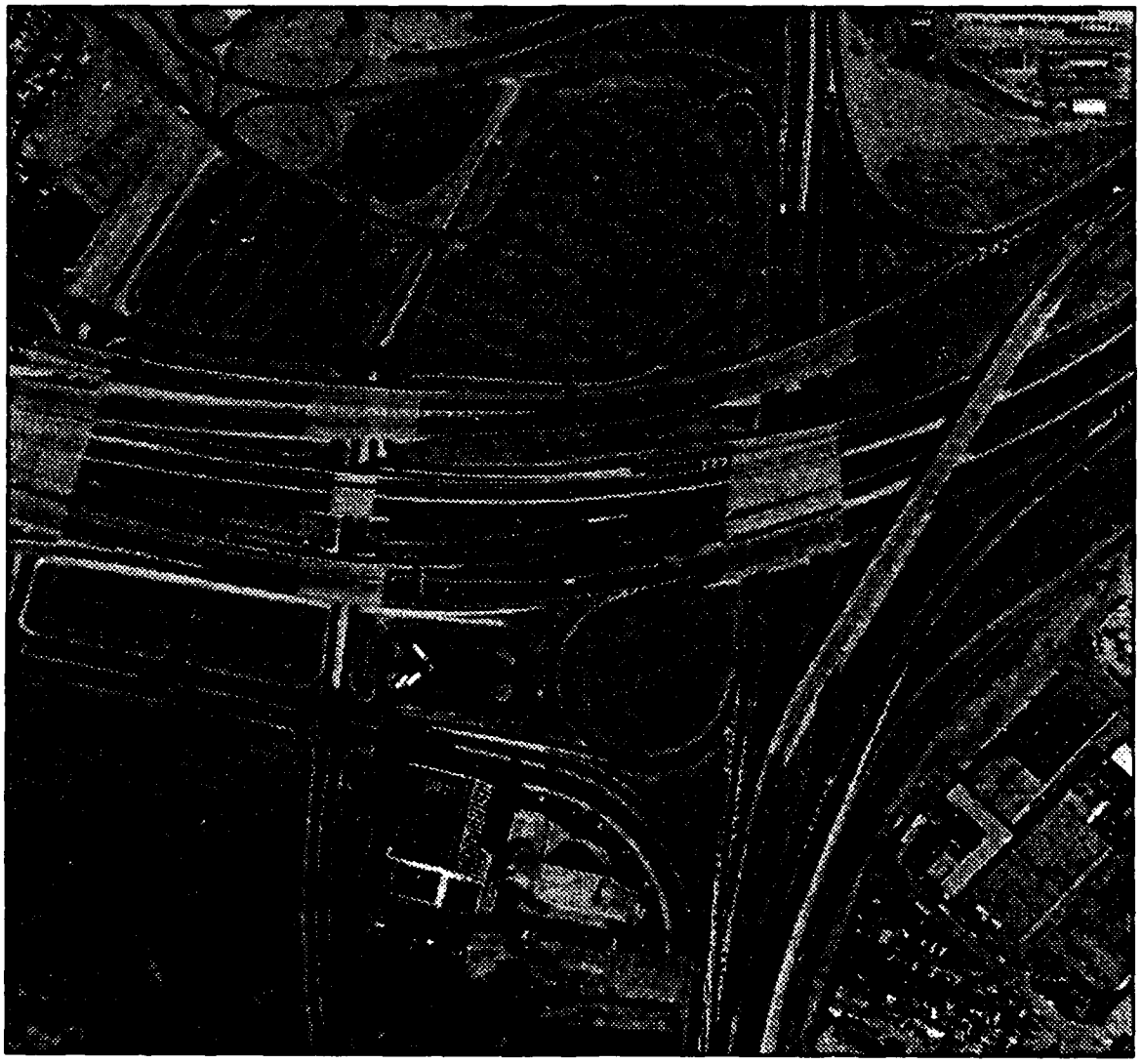

Figure 3-4: Road finding in local area-of-interest

complex scene containing a variety of multiple land roads. One drawback to this and other analysis techniques based upon intensity edges is that the structure and variability within roads can be quite complex. In some cases either the intensity edges are missing, or they

occur in unexpected configurations, such as when edges occur inside the road. In such cases local methods based upon linear feature detection are insufficient, and more detailed knowledge about roads must be brought to bear. The next section describes a road follower that takes road center hypotheses as its input and uses knowledge about road shape and road intensity profiles to extend or reject initial road hypotheses.

3.2. Road Tracking

Our research into effective road tracking has resulted in a system that uses multiple cooperative methods for extraction information about road location and structure from complex aerial imagery. It uses a multi-level architecture for image analysis that allows for cooperation among low-level processes and aggregation of information by high-level analysis components [15].

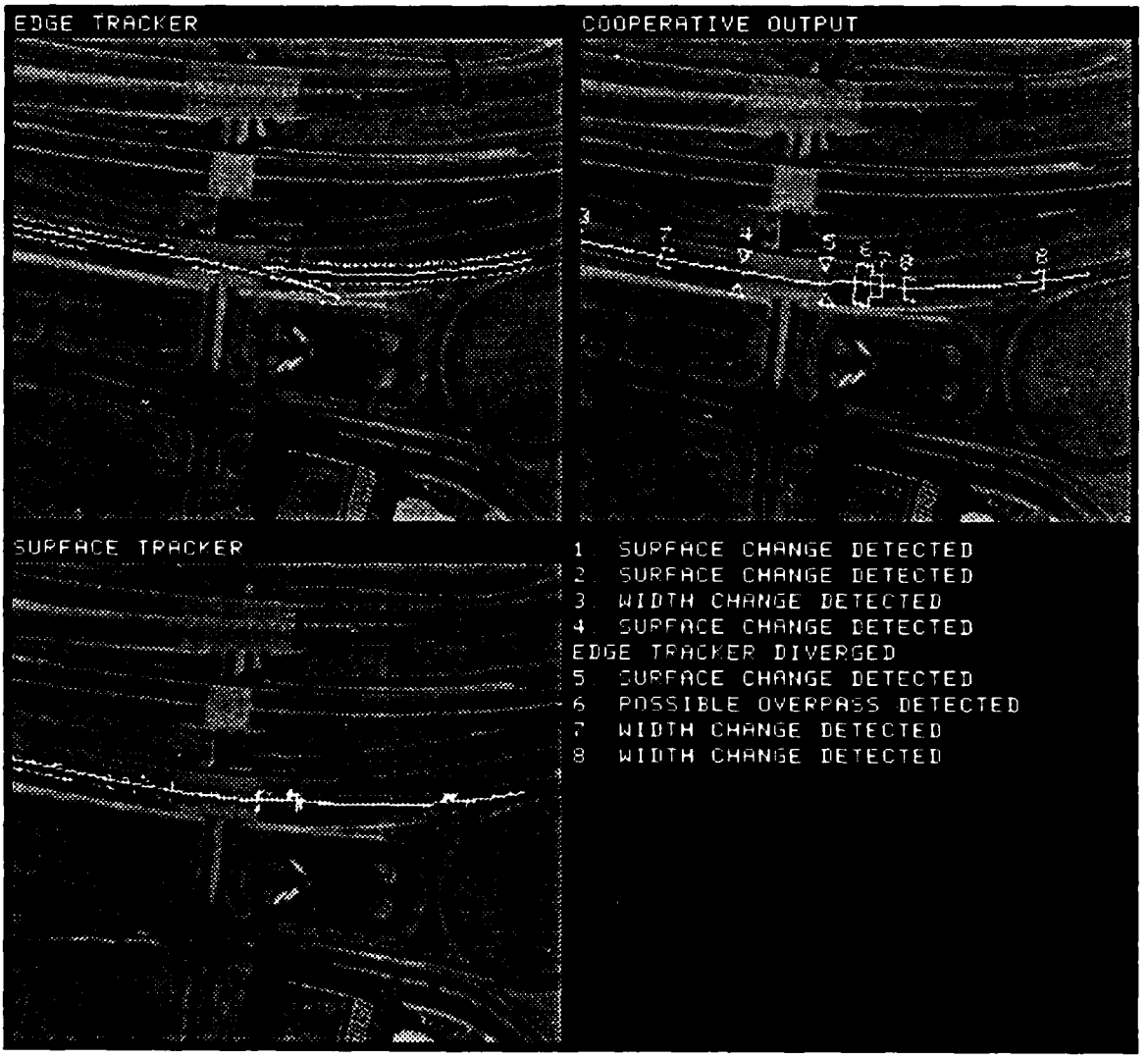

Figure 3-5: Road tracking in local area-of-interest

Two low-level methods have been implemented; road surface texture correlation and road edge following. Each low level method works independently to establish a model of the

center line of the road, its width, and other local properties. Intermediate-level processes monitor the state of the low-level feature extraction methods and make evaluations concerning the success of each method. They also extract various road properties such as width changes, surface material changes, and overpasses. As a result of these evaluations, one tracking method may be suspended due to apparent failure and restarted from the model generated by other successful trackers. Finally a high-level module generates a symbolic description of the road in terms of various attributes of the road such as center line, road width, surface material, overpasses, and an indication of potential vehicles on the road. This description is available in both map and image coordinate systems and can be used to generate a textual description of the road.

Figure 3-5 shows a snapshot of the tracker operating on one of the initial road starting points shown in Figure 3-4. We see the results of the independent edge and surface trackers as well as the cooperative output fused by the intermediate level analysis. Note that the edge tracker diverged off the road on the bridge deck due to a lack of continuity of edges across the concrete surface. However, the high-level control successfully restarted the edge tracker using the path model provided by the surface tracker. The cooperative output shows the types of information that are extracted by the tracker, as well as the annotation of where in the image various features appear. Using a frame camera model, we are able to convert these pixel locations into geographic coordinates to generate an accurate cartographic description of the road.

3.3. Road Network Construction

One interesting aspect of the partitioning of road finding from road tracking is that one can opportunisticly use a small area-of-interest as a starting area to track roads along the full extent of the scene. Figures 3-4 and 3-6 are good illustrations of this situation. We can see that road tracking begun in Figure 3-4 has actually delineated roads in Figure 3-6 far outside of the original area of interest. Strategies such as performing road finding along the perimeter of the image can greatly decrease the amount of processing necessary to extract complete road networks.

The road network system maintains a history structure that allows it to avoid multiple tracking over the same swath of the image using the same road width estimate. Nevertheless, the results of road tracking must be examined for overlaps and intersections. A graph representing the road network structure is constructed. Overlaps may occur when roads that had distinct road seeds merge. The determination of an overlap must take into account fluctuations in the road tracker's path. Figure 3-6 shows results of finding, tracking, and network construction. The white box marks the area in which road finding was started, black lines mark the starting points found to actually be on a road, and white lines mark the extension of these starting points using road tracking. In this example, whenever the road tracker could not continue, the road finder was invoked. If a road seed was found that overlapped the tracked road, tracking was resumed from that seed.

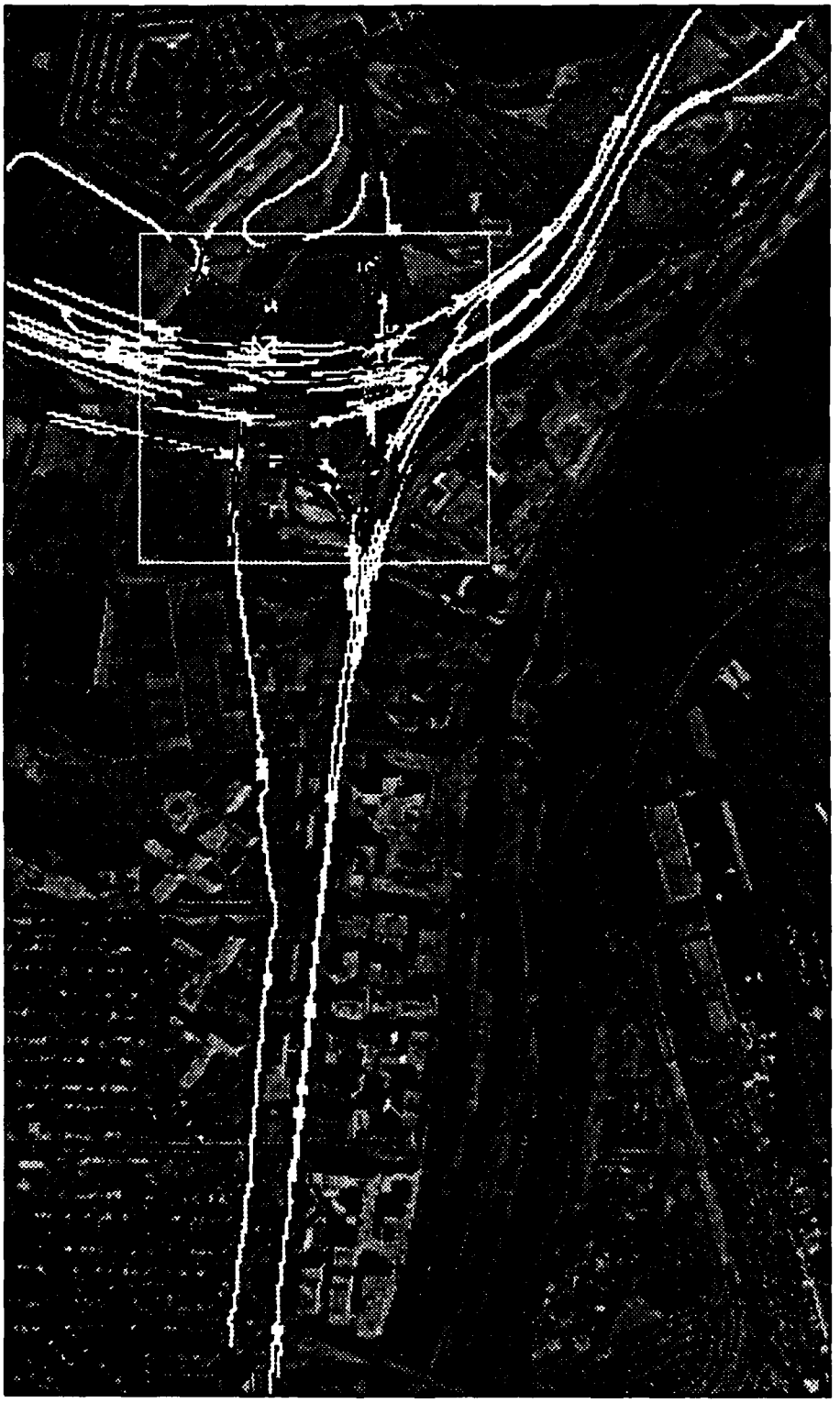

Figure 3-6: Complete Extracted Network

3.4. Discussion

Our experiments with road network extraction have focused on the use of imagery with high spatial resolution where details of the road surface were clearly evident. This is in contrast to other approaches, particularly using lower resolution imagery such as Landsat TM or SPOT. The main issue revolves around whether road network extraction can (should) be viewed as linear feature extraction, that is, the detection of lines patterns in digital imagery. Such work often lumps together roads, railroads, runways, drainage and terrain patterns as being "linear features". We do not believe that this view is on the path to high performance feature analysis systems. For example, there are intrinsic structural properties that can be used to accurately detect and delineate railroads that are irrelevant within the context of roads. Our style of knowledge-intensive research requires high spatial resolution so that our programs can not only detect the location of the road, but also use structural analysis of intersections, overpasses, and surface material changes to refine the feature extraction process.

Road tracking is also instructive in that it provides a crisp example of the utility of closely coupled cooperative feature extraction methods to improve aggregrate performance. Our experiments with 70 test image fragments showed that the correlation tracker performed as well as the cooperative tracker only 68% of the time when tested in isolation. Further, in only 35% of tests did the edge tracker perform as well as the combined tracker.

In the following section we discuss the use of multiple methods to perform stereo analysis using loosely coupled techniques where comparison is deferred until each method has performed a complete estimate of scene structure.

4. Stereo Analysis For Urban Scenes

Up until this point we have discussed techniques for monocular analysis of aerial imagery. We have seen that good cues concerning building height can be obtained by shadow analysis even from a single image. However, the traditional method for obtaining an accurate three-dimensional model of the terrain and man-made structures involves the use of stereo-pair imagery. Algorithms for stereo correspondence can be grouped into two major categories: area-based and feature-based matching [3]. Both classes of techniques, area-based and feature-based, have advantages and drawbacks that primarily depend on the task domain and the three-dimensional accuracy required. For complex urban scenes, feature-based techniques appear to provide more accurate information in terms of locating depth discontinuities and in estimating height. However, area-based approaches tend to be more robust in scenes containing a mix of buildings and open terrain.

4.1. Scene registration

Accurate scene registration is critical to stereo matching. Stereo pairs are usually matched with the assumption that corresponding points are on the same scanline in each image and the displacement between the points, or disparity, corresponds to the relative height of the three-dimensional scene point. This epipolar geometry is often imprecisely achieved and requires

careful local registration even after the scenes have been coarsely aligned.

We have been experimenting with a variety of techniques for automating local registration prior to stereo matching. Each involves the use of different types of control points for registration. While control points can be manually supplied as landmarks through a spatial database, it is more interesting to see how we might automatically derive ground control using feature extraction and matching processes. Candidates for automatic control point generation include shadow corners, shadow regions, BABE box hypotheses, uniform intensity regions, and elongated intensity line pairs.

There are some important criteria for automated control point selection. First, since the elevation of the control points is not known, it is important that the set of control points selected lies approximately in the same elevation plane. Second, the selection of control points should not rely on a single type of scene domain feature, such as road intersections, since not all control point features are abundant in all scenes. For example, in urban scenes there are often many buildings and shadow regions available as candidate control points, and they are usually well distributed throughout the imagery. However, in airport scenes elongated line pairs and uniform intensity regions appear to be a better choice. In any case we use an iterative selection algorithm that converges to a consistent set of control points that are usually a small subset of all of the possible matches in the stereo pair.

Another advantage of using multiple features for control point estimation is that the results of feature matching can be used to estimate disparity range of the scene. Once the scene is registered, all matched features can be remapped to the new coordinate frame. It is then possible to calculate the disparity of each feature. Since all features are not at the same height, we automatically have a rough estimate of the disparity range for this scene. This disparity range estimate is directly used by the stereo matching algorithms to control search for corresponding points and can greatly reduce initial matching errors.

Figure 4-1 shows the superposition of BABE results for the left and right image according to the coarse registration performed through our spatial database, CONCEPTMAP [12, 14]. A subset of the boxes having consistent displacement is matched and the center-of-mass of the matched boxes is used to define control points. Figure 4-2 shows the superposition of BABE results using the refined registration from Figure 4-1. In many cases we have been able to automatically reduce the row offset error to sub-pixel accuracy from an initial displacement of 12 to 15 rows in the coarse registration [17].

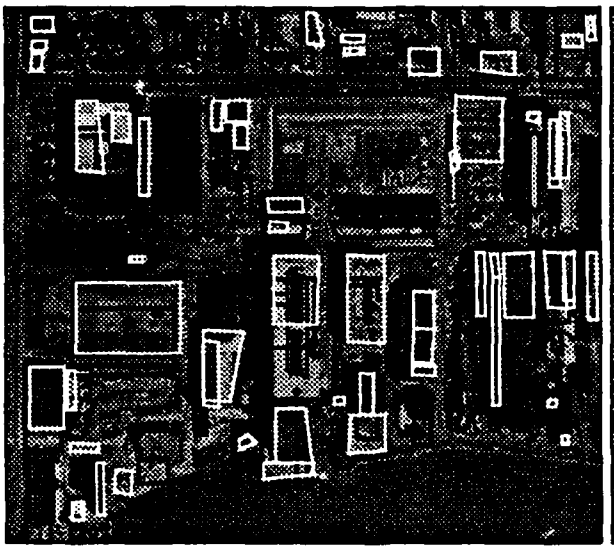

Figure 4-1: Automatic control points using structure matching

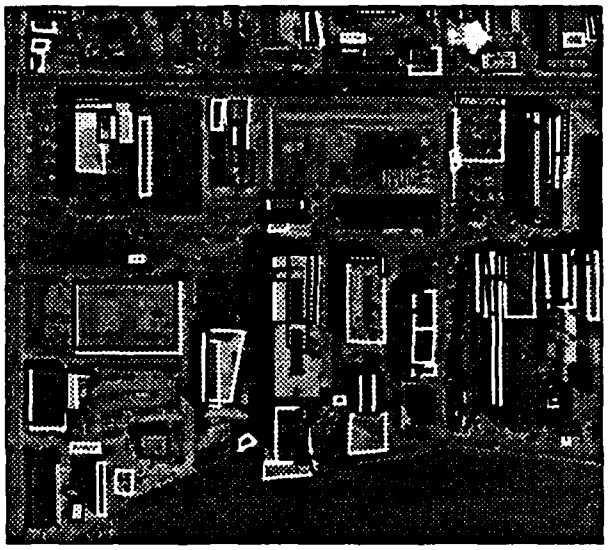

Figure 4-2: Superposition of structures using fine registration

4.2. Stereo Matching

As we have stated, complex urban scenes pose difficult matching problems for computer vision algorithms. We do not believe that any one technique is likely to be robust enough to perform well in the diverse set of scenes found in urban areas. For this reason we have developed two stereo matching algorithms that have complementary behaviors. S1 is an area-based algorithm and uses the method of differences matching technique developed by Lucas [10, 13]. S2 is feature-based using a scanline matching method that treats each epipolar scanline as an intensity waveform. The technique matches peaks and troughs in the left and right waveform. Both are hierarchical and use a coarse-to-fine matching approach. Each is quite general, as the only constraint imposed is the order constraint for the feature-based approach. The order constraint should generally be satisfied in our aerial imagery except in cases of hollowed structures.

4.2.1. S1

Both matching algorithms assume the epipolar geometry, but they have different sensitivity to its accuracy. The S1 area-based approach uses a hierarchical set of reduced resolution images to perform coarse-to-fine matching on small windows in the two images. At each level the size of the windows for the matching process depends on the resolution of the reduced image. An initial disparity map is generated at the first level. Subsequent matching results computed at successively finer levels of detail are used to refine the disparity estimate at each level. Therefore the amount of error in the scene registration that can be tolerated by this matching algorithm depends on the size of the matching windows. However, since there is a relationship between the matching window size and the level of accuracy, simply using larger matching windows may not be desirable.

To accommodate large disparities, we use a hierarchy of different spatial resolutions. Starting with a reduced resolution dataset we compute an initial estimate of the scene

disparity. With this estimate of disparity as an initial starting point, we can better refine our estimate than if we had begun matching at a coarser level. The disparity range of the scene can be used to estimate the number of different spatial resolutions, the number of levels for each resolution, and the size of the smoothing windows and scanning overlap at each level. A good estimate of the disparity range can be provided by shadow analysis, BABE box matching, or external knowledge of the terrain. We have found that good estimates of the disparity range are necessary to achieve reasonable results.

4.2.2. S2

The S2 feature-based approach matches epipolar lines in the left and right image. It uses a hierarchical approximation of the intensity waveforms to match peaks and valleys at different levels of resolution. To avoid mismatches it uses inter-scanline consistency that enforces a linear ordering of matches without order reversals. It also applies an intra-scanline consistency that considers the matches in adjacent scanlines. Application of intra-scanline constraint is used to increase the confidence of matches found to be consistent across multiple scanlines and to delete improbable matches.

The features used for matching are the intensity and gradient extremities of the scanlines. The matching criteria is simply the similarity between two extremities. Intensity extremities are easier to match than the gradient extremities, because intensity extremities vary in size and shape more so than the gradient extremities. However, intensity features may not correspond to the position of physical objects in the scene, so the gradient, the derivative of the intensity peak, is matched. Figures 4-3 and 4-4 show the left and right waveform for a single image scan-line. The horizontal black line is the scan-line being matched, the horizontal white line is the interpolated disparity profile for the scanline, and the black waveform is the gradient waveform. Minima and maxima that have been matched are marked in white.

Intensity features are matched hierarchically. In other words, S2 matches the most significant features first, such as points with highest and lowest intensity values. Points with succeeding values are matched later using matches at the previous coarse level as constraints. Due to the locality of matching algorithm, the optimum matches at the waveform level might not be desirable or correct from a global point of view. It is precisely for this reason that inter- and intra-scanline consistency constraints are imposed during the intensity matching phase. Inter-scanline consistency simply assumes that disparity should be nearly continuous across the scanlines. Intra-scanline assumes continuity along the scanline, unless there are strong supports for the disparity jump. The intensity waveform matches are then used to constrain allowable matches during position refinement using the gradient waveform.

One key issue in feature-based stereo matching is the interpolation process. Because we are obtaining depth estimates at sparse matching points, we must fill in depth estimates in a consistent manner in order to achieve a complete disparity estimate. While there has been much work in surface interpolation techniques, such as regularization, we still have not found a satisfactory technique that works in both urban environments with large disparity jumps and in smoothly varying terrain. At present, a constant step interpolation is used because it is the

most suitable method given the large depth jumps found in urban scenes.

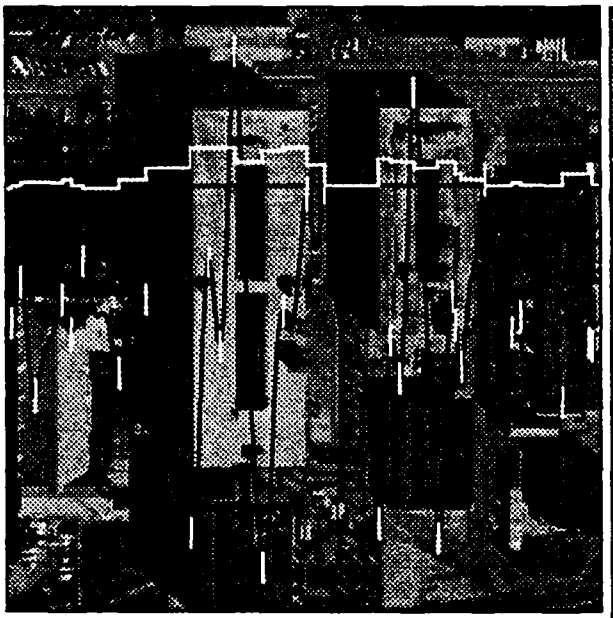

Figure 4-3: Gradient Wave Matched Points [Left]

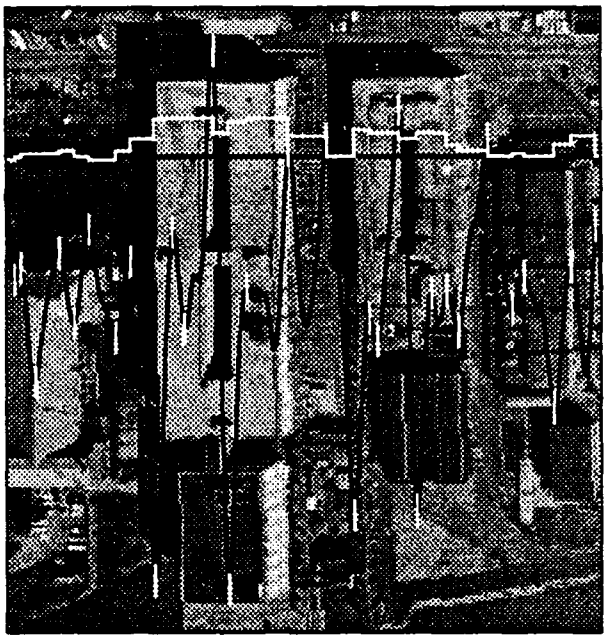

Figure 4-4: Gradient Wave Matched Points [Right]

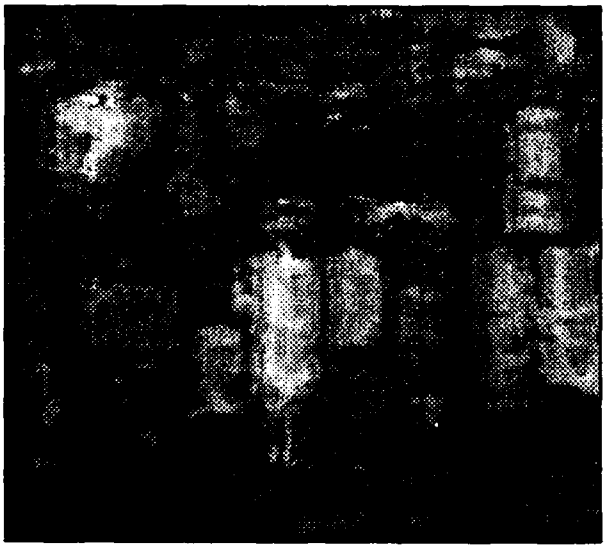

Figure 4-5: S1 disparity map

Figure 4-6: S2 disparity map

Figures 4-5 and 4-6 show the results of stereo matching using the scene registration shown in Figure 4-2. Disparity is scaled such that bright regions are closer to the observer and therefore have greater height. Dark regions are at or below the ground-plane. Figure 4-5 shows that, like most of the area-based algorithms, S1 performs well on textured, smoothed, and continuous regions. However the depth discontinuities are not well captured, and therefore the delineation of the high structures is not crisp. However, most of the terrain relief is present, particularly the slope of the land toward the water in the bottom portion of the scene. One advantage of S1 is that it is not overly reliant on perfectly registered stereo pairs

and therefore gives coherent results even without a perfect registration step. Figure 4-6 shows that S2 performs well on discontinuities even if we still have problems with the occlusions. However, in contrast to S1 it is very sensitive to the quality of the scene registration. To evaluate the precision of these disparity maps, we have manually generated a three-dimensional ground truth interpretation of the scene. Using anaglyph stereo visualization, we can get a subjective impression of the scene reconstruction using the ground truth data. We can also generate a quantitative accuracy assessment by performing a pixel-by-pixel comparison of the disparity map with the ground truth data.

4.3. Disparity Segmentation

Of course, the goal of stereo matching is not to generate only a disparity map. We need to extract hypotheses of the location and height of buildings and to develop a model of the underlying terrain. We have begun to experiment with two methods for disparity map interpretation. Once the disparity map is obtained, it is possible to extract regions with significant height and try to hypothesize them as buildings.

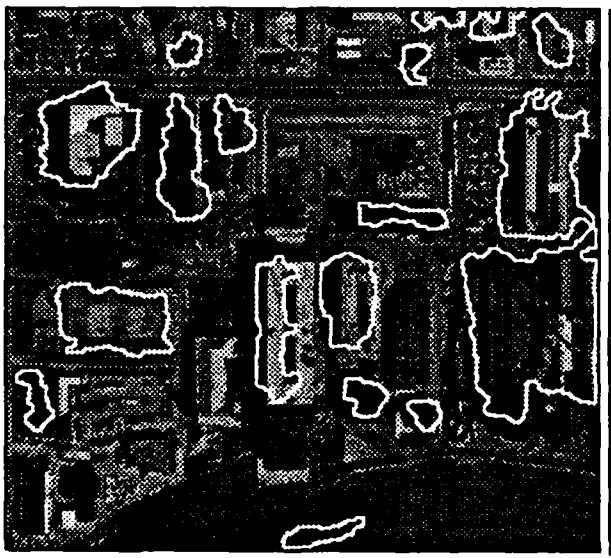

Figure 4-7: S1 disparity map segmentation

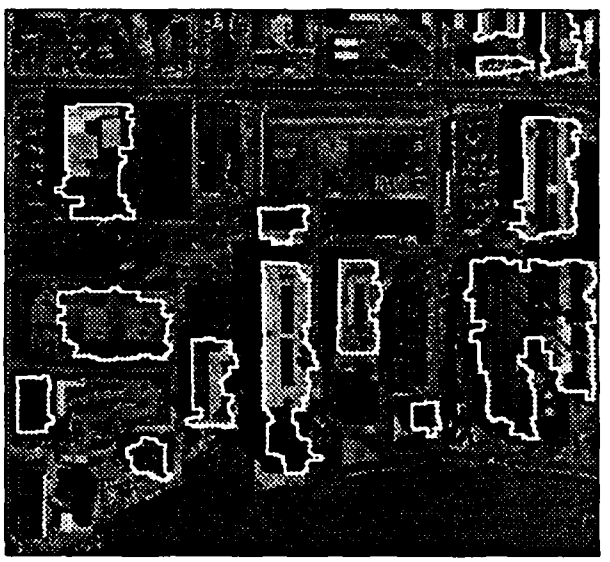

Figure 4-8: S2 disparity map segmentation

The first type of disparity segmentation is similiar to traditional connected component extraction used in image processing. The goal is to group pixels with similiar disparity values into sets of regions. There are two variations on this idea, a layer approach and a band grouping method. The layer method divides the disparity range into layers, where all disparity values above the specified layer are extracted and formed into connected component regions. The number of layers used depends on the disparity range. Band grouping selects overlapping ranges of disparity values and then performs connected component region extraction. In either case we arrive with a set of hypothesis regions that must be evaluated with respect to region size, shape, and elevation. Currently, our evaluation criteria has a simple but conservative definition of buildings based on region compactness and region height with respect to the local estimate of the terrain.

The second type of disparity segmentation extracts regions that are bounded by local

maxima in the disparity map. This is based on the assumption that buildings are not hidden in the ground, rather, they are higher than the surrounding terrain. Local extrema can be characterized by a positive jump in the disparity map followed by a negative jump. These depth jumps correspond to edges in the disparity maps and are linked to form regions. This method is more sensitive to errors in the initial disparity map than the layer or band variants, but provides better edge localization when buildings are found in rolling or uneven terrain.

Of course, the main difficulty with disparity map segmentation is due to the stereo matching errors produced by S1 and S2. We have experimented with various pre- and post-processing techniques to smooth the disparity maps. For S2, a vertical median filter is a logical choice since most mismatches manifest themselves as horizontal streaks across the disparity map. However, independent of intrinsic matching errors, it is difficult to detect low isolated buildings when they are embedded in rolling terrain or on high ground.

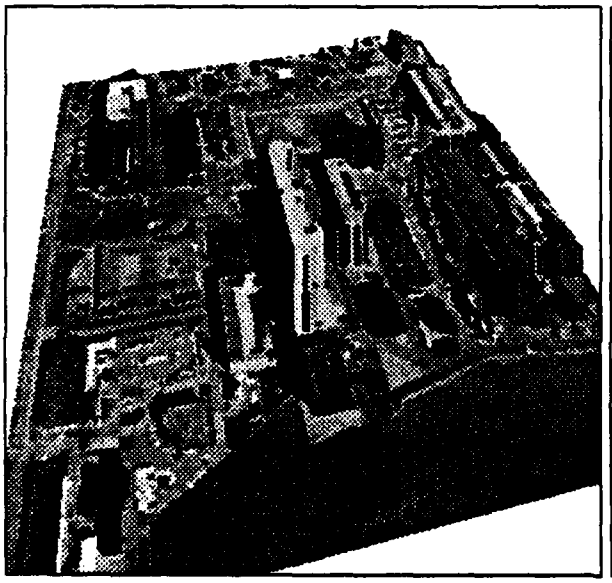

Figure 4-9: S1 results visualization

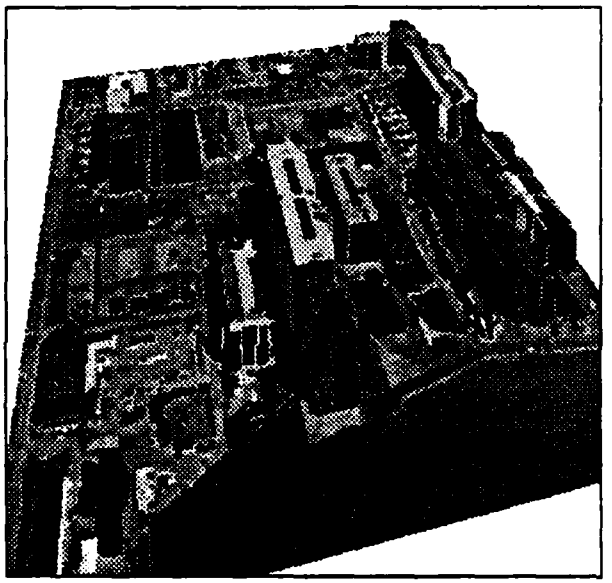

Figure 4-10: S2 results visualization

Figures 4-7 and 4-8 show the results of disparity map segmentation using the layer and band method of disparity segmentation. Note that in several cases what appear to be possible building structures (bright regions) in the disparity maps in Figures 4-5 and 4-6 are not aggregated into a disparity region. This is due to mismatches and occlusions on the boundaries that appear to make the regions more irregular and violates our compactness constraints. However, in the case of the S2 results there are no incorrect building regions and only one occurs in the S1 result. Figures 4-9 and 4-10 show the perspective view of the recovered three dimensional structure. This view is constructed in a manner similar to that in Figures 2-11 and 2-12 except that both the building location and height estimates are provided by stereo mensuration rather than approximated by shadow analysis and monocular building hypothesis.

4.4. Discussion

Fully automated stereo analysis in complex urban scenes is a difficult research problem. As we have discussed, there are three major areas in which improvements can be expected: scene registration, matching algorithms, and more sophisticated analysis of disparity results. We have not yet begun to exploit cues from monocular analysis nor have we looked at cooperative analysis of disparity maps. We believe that monocular analysis should allow us to make more intelligent decisions with respect to the disparity map based on evidence derived independently from each of the stereo pairs.

However, there are some sobering experiences. We have investigated the use of a bootstrap proceedure to initialize the S1 matcher using the disparity map generated by S2. We had high expectations that, given a better initial guess as to disparity, the area-based matching algorithm would provide both a refined height estimate and also improve the overall delineation. However after experimentation on several test scenes, we could not find any significant improvement over the original S1 matching results. Recovering low buildings in rolling terrain will require a significant improvement in reducing the errors generated by the matching process. Fundamentally, such buildings are lost in the noise due to mismatches. Since the terrain disparity tends to mask such structures, better spatial resolution may be required. However, higher spatial resolution will probably increase rather than diminish the number of mismatches due to increasing visibility of complex roof structures.

In the following section we discuss some research in integrating task domain knowledge with low-level and intermediate-level scene segmentations. The focus of this work is to utilize structural and spatial constraints concerning the relationships between objects to interprete a complex scene such as an airport or a suburban housing project.

5. Knowledge-Based Scene Analysis

Image interpretation requires substantial amounts of knowledge about the scene under consideration. Knowledge about the type of scene — airport, suburban housing development, urban city — aids in low-level and intermediate level image analysis, and will drive high-level interpretation by constraining search for plausible consistent scene models. Collecting and representing large knowledge bases requires specialized tools. In this section, we will describe an architecture for aerial image interpretation, a set of tools for creating and evaluating new knowledge for use in this architecture, and a technique, applicable to our system, for parallelizing production system applications.

5.1. SPAM Overview

SPAM [11, 13, 16] is a production system architecture for the interpretation of aerial imagery, with applications to automated cartography and digital mapping. It was initially implemented using the OPS5 production system language [5, 4] and has been reimplemented using ParaOps5 [9]. Depending on the task domain, SPAM has between 200 and 700 productions. As with many vision systems, SPAM attempts to interpret the 2-dimensional image of a 3-dimensional scene. A typical input image is shown in Figure 5-1. The goal of

the SPAM system is to interpret an image segmentation, composed of image regions, as a collection of real-world objects. For example, the output for the image in Figure 5-1 would be a model of the airport scene, describing where the runway, taxiways, terminal-building(s), etc., are located. SPAM uses four basic types of scene interpretation primitives: *regions*, *fragments*, *functional areas*, and *models*. SPAM performs scene interpretation by transforming image *regions* into scene *fragment* interpretations. It then aggregates these fragments into consistent and compatible collections called *functional areas*. Finally, it selects sets of functional areas to form *models* of the scene.

As shown in Figure 5-2, each interpretation phase is executed in the order given. SPAM drives from a local, low-level set of interpretations to a more global, high-level, scene interpretation. There is a set of hard-wired productions for each phase that control the order of rule executions, the forking of processes, and other domain-independent tasks. However, this "bottom-up" organization does not preclude interactions between phases. For example, prediction of a fragment interpretation in *functional-area (FA)* phase will automatically cause SPAM to reenter *local-consistency check (LCC)* phase for that fragment. Other forms of top-down activity include stereo verification to disambiguate conflicting hypotheses in *model-generation (MODEL)* phase and to perform linear alignment in *region-to-fragment (RTF)* phase.

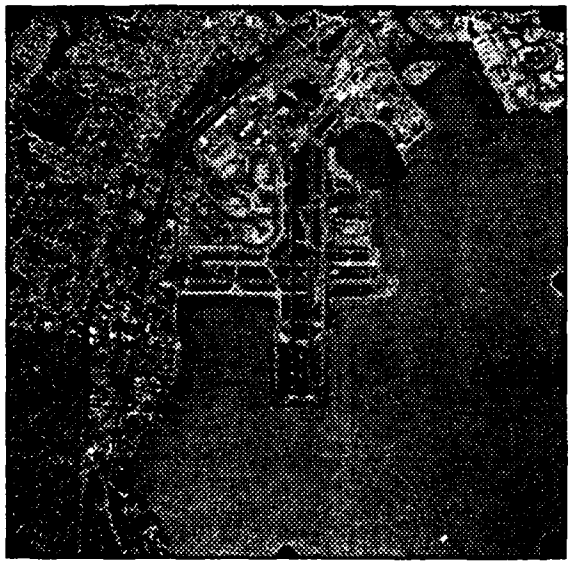

Figure 5-1: Aerial Image of San Francisco Airport

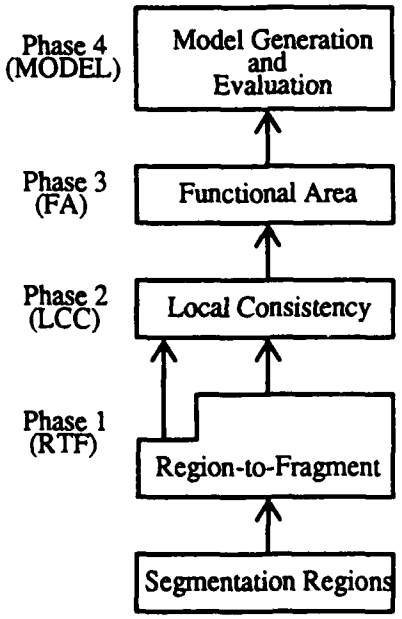

Figure 5-2: Interpretation phases in SPAM.

Another way to view the flow of processing in SPAM is that knowledge is used to check for consistency among hypotheses; contexts are created based on collections of consistent hypotheses and are then used to predict missing components. A collection of hypotheses must combine to create a context from which a prediction can be made. These contexts are refinements or spatial aggregations in the scene. For example, a collection of mutually consistent runways and taxiways might combine to generate a runway functional area. Rules that encode knowledge about runway functional areas may predict that certain sub-areas

within that functional area are good candidates for finding grassy areas or tarmac regions. However, an isolated runway or taxiway hypothesis cannot directly make these predictions. In SPAM the context determines the prediction. This serves to decrease the combinatorics of hypothesis generation and to allow the system to focus on those areas with strong support at each level of the interpretation.

5.2. Knowledge Acquisition and Compilation

SPAM is a scene-independent interpretation system. However, a large knowledge-base must be maintained for each scene type to be interpreted. Each knowledge-base consists of interdependent knowledge components to support the four phases of SPAM. Specialized tools are required to maintain and extend the knowledge-base for each phase while preserving consistency among them. In addition, a compiler is needed to convert the knowledge into a form usable by the interpretation system [16]. Maintaining the large body of knowledge needed to build the SPAM system requires that we have tools to add new knowledge, to refine and correct it, and ensure consistency between the components of the knowledge. We have developed a number of tools that aid the user in performing these tasks.

When adding new knowledge, new constraints can be inferred either from examples or from expert knowledge about the domain. For example, spatial constraints for an airport scene can be derived from the knowledge that engineers use to construct airports. Alternatively, the user can make measurements of a sample scene to derive the constraints. If ground-truth information exists, it is possible to summarize spatial information about each class of region, producing quantitative information about the region primitives. Figure 5-3 shows an example of the interactive measurement of house spacing in a suburban house scene, and the automatic generation of two-dimensional shape and orientation information for manually extracted houses.

Once knowledge has been entered in the form of constraints, it is not necessary to do a complete SPAM run to test it. The user may evaluate single constraints on sample data and then graphically view the results. Furthermore, if ground-truths are available, the results can be classified according to whether each region was evaluated correctly. The user may then adjust the constraint accordingly and reevaluate until the desired behavior has been achieved. Figure 5-4 shows the static evaluation of RTF rules for buildings and runways. A standard confusion matrix is used where the upper left quadrant depicts true positives, the lower right are true negatives, the lower left are false positives, and the upper right are false negatives. This display quickly allows a user to test the utility of an RTF constraint over several ground truth datasets.

With the large number of possible constraints, and the complex interactions between the four phases of SPAM, it is important that consistency be maintained over the entire knowledge-base. As the user enters or changes knowledge, the system checks for contradictory, extraneous, or missing information. A course of action may be suggested to the user depending on the error. Although the checks do not guarantee the correctness of the

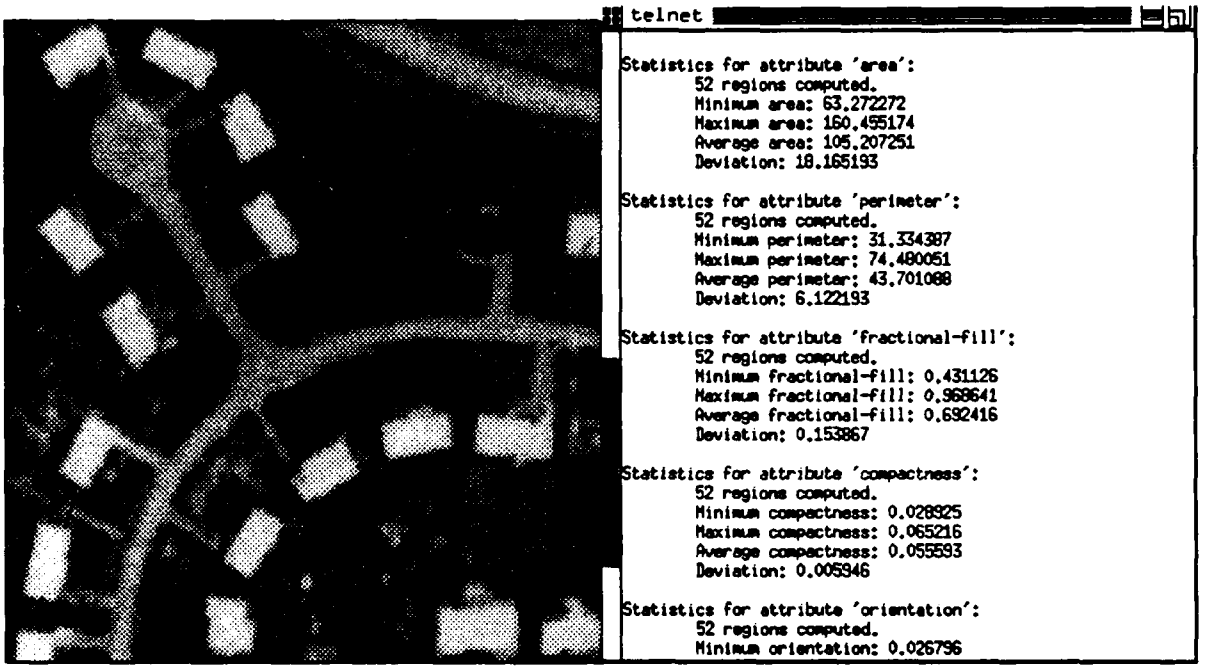

Figure 5-3: Deriving spatial constraints by hand (left), & from ground-truth analysis

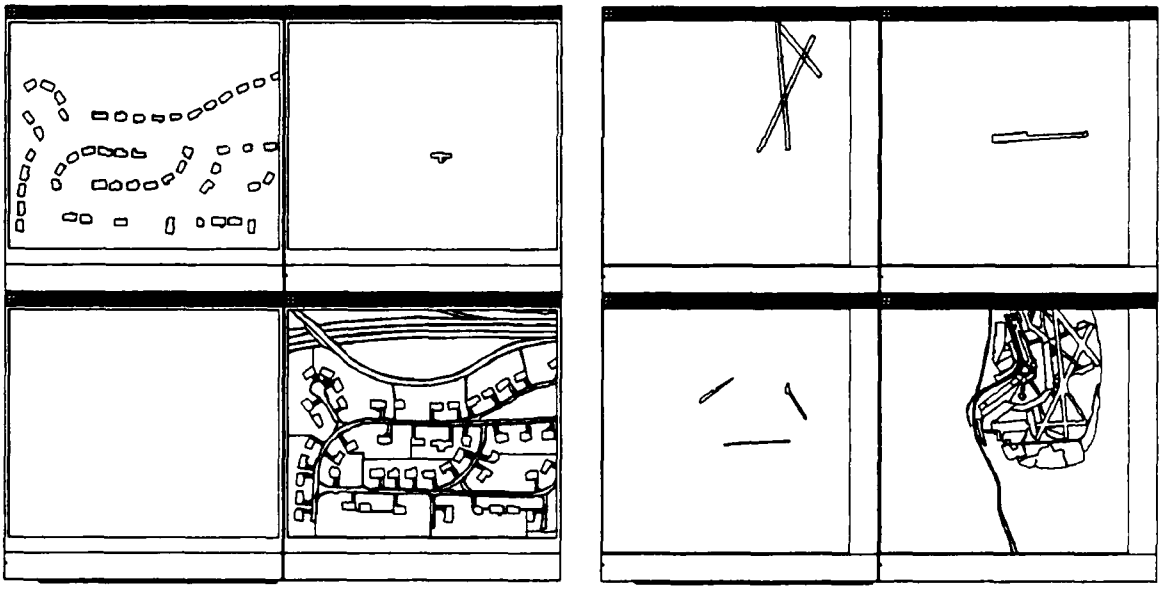

Figure 5-4: Results from evaluation of sample suburban and airport constraints

system, they do make it possible for the user to build a large knowledge-base free of inconsistencies that would prevent the knowledge from behaving as intended.

Before scene analysis can be performed, the knowledge base must be converted into a form that can be utilized by our interpretation system. A set of productions is compiled from the knowledge-base, and a set of generic control productions is added to coordinate the system's run-time behavior. Representing the knowledge-base independently of the run-time system simplifies its maintainance and is less error-prone than requiring a human to hand-code the knowledge.

5.3. Parallelism in SPAM

The production system programming model used by SPAM allows knowledge to be added easily. However, large production systems like SPAM continue to suffer from extremely slow execution times, which limits their utility in practical applications, as well as in research settings. Most efforts at speeding up these systems have focused on match or knowledge-search parallelism in production systems. Although good speed-ups have been achieved in this process, the total speed-up available from this source is not sufficient to alleviate the problem of slow execution in large-scale production system implementations. Such large-scale tasks can be expected to increase as researchers develop increasingly more competent rule-based systems.

Another area of our research, done in conjunction with the Production System Machine group at Carnegie Mellon, has focused on task-level parallelism (TLP) [6], which is obtained by a high-level decomposition of the production system. We will describe some timing results from the original, Lisp-based SPAM system, then show our problem decomposition, and finally show the resultant performance improvements obtained using task-level parallelism running on a 16 processor Encore Multimax.

5.3.1. Diagnostic Timing Results

Table 5-1 gives statistics for the run-time and number of production firings for each interpretation phase in SPAM for *San Francisco International Airport (SF)* . These statistics are representative of those obtained from other data sets. It is interesting to note that LCC and FA phases account for most of the overall time in a complete run. Further, within these phases, much of the rule evaluation is performed outside of the OPS5 production system using external processes. For example, FA spends much of its time doing evaluation outside of OPS5. RTF, on the other hand, spends most of its time within the traditional OPS5 evaluation model and consumes less time than FA, even though it executes a comparable number of productions. It is also clear from this table that the application of spatial constraints in LCC makes it by far the most expensive phase in terms of amount of time spent, number of productions, as well as number of production firings.

SPAM Phase	RTF	LCC	FA	MODEL	Total
Total CPU Time (hours)	1.5	144.5	7.3	0.71	154.01
Total Productions Fired	11274	185950	10447	3085	210756
Effective Productions/Second	2.08	0.357	0.397	1.20	0.380
Total Hypotheses	466	N/A	44	1	N/A

Table 5-1: San Francisco Airport (log #63)

During the LCC phase, knowledge of the structure or layout of the task domain (i.e. airports or suburban housing developments) is used to provide spatial constraints for evaluating consistency among fragment hypotheses. For example, *runways intersect taxiways* and *terminal buildings are adjacent to parking apron* are examples of the kinds of constraints that are applied to the airport scene segmentation. It is important to assemble a large collection of

such consistency knowledge because the results of these tests are used to assemble fragment hypotheses found to be mutually consistent as contexts for further interpretation within the functional area phase.

Just from the raw statistics, it seems clear that the LCC phase should be our candidate for applying parallelism. Another rationale for this approach is the observation that this phase has the largest potential for growth. If a single new scene primitive is added within the RTF phase, many constraints may be added in the LCC phase in order to describe the spatial relationships (and constraints) between each of the other primitives. For these reasons, we believe that as new knowledge is added to the existing SPAM system, the proportion of time can only increase in the LCC phase.

5.3.2. Decomposing the LCC Phase

As a result of this preliminary analysis we decided to focus our initial efforts on the parallel implementation of the LCC phase[1]. The LCC phase applies geometric knowledge (constraints) from the selected domain to the set of interpretations made from the dataset. This application of geometric knowledge can be logically decomposed into several levels, where the tasks within each level are independent and can be performed in parallel. This is illustrated in Figure 5-5.

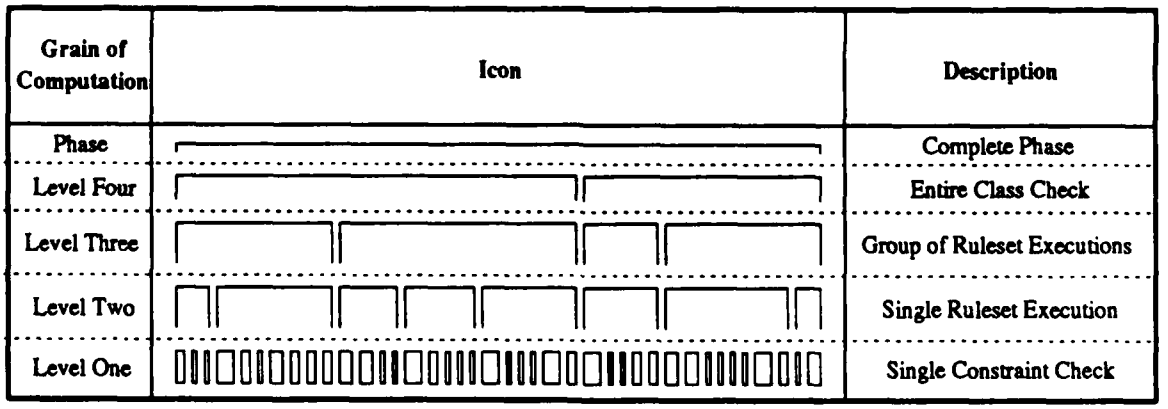

Figure 5-5: Levels of processing in SPAM LCC.

In order to choose the right level of decomposition at which to parallelize the SPAM LCC phase, we instrumented the SPAM system to obtain measurements at each level for the number of tasks and their run-time average, standard deviation, and coefficient of variance. The results of these measurements for each of the San Francisco airport dataset are presented in Table 5-2.

Using information from Table 5-2 the appropriate level of granularity can now be chosen. For Level 4, the task to processor ratio is smaller than one, so we immediately rejected

[1]Since the analysis is performed using the original, expensive Lisp-based SPAM system, we have extracted a representative subset of the airport dataset to drive the analysis.

Level	Average time per task (sec)	Standard deviation (sec)	Coefficient of variance	Number of tasks
Level 4	875.27	525.92	0.601	9
Level 3	65.65	29.51	0.449	120
Level 2	20.90	8.48	0.406	377
Level 1	0.489	0.0782	0.159	16104

Table 5-2: Average, standard deviation and coefficient of variance for SF.

pursuing parallelism at this level. Levels 3 and 2 are very similar to each other in that they have enough tasks, their variances are not large, and the task granularities are much larger than the expected task management and communication overheads. Both levels, therefore, seemed to us to be worthwhile candidates. Level 3 seemed somewhat more desirable as less effort appeared to be required of us to achieve amounts of parallelism similar to that available in Level 2.

Level 1 was rejected for several reasons. First and most importantly, the additional effort involved in decomposing the system at the granularity of Level 1 would not allow us to achieve any more parallelism than at Level 2 or 3 because of the limitation on the number of processors. Second, the task granularity is much smaller and thus closer to the overheads for task management and communication than any of the other levels. Finally, the task to processor ratio is on the order of 1000. This can have a detrimental effect due to the initialization overhead. Our conclusion, then, was to exploit parallelism at the granularity of Levels 2 or 3.

5.3.3. Results from Using Task-Level Parallelism

Results for parallelizing Levels 2 and 3 are shown in Figure 5-6. The speedups are computed against a baseline version, which represents an optimized uniprocessor implementation of the SPAM LCC phase. It is interesting to note that this uniprocessor baseline version provides approximately a 10 to 20 fold speedup over the original Lisp-based implementation for the LCC phase.

The results of applying task-level parallelism are shown in Figure 5-6. Curves are shown for the San Francisco dataset, as well as for two other airport datasets. The speed-up curves show near linear speed-ups for both levels of decomposition. The speed-ups within a level are almost the same among the three airport datasets. The maximum speed-up achieved using 14 processors is 11.90 fold in Level 3 and is 12.58 fold in Level 2.

For match parallelism, the theoretical maximum speed-up that can be obtained is limited according to the percentage of total execution time spent in match. As SPAM spends less than 50% of its time match, speed-ups due to match parallelism are limited to 2 or less. This is exactly what is observed.

We believe that the potential for additional speed-ups in SPAM from task-level parallelism is

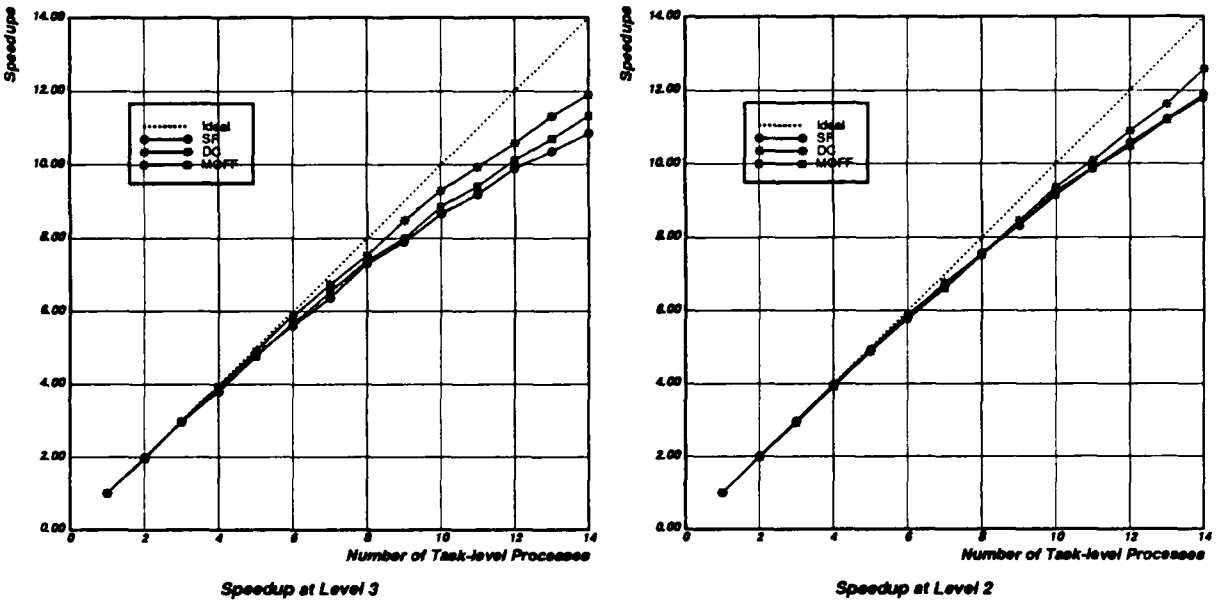

Figure 5-6: Speed-ups varying the number of task-level processes.

quite high; an expectation of 50 to 100 fold does not seem unreasonable, because:

- The tasks within any of the LCC decompositions are independent of one another;
- Several hundred tasks are available in Level 2;
- The task queue management overheads measured for Level 2 and Level 3 are very low, especially with respect to the task granularity, and thus are not a factor.

Although our scheme of parallelization has been presented in the context of a non-match-intensive system, the scheme is applicable to match-intensive systems as well. In match-intensive systems, match parallelism will make a substantial contribution to the speed-ups.

5.4. Discussion

SPAM provides a context for pursuing research in several areas in automated scene analysis. While our previous discussion of building and road extraction techniques clearly provides the scene domain primitives for high-level analysis, the automation of the analysis tasks requires the integration of domain and task knowledge. Our research in knowledge-based scene analysis attempts to provide a comprehensive analysis of the issues involved in building and using knowledge-based systems. SPAM is an architecture for using knowledge to interpret aerial images, but an interpretation architecture alone is insufficient to solve the problem. We require a means of storing, adding, and evaluating that knowledge, hence our focus on the issues of knowledge acquisition. We are also examining the automatic derivation of knowledge from examples and from abstract domain knowledge, such as a functional model of the scene. Such analysis can be used either directly to generate constraints, or indirectly as an interactive aid to the user as they modify or refine the knowledge-base. Finally, our work in utilizing task-level parallelism will not only enable SPAM to perform its work more efficiently, but will also provide the opportunity to bring more knowledge to bear during scene interpretation.

6. Conclusions and Future Work

Computer vision and image understanding address difficult problems in a variety of task domains. In many cases, such as in certain industrial robotics applications, one can choose to engineer the problem domain in order to make automated sensing and manipulation tractable using current technology. In cartography, however, one is presented with two-dimensional images of the unconstrained three-dimensional world. We can not paint red squares on the corners of buildings in order to make roof detection more tractable for our computer vision techniques. Success and failure in these tasks are easily determined since we have a well understood basis for human performance in the cartographic community.

Although it is clear that humans bring a great deal of knowledge and context to bear when attempting to understand the structural and spatial relationships inherent in a scene, we are still a long way off from having such a level of expertise embodied in computer interpretation systems. The variety and complexity of man-made structures and natural terrain make the automated extraction and analysis one of the most difficult challenges for computer vision research.

6.1. Third Year Research Plan

During the third year of this research contract we plan to continue with incremental improvements on all of the research sub-areas including road network analysis, building extraction, shadow analysis, stereo analysis, and knowledge-based scene interpretation. There are many small incremental improvements for which we understand the issues well enough to proceed. However, there are several areas in which we plan to concentrate our efforts to bring forth major new research results. These areas are:

1. The interpretation and analysis of stereo disparity maps using a fine grain monocular scene segmentation to guide interpolation. We believe that the application of region-based analysis to the original intensity image can be used to produce an over segmented set of planar surfaces that correspond to building and ground patches. These patches do not form a semantically meaningful scene description, but they are likely to share similar disparity, or height estimates. Analysis of the disparity image *guided* by the intensity image appears to be a promising technique to reject mismatches and to generate a refined disparity map that lends itself to further interpretation. The proposed technique appears to be superior to many interpolation based methods because it explicitly takes into account the nature of surface patches with similar albedo.

2. The improvement of the S2 feature-based algorithm, particularly with respect to waveform approximation and the use of inter-scanline consistency to detect and correct mismatches. As we perform analysis on a variety of stereo pairs (currently 10 datasets) with various disparity ranges including significant terrain relief we are better understanding issues in how to approximate the intensity waveform. This work, and related work in the automatic estimate of scene disparity using feature matching should give us significant improvements in the basic disparity map.

3. We will continue to improve our ability to detect and delineate shadow regions in aerial imagery. Particularly building that are oriented perpendicular to the sun direction do not cast shadows with the characteristic L shape that we have

used as a basis of shadow analysis. We believe that we can relax this requirement to handle these cases. Another issue is to extend the work on grouping using common shadow/building boundaries to cycle these building groups back into the monocular scene analysis component, BABE.

7. Publications, Reports, Presentations

During the second year of our research contract in Built-Up Area Feature Extraction we have continued to publish our results in refereed journals and conferences, and to present progress reports at various meetings. This section details the most significant publications and presentations supported under this contract.

7.1. Publications

- W. Harvey, D. Kalp, M. Tambe, D. McKeown, A. Newell, "Measuring the Effectiveness of Task-Level Parallelism for High-Level Vision" in Proceedings of *DARPA Image Understanding Workshop*, Palo Alto, California, May 23-26, 1989. Morgan Kaufmann Publishers., pp. 916-933.
- D. M. McKeown, Jr., Harvey, W.A., and Wixson, L. "Automating Knowledge Acquisition For Aerial Image Interpretation" Computer Vision, Graphics and Image Processing Volume 46, Number 1, April, 1989. pp 37-81.
- F. P. Perlant, and D. M. McKeown, "Scene Registration in Aerial Image Analysis" in Proceedings of *SPIE Conference on Reconnaissance, Astronomy, Remote Sensing and Photogrammetry* Los Angeles, Calif, January 19-20, 1989., Volume 1070, pp. 88-99.
- R. B. Irvin, and D. M. McKeown, "Methods for exploiting the relationship between buildings and their shadows in aerial imagery" in Proceedings of *SPIE Conference on Image Understanding and the Man-Machine Interface II* Los Angeles, Calif, January 17-18, 1989., Volume 1076, pp. 156-164.

7.2. Invited Presentations

- "Automated Feature Extraction in Urban Areas" Project 2851 Mission Rehearsal Special Interest Group, Defense Mapping Agency Aerospace Center, St. Louis, MO. September 28, 1989.
- Participant/Panel Leader: Specialist Meeting on Large Spatial Databases, NSF-National Center for Geographic Information and Analysis, Santa Barbara, Cal., July 19-22, 1989.
- "Artificial Intelligence in the Analysis of Aerial Imagery", IEEE Computer Society Workshop on Artificial Intelligence for Computer Vision, San Diego, Cal., June 5, 1989.

7.3. Personnel

The following members of the Digital Mapping Laboratory, School of Computer Science, Carnegie Mellon University were fully or partially supported during the second year of this research contract.

- David M. McKeown, Jr.
  Research Computer Scientist
- Aviad Zlotnick
  Post Doctoral Research Associate
- Wilson A. Harvey
  Senior Research Programmer
- Frederic P. Perlant
  Visiting Scientist
- Undergraduate Research Assistants
  Emily Burke, Bruce Irvin, Jeffrey Shufelt, Lambert Wixson

8. Bibliography

[1] Aviad, Z. and P. D. Carnine.
Road Finding for Road Network Extraction.
In *Proceedings: Computer Vision and Pattern Recognition*, pages 814-819. Ann Arbor, Michigan, June, 1988.

[2] Aviad, Z.
Locating Corners in Noisy Curves by Delineating Imperfect Sequences.
Technical Report CMU-CS-88-199, Carnegie-Mellon University, December, 1988.

[3] Barnard, S. T. and Fischler, M. A.
Computational stereo.
Computing Surveys 14(4):553-572, December, 1982.

[4] Brownston, L., Farrell, R., Kant, E., and Martin, N.
Programming Expert Systems in OPS5: An introduction to rule-based programming.
Addison-Wesley, Reading, Massachusetts, 1985.

[5] Forgy, C. L.
OPS5 User's Manual.
Technical Report CMU-CS-81-135, Computer Science Department, Carnegie Mellon University, July, 1981.

[6] Wilson Harvey, Dirk Kalp, Milind Tambe, David McKeown, Allen Newell.
Measuring the Effectiveness of Task-Level Parallelism for High-Level Vision.
In *Proceedings of the DARPA Image Understanding Workshop*, pages 916-933. Morgan Kaufmann, May, 1989.

[7] Huertas, A. and Nevatia, R.
Detecting Buildings in Aerial Images.
Computer Vision, Graphics, and Image Processing 41:131-152, April, 1988.

[8] R. B. Irvin and D. M. McKeown.
Methods for exploiting the relationship between buildings and their shadows in aerial imagery.
IEEE Transactions on Systems, Man and Cybernetics 19(6):1564-1575, November, 1989.

[9] Kalp, D., Tambe, M., Gupta, A., Forgy, C., Newell, A., Acharya, A., Milnes, B., and Swedlow, K.
Parallel OPS5 User's Manual.
Technical Report CMU-CS-88-187, Computer Science Department, Carnegie Mellon University, November, 1988.

[10] B. D. Lucas.
Generalized Image Matching By The Method of Differences.
PhD thesis, Carnegie Mellon University, July, 1984.

[11] McKeown, D.M., Harvey, W.A. and McDermott, J.
Rule Based Interpretation of Aerial Imagery.
IEEE Transactions on Pattern Analysis and Machine Intelligence PAMI-7(5):570-585, September, 1985.

[12] McKeown, D.M.,
Digital Cartography and Photo Interpretation from a Database Viewpoint.
In Gargarin, G. and Golembe, E. (editor), *New Applications of Databases*, pages 19-42. Academic Press, New York, N. Y., 1984.

[13] McKeown, D.M., McVay, C.A., and Lucas, B. D.
Stereo Verification In Aerial Image Analysis.
Optical Engineering 25(3):333-346, March, 1986.

[14] McKeown, D.M.
The Role of Artificial Intelligence in the Integration of Remotely Sensed Data with Geographic Information Systems.
IEEE Transactions on Geoscience and Remote Sensing GE-25(3):330-348, May, 1987.

[15] McKeown, D.M. and Denlinger, J. L.
Cooperative Methods for Road Tracking in Aerial Imagery.
In *Proceedings IEEE Computer Vision and Pattern Recognition Conference*, pages 662-672. June, 1988.

[16] McKeown, D.M., Harvey, W.A., Wixson, L.
Automating Knowledge Acquisition For Aerial Image Interpretation.
Computer Vision, Graphics and Image Processing 46(1):37-81, April, 1989.

[17] Perlant, F. P., McKeown, D.M.
Scene Registration in Aerial Image Analysis.
In *SPIE Proceedings on Reconnaissance, Astronomy, Remote Sensing, and Photogrammetry*, pages 88-99. January, 1989.
Also available as Technical Report CMU-CS-89-127.