CMU-CS-83-136

# MAPS: The Organization of a Spatial Database System Using Imagery, Terrain, and Map Data

David M. McKeown, Jr.

July 17, 1983

## Abstract

This paper presents the system description and organization of MAPS, the Map Assisted Photo interpretation System. MAPS is a large integrated database system containing high resolution aerial photographs, digitized maps and other cartographic products, combined with detailed 3D descriptions of man-made and natural features in the Washington D.C. area. Applications of the MAPS system in the areas of map-guided image segmentation, rule-based systems for image interpretation, and 3D scene generation are discussed. A classification of image database systems into three models is also presented. These models are the Image Database (ID) model, the Map Picture Database (MPD) model and the Image/Map Database (IMD) model.[1]

This research was sponsored by the Defense Advanced Research Projects Agency (DOD), ARPA Order No. 3597, monitored by the Air Force Avionics Laboratory Under Contract F33615-81-K-1539. The views and conclusions contained in this document are those of the author and should not be interpreted as representing the official policies, either expressed or implied, of the Defense Advanced Research Projects Agency or the US Government.

---

[1] This paper was presented at the DARPA IUS Workshop, held in conjunction with the IEEE Computer Vision and Pattern Recognition Conference (CVPR) on June 23, 1983, in Crystal City, Virginia.

# MAPS:
# The Organization of a Spatial Database System Using Imagery, Terrain, and Map Data

David M. McKeown, Jr.
Carnegie-Mellon University
Pittsburgh, PA 15213
April 1983

## Abstract

This paper presents the system description and organization of MAPS, the Map Assisted Photo interpretation System. MAPS is a large integrated database system containing high resolution aerial photographs, digitized maps and other cartographic products, combined with detailed 3D descriptions of man-made and natural features in the Washington D. C. area. A classification of image database systems into three models is also presented. These models are the Image Database (ID) model, the Map Picture Database (MPD) model and the Image/Map Database (IMD) model.*

## 1. Introduction

This paper presents the system description and organization of MAPS, the Map Assisted Photo interpretation System. MAPS is a large integrated database system containing high resolution aerial photographs, digitized maps and other cartographic products, combined with detailed 3D descriptions of man-made and natural features in the Washington D. C. area.

This paper discusses three major topics. First, a classification of different models of database systems for cartographic applications is presented together with a discussion of their inherent strengths and limitations. These models are the Image Database (ID) model, the Map Picture Database (MPD) model and the Image/Map Database (IMD) model. Second, we argue for the utility of the Image/Map Database model, discuss tasks and present a general description of the model. This model describes components, facilities and techniques that should be present in such a system, and a range of tasks that can be supported by the model. Finally, we describe the MAPS system in terms of our (IMD) model, and discuss three applications which utilize and integrate image, terrain, and map data in a powerful manner. We also discuss what we have learned during the implementation of the MAPS system, some ideas on the proper interfaces between components, where modularity should be achieved, and point to future work.

## 2. Background

Our early motivation for investigating image databases was as a component of a complete image understanding system. We had only a vague idea of what capabilities it should have, but we thought that it should represent "idealized segmentations" of an image, where the labeling of the segments was in fact the "scene interpretation". It should relate, or compare machine generated segmentations to this model, and provide the user with a qualitative and quantitative performance measure of the machine segmentation. We attempted this with the MIDAS system[1, 2] using the segmentation results for a set of Pittsburgh city scenes generated by the ARGOS[3, 4] system. The results of the performance analysis of the scene segmentation were less than encouraging. While we could give quantitative analysis of the segmentation and labeling by the ARGOS system, the qualitative results were couched in the original (subjective) hand segmentations. It was difficult to qualitatively distinguish between alternative machine segmentations, since the relative importance (or cost function) of missing or mislabeled regions or broken boundaries for different regions was not represented in the segmentation. How to perform such an evaluation is still an open research problem. Also, although we had a database of 18 high resolution color images of Pittsburgh, we had no general mechanism to relate one to another, except through analysis of the hand segmentations and the names given to buildings, roads, rivers, and other features in the scene. However, in the process of implementing and using MIDAS we did learn a great deal about image database organization and symbolic representation of scene descriptions.

We decided to look at map-guided image interpretation and began to assemble an aerial photograph database of the Washington, D. C. area. Using this imagery, we felt, we could quickly generate a map database that would allow us to explore image analysis of complex aerial photographs using a simple map database that constrained where to look, and what to look for. This idea of map-guided segmentation was not new. The HAWKEYE system[5] and succeeding "road expert"[6, 7] were based on similar ideas, and use of world knowledge had been a well accepted paradigm in image interpretation. However, we wanted to focus on more general capabilities, to represent large scale spatial organizations normally encountered in complex urban scenes. The generation of the map database turned out to be a much harder

---

[*] This research was sponsored by the Defense Advanced Research Projects Agency (DOD), ARPA Order No. 3597, monitored by the Air Force Avionics Laboratory Under Contract F33615-81-K-1539. The views and conclusions contained in this document are those of the author and should not be interpreted as representing the official policies, either expressed or implied, of the Defense Advanced Research Projects Agency or the US Government.

problem than we initially estimated, and it quickly became the focus of our research. In retrospect, I believe, it was exactly the right problem to work on, and although there is still much to do in the area of image/map databases, we now have the right tools and understanding to begin to tackle the original problem. This work has direct application in three areas:

- photo-interpretation: representation of world knowledge for image understanding.
- situation assessment: a spatial expert for decision support systems.
- cartography: toward digital map generation and use.

## 3. Classification of Databases

There has been, over the last ten years, a perceived need for organizing and structuring image and map data for cartographic applications. It has been difficult to compare various capabilities and limitations of systems because there were few common denominators by which systems could be compared. Systems reported in the literature could loosely be categorized either as research vehicles, or production-oriented systems for particular well defined subtasks of the general cartographic problem[8, 9, 10]. Research vehicles generally had a high degree of organizational complexity tested on very small scale databases. Systems used in production environments tended toward simple models running very large scale databases. Further, while the tasks being performed involved the analysis of aerial or satellite data, it is often unclear whether the image data was an integral part of the resulting database, or simply used for data acquisition. One example is the development of digital filing systems that store facts about a large number of images without storing the actual image data. The best example of such a system is the EROS Data Center database maintained by the U.S. Dept. of the Interior. This database has approximately $2x10^6$ frames of Landsat imagery and $5x10^6$ frames of aircraft (aerial mapping) photography. Users may specify an area of interest by geodetic point or rectangular area and sub-select those frames based on time of year, cloud cover, type of sensor** and a a scene quality rating. However, the actual frames of data are stored on high density magnetic tape. Similar situations exist in map producing organizations such as the United States Geological Survey (USGS) and the Defense Mapping Acency DMA.

One notable exception is described in Kondo et.al.[11] where an image database using Landsat imagery was integrated with map descriptions for geographic, natural, and cultural features. Features can be displayed superimposed on the image data, and imagery could be indexed by geodetic location or by feature name. There are limitations such as: the image-to-map correspondence was based on a fixed decomposition of landsat data into a latitude/longitude grid at a map scale of 1:50000; the spatial relationships between features were entered manually; and the overall complexity of the image and map database was small. Nevertheless, this represents an ambitious new direction for the development of land-use systems using Landsat imagery.

In this discussion of database systems for cartographic and situation assessment applications, we are assuming that the following minimal capabilities hold: (1) on-line display of digital imagery and map data, and (2) ability to query interactively about attributes of the imagery and map. The following is our classification of the capabilities of three models which we can use to compare various existing systems or approaches. These models are the Image Database (ID) model, the Map Picture Database (MPD) model and the Image/Map Database (IMD) model.

### 3.1. Image Databases

The Image Database model (ID) is the simpliest and most common database model. It is organized to relate attributes about the sensed image such as sensor-type, acquisition, cloud cover, or geodetic coverage***. These databases generally do not represent the content of the scene, but rather attributes of the scene. When the semantics of the scene are present, the location of cartographic features are represented in the image (pixel) coordinate system. This poses obvious limitations to the application of relevant knowledge from other images or from external sources, since there is no general mechanism to relate map feature position between images that overlap in coverage or to an external map. Although the features represented may appear to be map-oriented, is is difficult to compute general geometric properties using the image raster as the coordinate system.

Although relational database techniques have been applied to the ID model, we feel these techniques are not appropriate to spatial database organizations for several reasons. First, using the basic *<attribute, value>* tuple to represent vector lists of map coordinate data requires that all of the primary key attributes be duplicated in each relation, since there is no mechanism for allowing multiple valued (sets, lists, order pairs) as a primitive attribute in a relation. Further, the relational database operations such as union, intersection, join, project, are not good primitives for implementation of inherently geometric operations such as containment, adjacency, intersection and closest point. Operations such as feature intersection are reduced to searching for line segments which share the same pixel position. Finally, in any large system, a logical partitioning of the database must be performed in order to avoid extensive and often unnecessary search when performing spatial operations. Partitioning is difficult to achieve in relational systems since the relational model restricts itself to homogeneous (only one record type) sequential sets. Previous work advocating such organizations did not address the issues of system scale, and focused more on issues of query languages using relational models for geographic databases than the actual construction of complex systems[12, 13, 14]. When measured by the number of images, image-based features, and by the complexity of the relationships represented, these systems were quite simplistic.

---

** for aerial mapping photography

*** using flight annotation such as the center point and corner points not using general image-to-map correspondence

### 3.2. Map Picture Databases

The Map Picture Database model (MPD) describes databases that are generated by digitizing cartographic products, such as pre-existing maps and charts. These databases are attractive in environments where paper maps have played a large role in planning and analysis. There are, however, some major limitations to spatial systems based on digitized cartographic products. First, in the original map production, spatial ambiguity has been rectified by the cartographer in a manner that is not often reversable. The cartographic process involves simplication (generalization), classification (abstraction), and symbolization of real-world ambiguity. Constraints imposed by the scale of the map often determine which world features can be depicted despite the desirability of portraying a complete spatial representation. Therefore, map icon and symbology placement may not be as accurate as the original source material. Since the deduction of the actual spatial arrangement of objects from an iconic representation is an open problem, MPD's represent chaos masquerading as rationalized order. The key issue is that MPD's are pictures of a map (however detailed) rather than the underlying map structure and spatial organization. Although the graphics display of MPD appears to convey a great deal of semantic information, that impression is a result of the human observer, not a reflection of an underlying map representation.

When a map is digitized into a map picture, another subtle simplification occurs. The digitization process results in a map image on a rectangular grid whose size is generally limited either by custom or as an artifact of the digitization process. Common limitations are scanner resolution, maximum size of image raster, and the physical size of source map. One popular representation is to subdivide regions of the map picture into a regular decomposition such as quad-tree[15, 16], or k-d tree[17]. The implementation of this representation is greatly simplified in MPD models since one no longer has to contend with positional ambiguity of map features because of the cartographic process outlined above, and the discrete nature of the digitization process.

One common use for the MPD model is in geographic information systems for land use and urban planning. In these systems, aggregrate values such as population of an area and crop yield of an area are computed. The scale of the original map becomes the limiting factor for accuracy in information computation. However, the grain of computation is usually large enough that these inaccuracies are not a practical problem. Incremental update of the database due to new residential and industrial areas and the concomitant loss of rural areas is a difficult problem since database update requires careful map editing tools not usually associated with these MPD systems.

A recent trend has been to take existing MPD databases and add a map feature database component, usually relational to describe attributes of various features. We believe that augmenting traditional MPD databases with semantic information has merit in those enviroments where analysis is being performed by humans, since information synthesis is not a requirement of the database system. However, once such a system is in place, there is a tendency to attempt to automate analysis functions requiring spatial interpretation, and the generation method of the MPD model has several drawbacks for use in photo-interpretation, situation assessment, and cartography. The chief problems are the method of generation as outlined above, the lack of semantic information about map features, and the requirement that a map exist at the appropriate level of detail for the area under consideration. The IMD model discussed in the following section addresses these issues.

### 3.3. Image/Map Databases

The Image/Map Database model (IMD) relates map features to image database through camera models. It therefore has the capability to describe relationships between features acquired from different images through the map database. This capability is in contrast to the image database model where the feature descriptions can only be related if the descriptions come from the same image.

Since the map database is built directly from aerial imagery in the IMD model, the resolution / accuracy issue is a function of the ground resolution of the imagery, the intrinsic position measurement error due to camera model, ground control, etc. rather than an artifact of the map depiction scale as in the MPD model. A greater variety of feature descriptions is possible since they are not restricted to those that can be portrayed in a cartographic product. Further, the complexity of a particular feature description is independent of any particular task requirement and can represent a rich set of attributes, semantic interpretations, and knowledge from diverse sources. This flexibility is a key element for map data representation as we look toward spatial database systems with applications in cartographic production, expert photo-interpretation, and situation assessment.

However, just as the cartographer must resolve ambiguity, so the spatial database must be able to represent inconsistency in a consistent manner. For example, errors in correspondence between images and the geodetic model cause the same point on the earth to be given a different geodetic position, ie. when viewed from different images the same geodetic point produces a different world position. If this point is on a common boundary between two features, say a political boundary, there should be ambiguity as to which region the point is in. By the same token, if two large residental areas are found to intersect because of positional uncertainty, and the result of the intersection is several small polygonal areas, the IMD model should be able to rectify this ambiguity. This rectification might take the form of a symbolic relationship that indicates that the residential area share a common boundary, while maintaining the ability to represent the original errorful signal data. Since the original data is maintained in the database, the symbolic relationships do not have to be static. For example, these relationships can be dependant on attributes similiar to those used by cartographers when they perform simplification and generalization. The link from the symbolic interpretation back to the original source data is not possible in MPD systems.

### 3.3.1. Spatial Knowledge

The IMD model gives us the tools to construct our map database from "first principles" and tie together partial spatial knowledge at different levels of detail. This is possible because individual map features may be specified directly from source imagery. This capability is precluded by the derivative nature of the MPD model. That is, it is difficult to assimilate new and possibly errorful knowledge because of the mismatch between the new errorful data and the cartographic rectification of ambiguous data.

The representation of a multiple levels of detail paradigm is often invoked as a part of a coarse-fine or hierarchical matching strategy in image processing and interpretation. Given the scale and digitized ground resolution of an image, the IMD model can generate a map description that will suppress any features that would be too small to be recognized, with remaining descriptions at the appropriate level of detail. This technique is more than camera scaling and transformation, since the criterion for "too small" can be an attribute of the map feature itself. Consider the map feature description of a university campus. At some level of detail corresponding to pixel ground resolution distance (GRD), features such as playing fields, dormitories, instructional buildings and offices, access roads, and campus greenery are all individually distinguished. Using spectral properties of the features[****] and spatial relationships between these features, we can determine those feature boundaries that are likely to be muddled, and those with sufficient detail to be recognized.

The multiple level of detail paradigm need not be applied in a homogeneous manner. For example, tasks such as decision aids for photo-intelligence may require high resolution detail to support analysis, but low resolution detail to establish overall context. A large scale spatial organization containing urban, residential, and rural areas will require flexibility to represent the high feature density and complexity in the urban area as well as significantly lower density in rural areas.

Flexible knowledge acquisition is necessary because in photo-interpretation, situation assessment, and cartography, world knowledge is inherently fragmented. Knowledge fragmentation in these domains arises from:

- **methods of knowledge acquisition**
  There are diverse sources of knowlege that are used to acquire map feature information. Some of the most common are direct measurement from imagery, old maps and charts, sketches, and collateral data.
- **task requirements**
  If the task requirement is to support radar scene simulation, then elevated roads are significant, and road networks in general are not significant. If the task is to support map generation at a particular scale (say 1:50000), the feature size density may determine whether it is directly portrayed, generalized, or omitted entirely. There are, of course, well defined rules that govern these decisions, but they are generally not consistent across a wide range of map scales.
- **specialization in feature extraction**
  There is a certain amount of specialization in cartographic and situation assessment activities. Analysts may specialize in a particular area of the world, be knowledgable in hydrology, geology, local construction customs, or political matters. In the production of large scale maps it is rare to find map generalists, although this may not be true for low level feature extraction activities. This specialization tends to fragment knowledge, and is often given as a justification for building database systems that provide access to a wide range of map knowledge and may have general capabilities for knowledge synthesis.

The IMD model methodology provides a mechanism for feature unification in a cohesive framework. It provides a framework to relate symbolic descriptions to their original data sources. It is not tied to a particular cartographic representation nor to limitations of cartographic production.

## 4. The Database Problem in Image Interpretation

The database problem has been addressed in a variety of ways in systems that perform image analysis and interpretation. However, it has rarely been pursued as a separate research problem. One explanation for this is that portions of general database representation are often embedded in the experimental image processing systems and become highly tuned to the application. This is sometimes a result of system performance issues, or ease of task-specific implementations, but often it is a result of not recognizing the database problem as a separate issue.

It is difficult to give a precise analysis of the use of map databases in image interpretation, since the detailed organizations of experimental systems are rarely available. However, there are several recent examples. Work at SRI used a map database of road intersections to construct a camera model in the HAWKEYE and subsequent "road expert" systems[5, 6, 7].

The ARGOS[3, 4] system used a digitized city plan map and elevations for buildings to build a 3D graphics model of downtown Pittsburgh. This model was directly compiled into a knowledge network representation which described size, shape and relative positions of buildings, roads, rivers, and bridges for an arbitrary view point. Although it was not tied to a geodetic grid, it was a general map model.

---

[****] for example: roads preserve linear properties until the GRD approximately equals the width of the road

Recent work at Hughes[18] based on the ACRONYM system developed by Brooks and Binford[19] uses image registration to a geographic model. The system uses pre-selected regions of interest and attempts to locate and identify pre-defined object instances within these areas.

ACRONYM is currently the best example of a model-based system that incorporates viewpoint-insensitive mechanisms in terms of its model description. Its recognition process is to map edge-based image properties to instances of object models. In the domain of aerial photo interpretation, results have been reported for the recognition of a small number of models (3) for wide-bodied jets in aerial photographs. It is not clear how map knowledge would be directly integrated into the ACRONYM framework, but one could speculate that it could be added by a method similar to the work at Hughes described above.

Matsuyama[20, 21] has demonstrated a system for segmentation and interpretation of color-infrared aerial photographs containing roads, rivers, forests, and residential and agricultural areas. It uses rules to make assignments based on region adjacency and multi-spectral properties. These rules make use of informal map knowledge but do not directly use a particular map to guide interpretation. It generates good descriptions of a variety of fairly complex aerial scenes getting a great deal of constraint from the multi-spectral data.

In his recent thesis, Selfridge[22] proposed using adaptive threshold selection for region extraction by histogramming and region growing using an image-based "appearance model". Although the work describes feature positions and shapes in terms of pixel descriptions, it is not difficult to imagine a more general map-based approach that would result in the automatic generation of constaints to his adaptive operators.

At CMU, Herman[23] has demonstrated the feasibility of incremental acquisition of 3D scene descriptions from stereo-pair aerial photographs in the MAPS database in the 3D Mosiac project. This system requires a known stereo camera model but uses no *a-priori* knowledge about the scene other than weak geometric assumptions about urban environments.

## 5. The Image/Map Database Model

In this section we discuss four classes of tasks that are common to photo interpretation, situation assessment, and cartography. We then list some criteria by which one can evaluate the strengths and limitations of database systems. These criteria are not exhaustive, rather they point to four areas that should be present in IMD implementations and system capabilities in each of the areas.

### 5.1. Tasks for Image/Map Database

In this section we give a classification of tasks that are common to applications in photo-interpretation, situation assessment, and digital cartography systems. The four tasks are *selection* of image, terrain, or map data based on attributes of the data, *spatial computation* of map feature relationships, *semantic computation* of map features, and *synthesis* of imagery, terrain and map data.

1. **Selection**
   The selection task requires that the IMD system be able to select from a potentially large set of database entities based on attributes of image, terrain, and map database features. The selection task does not require image-to-map correspondence, and is the task normally performed by ID model systems. For example:
   - select imagery with particular intrinsic characteristics: sensor, scale, date, cloud cover, processing history
   - select map features based on symbolic description, partially specified description, similarities in image acquisition
2. **Spatial Computation**
   Spatial computation is ubiquitous in cartographic, situation assessment and photo-interpretation tasks. An IMD system must provide tools to compute common spatial relationships such as containment, closest point, adjacency, and intersection. One issue is how to structure the environment in order to constrain search and thereby avoid unnecessary computation. Consider four views of the same problem:
   - given a geodetic area, which images cover, or partially cover this area
   - which roads can be found within the image
   - which images contain this building
   - given an image, find all images which overlap it
3. **Semantic Computation**
   There are a number of tasks that require more than basic spatial computation, or where the appropriate spatial operation depends on the meaning of the map objects. Are there intrinsic high-level properties of map features that we can extract from basic spatial geometry that give a meaning to the feature? Semantic computation needs to be investigated as we develop more complex spatial databases. For example, what is the semantics of 'intersection' for the following pairs of map objects?
   - intersection of two roads
   - intersection of bridge and river description
   - intersection of a building and a road
4. **Synthesis**
   One goal of any database system should be to bring together diverse sources of knowledge into a common framework. Synthesis is the generation of new information using a new method of presentation, computation, or analysis. For example:

- cartographic superposition of map data on newly acquired image
- 3D display of terrain and cultural features from map database including man-made structures, political boundaries, neighborhoods, arbitrary collections of physically realized features
- to predict spatial (location) and structural (appearance) constraints; where to look and what to look for based of task knowledge, previous experience, or expectations
- a spatial framework within which to embedd task-specific knowledge

### 5.2. Criteria for Image/Map Database

In this section we list some criteria that can be uses to evaluate database systems in four general areas. These areas are image-to-map correspondence, map feature representation spatial computation, and database synthesis.

1. **Image-to-Map Correspondence**
   - can the it relate image-based features to a map coordinate system
   - can these features be projected onto new imagery using the correspondence mechanism
   - chat capabilities exist for incrementally updating feature descriptions based on updates to the camera model, or to intrinsic changes to the feature itself.
2. **Representation**
   - what are the capabilities for feature representation; what complex spatial relationships can represent; how is inconsistency recognized and handled
   - can the user describe features and associated attributes in a flexible manner; what is the variety of attributes.
   - can the representation accommodate map-based information coming from a variety non-imagery sources
   - what is the relationship between the representation of signal and symbolic data
   - what synthesis tasks does the representation support
3. **Spatial Computation**
   - does the system support dynamic spatial queries
   - what spatial relationships does the system compute directly from the underlying data, which relationships are specified by the user, how do they interact, how does one maintain consistency
   - what mechanisms are available to partition the search space when computing spatial relationships
4. **Database Synthesis**
   - imagery, terrain and map data are components, each with an appropriate representation, operation semantics, and utility; in what ways does the database support synthesis of these components
   - what concrete tasks requiring synthesis are performed

## 6. MAPS Overview

In the previous sections we have attempted to raise issues of Image/Map Database organization, tasks and capabilities. In this section we will discuss the MAPS system components capabilities. We will only briefly describe those aspects that have been reported on in other papers. Our latest work in the area of hierarchical organization, decomposition, and search is reported beginning in Section 6.6. New work in map feature semantics is discussed in Section 6.7. For a more detailed description of the image segmentation program (Section 6.1.2) and the image-to-map correspondence program (Section 6.3) see McKeown[24]. For a detailed description of the CONCEPTMAP database see McKeown[25]. Appendix I contains a nearly complete list of the programs associated with each system component.

### 6.1. BROWSE: Interactive Image/Map Display

BROWSE[26] is an interactive window-based image display system. It provides a common interface to all of the MAPS system components to display results of queries, graphical prompts for interactive image-to-map correspondence, superimpostion of map data on imagery, and other similar functions. While often viewed as an application issue, a flexible, functional user interface is critical for building more complex tools. BROWSE provides the user with a window-oriented interface, which greatly increases the effective spatial resolution of the frame-buffer, and provides multiple processing contexts which allow users to manipulate dynamically the size, level of detail, and visibility of imagery.

#### 6.1.1. Window-based Display

We have applied and extended the bit-map window[27] paradigm to handle high resolution, multi-bit per pixel digitized images. However, due to nearly an order of magnitude difference in the amount of data needed to perform screen updates and due to processing limitations found in most frame-buffer architectures, many of the solutions used for single bit per pixel displays[28] are not suitable for direct implementation. A detailed discussion of the design and organization of the window manager appears in McKeown & Denlinger[26].

Besides the display of imagery, we have found the window representation to be useful as a communication mechanism between MAPS components, to invoke image processing programs, and to retrieve and display the results of such processing. All MAPS components (see Appendix I) that display imagery, map data or graphics use the BROWSE window mechanism for display and communication. For example, the interactive image correspondence program, CORRES, uses the window mechanism to automatically display landmark image fragments and to create a high resolution window containing the approximate position of the landmark ground control point to cue the user. PICPAC contains a collection of image processing routines that can be invoked on BROWSE windows simply by specifying the window name. BROWSE routines use the window name to determine the image name, resolution, and rectangular image bounds. This information, along with parameters specific to the particular processing operation, are passed to the image processing routine. The results of the operation can be displayed in a new window.

### 6.1.2. Interactive Image Segmentation

SEGMENT is an interactive image segmentation program which uses the BROWSE window facility to provide an interface to our frame buffer. Users can extract image-based descriptions of map features, edit existing features, and assign symbolic names to the features. SEGMENT produces a standard format [SEG] file that is used throughout the MAPS database to represent image-based descriptions of point, line, and polygon geometric data. Database routines discussed in Section 6.5 are available to convert the [SEG] description to a map-based description [D3].

## 6.2. Image Database

The MAPS system currently contains approximately 100 digitized images, most of which are low altitude aerial mapping photographs. Typical ground resolution distances (GRD) are 120cm$^2$, 360cm$^2$, and 600cm$^2$ per pixel. The imagery is mainly comprised of three data sets taken in 1974, 1976 and 1982. In addition to aerial mapping photographs, we have several digitized maps including a USGS topographic map, and tour guide maps. Figure 1 gives the current status of the MAPS Washington D.C. image database. Although we have several Landsat, Skylab and high altitude aerial photographs taken over the Washington D.C. area, we have focused our work on those images that provide the greatest ground detail.

| CLASS | NUMBER | IMAGE SCALE | DATABASE RASTER | COMMENTS |
|---|---|---|---|---|
| ASC'74 | 25 | 1:36000 | 2048x2048x8 | Aerial mapping BW |
| WGL'76 | 37 | 1:12000 | 2200x2200x8 | Aerial mapping BW |
| AER'79 | 2 | 1:124000 | 2288x2288x8 | Color infrared |
| ASC'82 | 29 | 1:60000 | 2300x2300x8 | Aerial mapping BW |
| MAP'71 | 1 | 1:24000 | 4096x4096x8 | USGS topo map |
| MAP'74 | 1 | 1:160000* | 4096x3880x8 | D.C. region map |
| MAP'79 | 1 | 1:16000* | 4096x4096x8 | Tourist guide map |

* not cartographically accurate.

Figure 1: MAPS: Image Database Component

### 6.2.1. Generic Image to File Mapping

The MAPS system uses a generic naming convention to refer to images in the database. The generic name is a unique identifier assigned to the image when it is integrated into the database. For example, DC38617, DC1420 are representative generic names that correspond to flight line annotation on the photographic film. All types of image access that require the filesystem name of the image, or require associated image database files, use the generic name mechanism to construct the appropriate physical file name. It is possible to change the logical and/or physical location of imagery by updating the generic name file or to add another image to the database. As we move to larger image/map systems this naming isolation allows us to construct a database that can be distributed over multiple

The decoupling of name with physical or logical location fits well with name server organizations usually employed with such distributed systems.

The following table lists the database files associated with each active image in the MAPS database. Each is accessible using the generic image name.

- [GENERIC] **image-to-file system mapping**
  - contains the file system location of the database image
  - identifies which reduced resolution images are computed and available for hierarchical display
- [SDF] **scene description file**
  - contains image specific information: source, date, time of day, raster size, digitization, image scale, geodetic corner points, camera information
- [COF] **image-to-map coefficients file**
  - contains camera model coefficients, error model, polynomial orders solved, best correspondence (default polynomial order)
  - independent coefficients for <latitude>, <longitude>, <image row> <image column>
- [COR] **correspondence pairs file**
  - mapping of ground control points to image point specification
  - lists of landmark names and their geodetic position combined with image pixel position of landmark specified by user
- [HYP] **hypothesized landmark file**
  - lists of landmark names which are within the image geodetic coverage, but were not used to perform image-map correspondence

### 6.2.2. Image-Based Segmentations

MAPS maintains several types of image segmentations and map overlay descriptions associated with each image in the database. These segmentaions either are feature descriptions generated using the image as the base coordinate system, or the projection of map features onto the image using map-to-image correspondence, or segmentations from other images registered to the image. In the latter case, image-to-map correspondence is used to register the two images. Users can point to segmentation overlay features using the display interface in BROWSE and CONCEPTMAP, identify the segmentation feature name and retrieve its image and geodetic coordinates. For the [DLMSSEG] and [CONCEPTSEG] segmentation descriptions, the name of the segmentation feature is used to retrieve the associated DEAD (see Section 6.4) or

CONCEPTMAP description. The following table is a list of image segmentations associated with each image in the database. Segmentations that require map correspondence for their generation can be automatically recreated when image camera model is updated.

- [HANDSEG] **hand (human) segmentation**
  - collection of all hand segmentations performed on this image
- [HCOMPSEG] **composite hand segmentation**
  - collection of all features in the [HANDSEG] database that are spatially contained in this image
- [MACHSEG] **machine segmentation**
  - collection of all machine segmentations performed using the image
- [MCOMPSEG] **composite machine segmentation**
  - collection of all features in the [MACHSEG] database that are spatially contained in the image
- [DLMSSEG] **DLMS map overlay**
  - all features from the DLMS digital feature analysis database that are spatially contained in the image
- [CONCEPTSEG] **CONCEPTMAP map overlay**
  - all features from the CONCEPTMAP database that are spatially contained in the image
- [COVERSEG] **image coverage overlay**
  - all images whose area of coverage is overlapped or wholly contained within the image

### 6.3. Image-to-Map Correspondence

The MAPS system uses an interactive image-to-map correspondence procedure to place new imagery into correspondence with the map database. It has three major components: a landmark database, a landmark creation and editing program, and an interactive correspondence program. The process of landmark selection, description, and interactive correspondence has been described in detail in McKeown[24].

#### 6.3.1. Landmark Database

MAPS maintains a database of approximately 200 geodetic ground control points in the Washington D.C. area. Landmarks are acquired using USGS topographic maps, but in principle can be integrated from any source that provides accurate geodetic position *<latitude/longitute/elevation>*. Users can query the database to find landmarks by name, within a geodetic area, or the closest landmark to a geodetic point. Landmark features are also integrated into the CONCEPTMAP database and can be found using the *<role-derivation>* attribute (see Section 6.5.2) of a concept role schema.

#### 6.3.2. LANDMARK

LANDMARK is an interactive tool used to generate new landmarks, their text descriptions, and associated image fragments. The following information is maintained by LANDMARK to support landmark database access.

- [LDM] **landmark name directory**
  - associates the list of landmark names with their geodetic position
  - sorted for spatial proximity
  - partial name matching also provided
- [ETY] **landmark text description**
  - contains a detailed text description of the location of the landmark and general factual properties of the landmark
  - stores the location and name of the associated image fragment file [LIMG], and replicates the geodetic position from ldm file
- [LIMG] **landmark image fragment**
  - contains a high-resolution image fragment which clearly shows the ground control point and scene context around the point

#### 6.3.3. CORRES

CORRES is an interactive image-to-map correspondence program. It uses the BROWSE window interface, the LANDMARK database, and image database routines to interactively build an image-to-map correspondence. Once an initial guess of the corner points is performed and the [COR] and [COE] files have been created in the image database, CORRES automatically suggests new possible landmark points using the image database [HYP] files. The LANDMARK database [LIMG] files are used to display the ground control point when the user selects it from the list of hypothesized points.

### 6.4. DLMS: An External Database

The ability to rendezvous with externally generated map databases is a key capability in order to integrate information from a variety of sources. One example of the flexibility of the MAPS database is illustrated by our experiences with the Defense Mapping Agency's (DMA) Digital Landmass Simulation System (DLMS)[29].

DLMS is composed of a digital feature analysis database (DFAD) which describes man-made cultural features and a digital terrain elevation database (DTED) which is organized as a raster elevation grid. The specified resolution of the DFAD data is comparable to map scales of 1:250,000 to 1:100,000. The specified resolution of DTED data is within a meter vertical resolution over a $100^2$ meter (3 arc sec) grid.

#### 6.4.1. DFAD: Digital Feature Analysis Database

In order to integrate the DFAD database into MAPS, we reorganized the internal DFAD data structures to allow for random access using a feature header list. We converted the representation of geodetic coordinates from an offset format that was relative to an internal base coordinate, to an absolute coordinate system. Our DFAD database covers a two degree square area, from latitude N $38^0$ to N $40^0$ and longitude W $76^0$ to W $78^0$. It is composed of 64 "map sheets", each containing a 15'x15' map area. We assigned unique feature identifiers (names) to map features because feature numbers were not unique across map sheets. There are no feature names or semantics associated

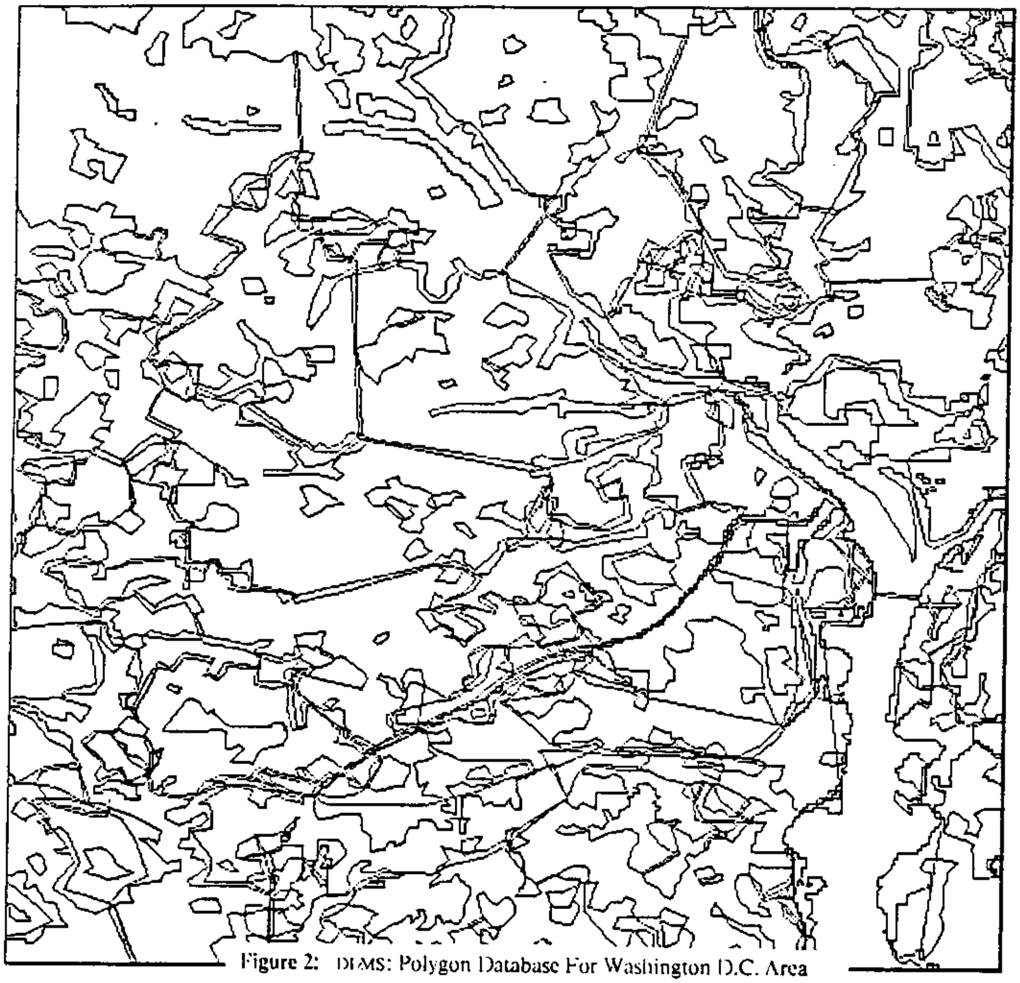


Figure 2: DLMS: Polygon Database For Washington D.C. Area

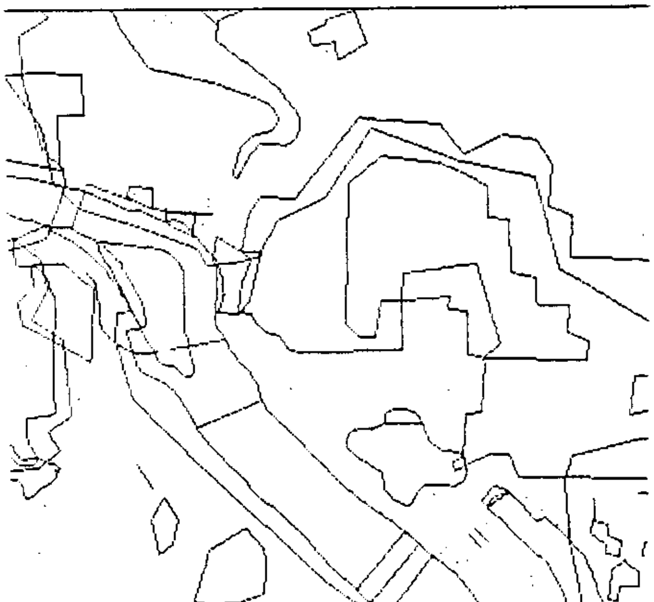


Figure 3: DLMS: Detail of Northwest Washington Area

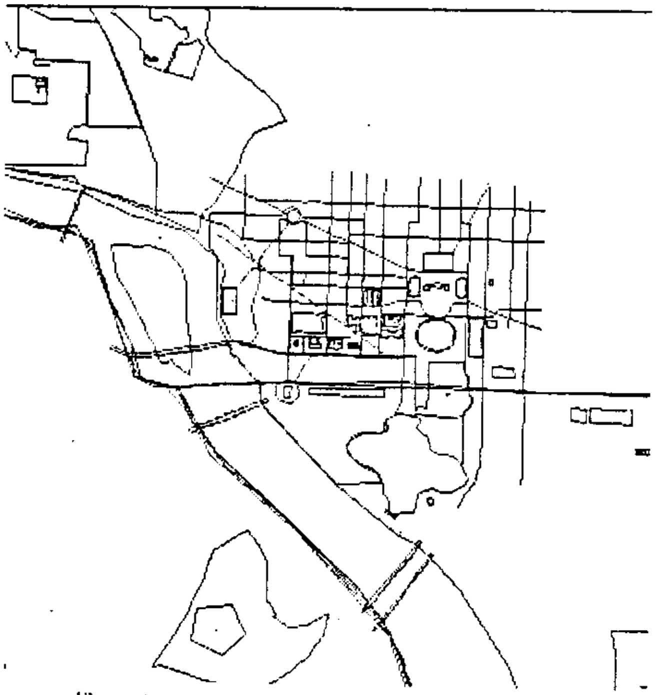


Figure 4: MAPS: CONCEPTMAP Database For Figure3

with DFAD entries primarily because the database was not intended to be used as a general purpose geographic information system. The feature header mechanism allows us to perform random access to features in a map sheet. We can also search using feature attributes such as feature analysis code, feature type, surface material code, and feature id code. This type of reorganization is necessary to support an interactive query-based interface for human and application programs.

Figure 2 shows a plot of polygon features in the area corresponding to our entire Washington D.C. database. Figure 3 is a detailed portion of the DFAD database centered on Foggy Bottom. For comparison, Figure 4 is the corresponding area from the CONCEPTMAP database plotted on the same scale.

Some of the DFAD database entries are easily recognizable as natural or man-made features, although as discussed, this information is not in the original database itself. Figure 5 is the description for the Tidal Basin, Figure 6 is the Rochambeau Bridge. Figure 7 is a description for a large irregular area in central Washington D.C. that contains the major government office buidings. The feature name assigned by MAPS is the first entry in each of the Figures.

```
feature 'd25f471a909'
feature header: 471   (seek:72416)
feature analysis code: 1082
feature type: areal feature
surface material code: (6) water
feature id code: (909) not assigned
subcategory: fresh water (shallow)
average height (meters): 0
aerial feature: 471   polygon with 76 vertices
tree cover: 0   roof cover: 0   density: 0
min point (south west) 5298,7979
max point (north east) 5567,8385
```

Figure 5: DFAD: Description for Tidal Basin

```
feature 'd25f4741250'
feature header: 474   (seek:73132)
feature analysis code: 1085
feature type: linear feature
surface material code: (3) stone / brick
feature id code: (250) not assigned
subcategory: not assigned (general)
average height (meters): 2
linear feature: 474   line with 3 vertices
width: 24   reflectivity: 2
first point: 5024,8064
last  point: 5192,8227
```

Figure 6: DFAD: Description for Rochambeau Bridge

```
feature 'd25f402a610'
feature header: 402   (seek:63688)
feature analysis code: 1010
feature type: areal feature
surface material code: (3) stone / brick
feature id code: (610) not assigned
subcategory: institutional (general)
average height (meters): 28
aerial feature: 402   polygon with 27 vertices
tree cover: 10   roof cover: 70   density: 3
min point (south west) 5705,7971
max point (north east) 6260.8799
```

Figure 7: DFAD: Description for Government Buildings

### 6.4.2. DTED: Terrain Elevation Database

The organization of the digital terrain database is more straightforward. The DTED database covers the same geodetic area as our DFAD data. It is organized into 64 raster images using the same image format as our digital aerial imagery. Each image containing a 15' x 15' array of terrain samples, where each "pixel" is a discrete elevation point. The terrain package, ELEVATION, provides a transparent interface to the DTED database. Users can retrieve elevation information based on rectangular geodetic area, closest sample point to a geodetic point, or by weighted interpolation. ELEVATION uses the CMU image package to efficiently buffer blocks of contiguous terrain data.

### 6.5. Conceptual Map Database

The map database component of MAPS, CONCEPTMAP, has been described in McKeown[25]. We will give a brief overview of the organization and concentrate on our new work in hierarchical organization and feature semantics.

### 6.5.1. Concept Schema

The basic entity in the CONCEPTMAP database is the concept schema. The schema is given a unique ID by the database, and the user specifies a 'symbolic' print name for the concept. Each concept may have one or more role schema associated with it. Role schema specify one or more database views of the same geographic concept. For example, 'northwest washington' can be viewed as a residential area as well as political entity. Another aspect is the ability to associate the same name to two different but related spatial objects. Consider the 'kennedy center' as a building and as the spatial area (ie. lawn, parking area, etc.) encompassing the building. The principle role of a concept schema indicates a preferred or default view. The CONCEPTMAP database is composed of lists of concept schema.

### 6.5.2. Role Schema

The role schema is a further specification of the attributes of the map feature. It contains the *role name* attribute (building, bridge, commercial area, etc.), a *subrole name* attribute (house, museum, dormitory, etc.), a *role class* attribute (ie., buildings may be *government, residential, commercial*, etc.), a *role type* attribute (ie. physical, conceptual or aggregate), and a *role derivation* attribute (ie. derivation method).

The role name, subrole, and role class attributes categorize the map feature according to its function. For example: this feature is a building, used as an office building, used for government purposes. The role type attribute describes whether the map feature is physically realized in the scene, or if it is a conceptual feature such as a neighborhood, political, or geographic boundary. The role type attribute also provides a mechanism to define the role schema as a collection of physical or conceptual map features. For example, the concept schema in MAPS for 'district of columbia' has a role type aggregrate-conceptual, with aggregrate roles, 'northwest washington', 'northeast washington', 'southwest washington', and 'southeast washington'. This mechanism allows the user to explicitly represent concepts that are strictly composed of other role schema. The role derivation attribute describes the method by which the role and its associated geodetic position description were added to the CONCEPTMAP database.

Each role schema contains a 3DID identifier that is used to access a set of CONCEPTMAP database files which contain geodetic information about the map feature. These identifiers can be shared when multiple roles have the same geodetic description, as in the previous example of 'northwest washington' viewed as both a residential and political area. The CONCEPTMAP 3D description allows for point, line, and polygon features as primitives, and permits the aggregration of primitives into more complex topologies, such as regions with holes, discontinous lines, and point lists. Associated with each feature that was acquired from a image in the database is the generic name of the image. If the correspondence of the generic image changes due to the addition of more ground control points, or better a camera model, the position of the ground feature can be automatically recalculated.

The following is the set of files associated with each 3DID.

- [D3] **3D geodetic location**
  - a set of <latitude/longitude/elevation> triples which define the geodetic position of the role
- [D3F] **3D feature shape description**
  - metric values for lenght, width, area, compactness, centroid, fourier shape approximation etc.
- [FC] **feature image coverage**
  - a list of generic images which contain this feature
  - image mbr and feature coordinates for each image
- [PROP] **feature property list**
  - list of properties of the map feature
  - some general properties such as 'age', 'capacity', '3D display type'
  - feature type specific properties such as 'number of floors', 'basement', 'height', and 'roof type' for buildings

### 6.5.3. Database Query

CONCEPTMAP supports four methods of database query. The methods are *signal access*, *symbolic access*, *template matching* and *geometric access*. The following table gives a brief description of each query method.

- **signal access**
  Given a geodetic specification (point, line, area)[*****], perform the following operations:
  - display all imagery at which contains point, line or area.
  - retrieve all map features within geodetic specification
  - retrieve terrain elevation
- **symbolic access**
  Given a symbolic name, such as 'treasury building' perform the following operations:
  - convert name into geodetic specification to perform signal access operations listed above
  - retrieve database description, facts and properties of the map feature
  - retrieve imagery based on symbolic (generic) name
- **template matching**
  Given a partial specification of symbolic attributes perform the following operations:
  - find all map features which satisfy the specification template and return their symbolic name
  - find all images and return symbolic (generic) name
- **geometric access**
  Given a geometic operation such as 'contains' and a geodetic specification perform the following operations:
  - find all map features which satisfy the operation performed over the geodetic specification and return their symbolic name. - find all image features and return symbolic name

These primitive access functions can be combined[25] to answer queries such as: 'display images of Foggy Bottom before 1977', 'what is the closest commercial building to this geographic point', and 'how many bridges cross between Virginia and the District of Columbia'. Figure 8 is a simple schematic giving the processes by which MAPS provides *signal* and *symbolic* access into the CONCEPTMAP database and display of the query result.

### 6.5.4. Spatial Computation

CONCEPTMAP computes geometric properties based on the geodetic descriptions associated with each role schema in the database. A static description of all spatial relationships between map features for contains, subsumed by, intersection, adjacency, closest point, partitioned by is maintained in the database.

- **'contains'**
  - an unordered list of features which the map feature contains
- **'subsumed by**
  - an unordered list of features which contain the map feature

---

[*****] this specification may be in geodetic coordinates or require image-to-map correspondence

- **'intersection'**
  - an unordered list of features which intersect the map feature
- **'closest point'**
  - single feature which is closest to the map feature
- **'adjacency'**
  - an unordered list of features that are within a specific distance of the map feature
- **'partitioned by'**
  - the locus of points where two areal features share a common boundary.

If one or more of the map features in a spatial computation is a result of a dynamic query (and therefore not in the static database), these relationships are computed as needed. A simple 'memo' function is implemented to avoid recomputation of dynamic properties. The use of the static description can also be 'turned off' to evaluate hierarchical search as described in the following section.

The CONCEPTMAP database stores both factual and exact information describing the spatial relationship. For example, if two features intersect, the list of geodetic intersection points is stored, as well as the fact that they intersect at least once. This is necessary for query which require the display of imagery containing a geometric fact, and may possibly be useful for describing the semantics of the intersection. In the following section we will discuss the use of a hierarchical organization based on the 'contains' relation primitive, and show how it can be used to structure the spatial database.

### 6.6. Hierarchical Organization

In this section we discuss the use of hierarchical organization of spatial data in the MAPS system. The CONCEPTMAP database is used to build a *hierarchy tree* data structure which represents the whole-part relationships and spatial containment of map feature descriptions. This tree is used to improve the speed of spatial computations by constraining search to a portion of the database. In the following sections we briefly discuss why we believe this is a good alternative to regular spatial decompositions such as quadtree[15, 16], or k-d tree[17] usually proposed for MPD model databases.

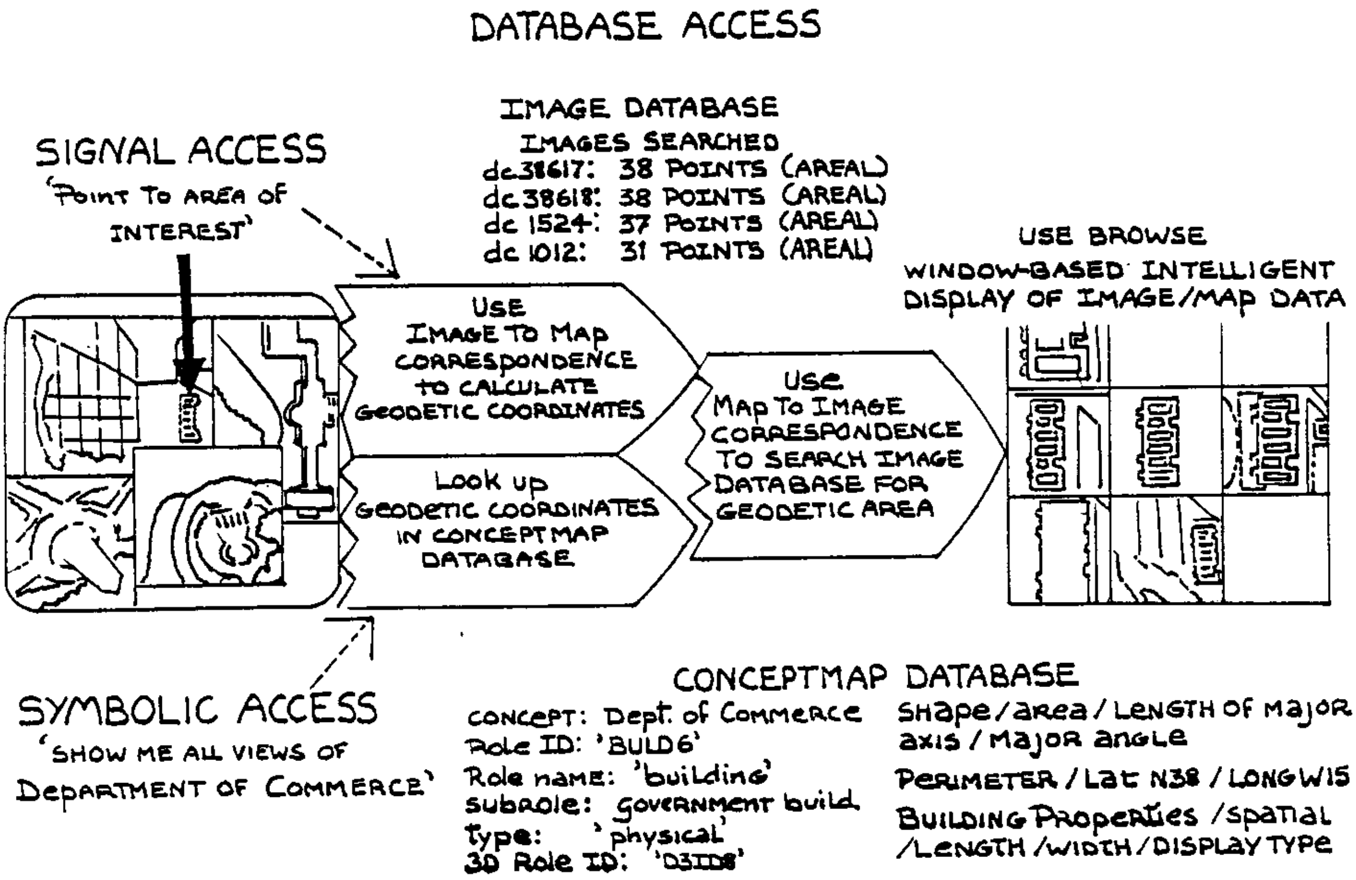


Figure 8: MAPS: Signal / Symbolic Database Access

### 6.6.1. Regular Decomposition

Regular decompositions such as the quadtree organizations do not explicitly exploit the inherent structure in spatial organizations. Practical implementations of these organizations often use image-based (integer) coordinate systems and therefore have a bounded position resolution. In general cartographic systems it is important to be able to represent and manipulate map feature descriptions at radically different resolutions using a real valued coordinate system. For example, consider a dynamic query that results in the creation of a very small polygonal area. When computing containment or intersection against a static map database with features represented as a quadtrees, the quadtrees for the static map feature must be generated to a much finer level of detail in order to compare the two data structures. Recent work is beginning to represent quadtrees on real valued coordinate systems[15], but little is known of its practical implementation, complexity, and storage efficiency. K-d trees show storage efficiency improvements over quadtrees[17], since they allow for a more flexible decomposition tailored to spatial feature density. However, they have the same fundamental limitations when used to represent map features in a real valued coordinate system.

In MAPS we perform geometric computations on the feature data in the geodetic coordinate system using point, line, and polygon as map primitives. We constrain search by using a hierarchical representation computed directly from the underlying map data. These spatial contraints can be viewed as natural, that is, intrinsic to the data, and may have some analogy to how humans organize a "map in the head" to avoid search. For example, when a tourist who is looking for the Watergate Hotel is told that the building is in Northwest Washington, she will not spend much time looking at a map of Virginia. Depending on her familiarity with the area, she may avoid looking at much of the map outside of the Northwest District.[******] As we begin to represent large numbers of map features with more complex interrelationships, we believe that the use of natural hierarchies in urban areas, such as political boundaries, neighborhoods, commercial and industrial areas, serve to constrain search. They may also allow us to build systems that organize data using spatial relationships that are close to human spatial models.

### 6.6.2. Hierarchical Decomposition

The hierarchical containment tree is a tree structure where nodes represent map features. Each node has as its descendants those features that it completely contains in *<latitude/longitude/elevation>* space. The hierarchical tree is initially generated by obtaining an unordered list of features (containment list) for each map database feature. Starting with a designated root node ('greater washington d.c.') which contains all features in the database, descendant nodes are recursively removed from the parent node list if they are already contained in another descendant node. The result is that the parent node is left with a list of descendant features that are not contained by any other node. These descendant nodes form the next level of an N-ary tree ordered by the 'contains' relationship. This procedure is performed recursively for every map feature. Terminal nodes are point and line features, or areal features that contain no other map feature. We will discuss the point containment and closest point computation using the hierarchy tree in the following section.

Figure 9 shows a small section of the hierarchical containment tree. The use of conceptual features-- features with no physical realization in the world but represent well understood spatial areas-- can be used to partition the database. In this case the map feature 'foggy bottom'

```
  41 entries for `contains' for `northwest washington'
  entry 0: `mcmillan reservoir (role: 0)'
  entry 1: `kennedy center (role: 0)'
  entry 2: `ellipse (role: 0)'
  entry 3: `executive office building (role: 0)'
  entry 4: `white house (role: 0)'
  entry 5: `treasury building (role: 0)'
  entry 6: `department of commerce (role: 0)'
  entry 7: `museum of history and technology (role: 0)'
  entry 8: `key bridge (role: 0)'
  entry 9: `thomas circle (role: 0)'
  entry 10: `dupont circle (role: 0)'
* entry 11: `foggy bottom (role: 0)'
  entry 12: `whitehurst freeway (role: 0)'
  entry 13: `mclean gardens (role: 0)'
  entry 14: `macomb playground (role: 0)'
  entry 15: `theodore roosevelt island (role: 0)'
  entry 16: `interior department (role: 0)'
  entry 17: `district building (role: 0)'
  entry 18: `lafayette park (role: 0)'
  entry 19: `constitution hall (role: 0)'
  entry 20: `national press building (role: 0)'
  entry 21: `23rd street (role: 0)'
  entry 22: `constitution avenue (role: 0)'
  entry 23: `virginia avenue (role: 0)'
  entry 24: `national zoo (role: 0)'
  entry 25: `georgetown (role: 0)'
  entry 26: `glover park (role: 0)'
  entry 27: `national cathedral (role: 0)'
  entry 28: `21st street (role: 0)'
  entry 29: `north 20th street (role: 0)'
  entry 30: `19th street (role: 0)'
  entry 31: `east pennsylvania avenue (role: 0)'
  entry 32: `e street (role: 0)'
  entry 33: `treasury place (role: 0)'
  entry 34: `state place (role: 0)'
  entry 35: `26th street (role: 0)'
  entry 36: `west pennsyvania avenue (role: 0)'
  entry 37: `16th street (role: 0)'
  entry 38: `l street (role: 0)'
  entry 39: `vermont avenue (role: 0)'
  entry 40: `13th street (role: 0)'

* 11 entries for `contains' for `foggy bottom'
  entry 0: `kennedy center (role: 1)'
  entry 1: `washington circle (role: 0)'
  entry 2: `state department (role: 0)'
  entry 3: `american pharmaceutical association (role: 0)'
  entry 4: `national academy of sciences (role: 0)'
  entry 5: `federal reserve board (role: 0)'
  entry 6: `national science foundation (role: 0)'
  entry 7: `civil service commission (role: 0)'
  entry 8: `c street (role: 0)'
  entry 9: `22nd street (role: 0)'
  entry 10: `south new hampshire avenue (role: 0)'
```

Figure 9: MAPS: Hierarchical Spatial Containment

---

[******] If she is told that the Watergate is also near the Potomac river, that should further constrain her search, but that is another story.

allows us to partition some of the buildings and roads that are contained within 'northwest washington'. As more neighborhood areas and city districts are added to our database, we expect to see improved performance especially in areas with dense feature distributions. This will also improve the richness of the spatial description available to the user.

### 6.6.3. Hierarchical Search

In this section we discuss the use of our hierarchical organization to partition the map database to improve performance by decreasing search when computing the spatial relationships of map features. The hierarchical searching algorithm is basically an N-ary tree searching algorithm. Consider a user at the CONCEPTMAP image display who invokes the geometic database to compute a symbolic description of what map feature he is pointing at. First, using image-to-map correspondence, the system. calculates the following map coordinates:

```
latitude  N 38 53 49 (276)
longitude W 77 03 53 (337)
```

This point is converted into a temporary map database feature and is tested against the root node of the hierarchy tree. If it is not contained in this node (not generally the case), then the point cannot correspond to a database feature, and the search terminates. The user is informed that the point is outside the map database.[*******] If the 'contains' test succeeds, it recurses down the tree and performs the test against the siblings of the node just tested. The search allows several paths to exist for any point, thus more than one sibling may contain a path to the point. This sort of anomaly occurs when a feature happens to exist in the intersecting region of two larger regions. However, if the feature is not contained by the node, it is not contained by any of the node's descendants, and that portion of the tree is not further searched. Figure 10 shows the answer to our hypothetical query. The query point is contained within 'theodore roosevelt island', and two search paths in the containment tree are given. The same mechanism is used for line and polygon features, although the primitive determination of containment depends on the geometric type of the feature.

```
This node belongs in the following place(s):
3 entries for 'contains' for 'theodore roosevelt island'
entry 0:          'northwest washington'
entry 1:          'district of 'columbia'
entry 2:          'greater washington d.c.'
        ************* A N D *************
2 entries for 'contains' for 'theodore roosevelt island'
entry 0:          'potomac river'
entry 1:          'greater washington d.c.'
```

Figure 10: MAPS: Containment Tree Entry for Theodore Roosevelt Island

### 6.7. Toward Feature Semantics

We have begun to investigate the generation of map feature semantics directly from the hierarchical representation of the map feature data. A simple example is the semantic description of a bridge: the feature names and map locations that it connects as well as the names of the map features that it crosses over. Figures 11 and 12 show the result of applying a procedural description of the semantics of a bridge concept to calculate the 'connects' and 'crossover' relationship using the map feature descriptions of 'arlington memorial bridge' and 'theodore roosevelt memorial bridge'. These results are generated directly using the MAPS hierarchical organization for spatial data. We do not pose this as a theory of map feature semantics, but envision a set of feature specific procedures that can build these types of descriptions.

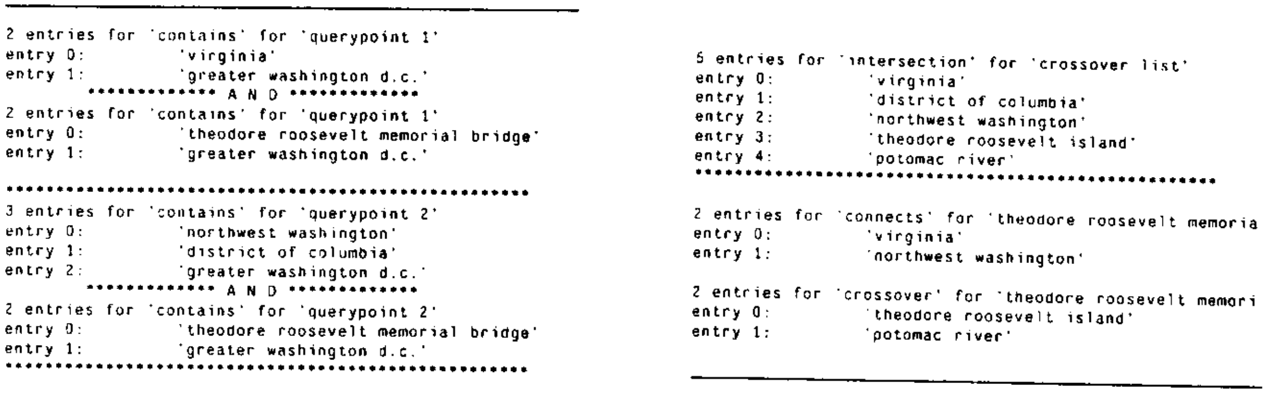
```
2 entries for 'contains' for 'querypoint 1'
entry 0:          'virginia'
entry 1:          'greater washington d.c.'
        ************* A N D *************
2 entries for 'contains' for 'querypoint 1'
entry 0:          'arlington memorial bridge'
entry 1:          'greater washington d.c.'

*******************************************
4 entries for 'contains' for 'querypoint 2'
entry 0:          'mall area'
entry 1:          'southwest washington'
entry 2:          'district of columbia'
entry 3:          'greater washington d.c.'
        ************* A N D *************
2 entries for 'contains' for 'querypoint 2'
entry 0:          'arlington memorial bridge'
entry 1:          'greater washington d.c.'

*******************************************
5 entries for 'intersection' for 'crossover'
entry 0:          'virginia'
entry 1:          'district of columbia'
entry 2:          'southwest washington'
entry 3:          'mall area'
entry 4:          'potomac river (Role: 0)'
*******************************************

2 entries for 'connects' for 'arlington memorial bridge'
entry 0:          'virginia'
entry 1:          'mall area'

1 entries for 'crossover' for 'arlington memorial bridge
entry 0:          'potomac river'
```

Figure 11: MAPS: Semantic Computation from Spatial Data Arlington Memorial Bridge

The procedure for bridge semantics is as follows: A bridge can be represented in the CONCEPTMAP database as an polygonal area, a list of linear segments, or as a geodetic point. The polygonal area arises when the bridge deck is represented, the list of linear segments approximates the center line of the bridge, and the point feature generally represents that the bridge is a landmark feature. No semantics are computed in the latter case. If the bridge is represented as a line, the end points are selected, otherwise the endpoints of the major axis of the bounding ellipse are retrieved from the feature [DBF] file. At some level of description, these endpoints define the 'connects' relationship, but this

---

[*******] This can actually occur since users are allowed to enter arbitrary coordinates through the terminal. Therefore the database has some crude idea of its extent of map knowledge



```
2 entries for 'contains' for 'querypoint 1'
entry 0:          'virginia'
entry 1:          'greater washington d.c.'
        ************ A N D ************
2 entries for 'contains' for 'querypoint 1'
entry 0:          'theodore roosevelt memorial bridge'
entry 1:          'greater washington d.c.'

***********************************************
3 entries for 'contains' for 'querypoint 2'
entry 0:          'northwest washington'
entry 1:          'district of columbia'
entry 2:          'greater washington d.c.'
        ************ A N D ************
2 entries for 'contains' for 'querypoint 2'
entry 0:          'theodore roosevelt memorial bridge'
entry 1:          'greater washington d.c.'
***********************************************
```

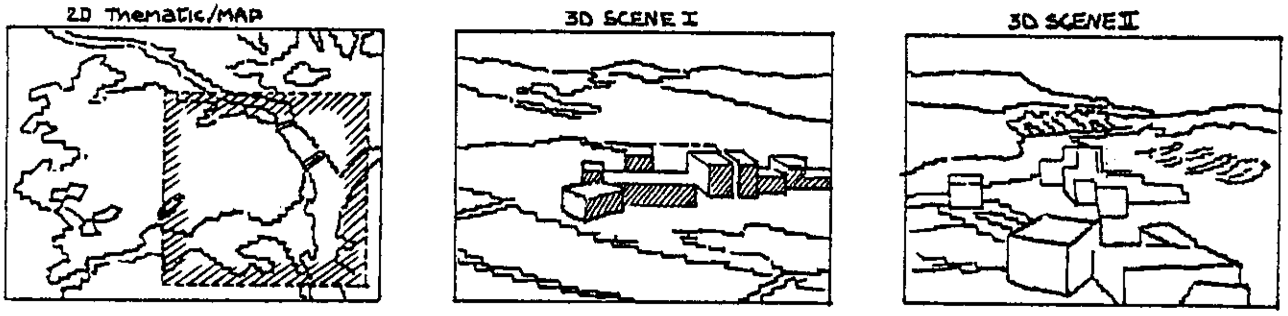
```
5 entries for 'intersection' for 'crossover list'
entry 0:          'virginia'
entry 1:          'district of columbia'
entry 2:          'northwest washington'
entry 3:          'theodore roosevelt island'
entry 4:          'potomac river'
***********************************************

2 entries for 'connects' for 'theodore roosevelt memoria
entry 0:          'virginia'
entry 1:          'northwest washington'

2 entries for 'crossover' for 'theodore roosevelt memori
entry 0:          'theodore roosevelt island'
entry 1:          'potomac river'
```

Figure 12: MAPS: Semantic Computation from Spatial Data
Theodore Roosevelt Memorial Bridge

## 3D MAP DISPLAY

MAPS can be used to generate a 3D scene of a designated area by combining terrain, conceptmap database and thematic map data

USER specifies area of interest by symbolic or signal access, also specifies 3D viewing position and illumination position



Displays thematic color data

▨ Metal
▦ stone/brick
░ water
≈ composition
⁙ earthen works
▩ trees
buildings

Searches terrain database to elevation

<W ^N

| | | | | | |
|---|---|---|---|---|---|
| 15 | 15 | 15 | 15 | 15 | 15 |
| 13 | 13 | 14 | 15 | 14 | 13 |
| 12 | 12 | 12 | 13 | 12 | 12 |
| 11 | 11 | 11 | 11 | 11 | 11 |
| 9 | 8 | 10 | 10 | 10 | 10 |
| 8 | 8 | 9 | 9 | 9 | 9 |

LAT: N38 53 21 (499)
LON: W77 2 8 (149)

Searches conceptmap database

Political

| | |
|---|---|
| CONCEPT2 | district of columbia |
| CONCEPT3 | NORTHWEST WASHINGTON |

Natural Features

| | |
|---|---|
| CONCEPT1 | tidal basin |
| CONCEPT43 | anacostia |

Residential

| | |
|---|---|
| CONCEPT51 | georgetown |
| CONCEPT 54 | buleith |

Figure 13: WASH3D: 3D Map Display

is not useful if we are envisioning generation of a reasonably complex symbolic representation.

The 'contains' relationship is applied to each endpoint using the hierarchical tree to order the search. As before, this search returns a list of features ordered by spatial containment, and there may be several independent containment paths. Redundant paths are eliminated by examining whether the bridge is in the containment path. The first entry (0) in each of the remaining paths is one of the areas connected by the bridge. Using the 'contains' relationship, the other entries in the path are also valid connecting areas.

To compute the crossover' relationship, the 'intersection' relationship is computed for the bridge using the complete list of line segments or the polygonal description. A list of all the features that the bridge intersects is assembled. Entries in the intersection list are removed if they are also present in either of the 'connects' lists. The assumption is that those features that didn't contain a bridge endpoint, but intersected with the bridge description, are those features that the bridge crosses over. If there is sufficiently detailed elevation data for man-made features it should be possible to compute semantics for 'passes over' and 'passes under' by calculating the feature elevation at the actual geodetic point of intersection.

## 7. Synthesis Tasks

In this section we will discuss three applications of the MAPS database to cartographic and image interpretation tasks. These tasks are 3D scene generation of views of Washington D. C., the use of the map database to guide image segmentation, and some preliminary results on a rule-based system for airport scene interpretation. Each task requires the capabilities of various aspects of the IMD model as implemented in the MAPS system. These applications pull together external and image/map databases, and are only possible using an integrated system that relates imagery, terrain, and map data through a unified cartographic representation.

### 7.0.1. WASH3D: 3D Scene Generation

The first application of the MAPS database is in the area of 3D computer graphics for scene simulation and database validation. Computer graphics play an important role in the areas of image processing, photo-interpretation, and cartography. In cartography various phases of the map generation process use graphics techniques or source material analysis, transcription and update, and some aspects of map layout and production. However, many major steps in the generation of a cartographic product remain largely manual. One important step for which inadequate tools exist is the integration of terrain and cultural feature databases. This integration step is often

Figure 14: WASH3D: Vertical View $85^0$ Northwest Washington

used to verify the geodetic accuracy of natural and man-made features in the digital database prior to actual map layout and production. Another application is sensor simulation[30, 31]. Radar, visual, and multi-sensor scenes are digitally generated to verify the quality of digital culture and terrain databases or to determine the quality of the sensor model. Improvements to the level of detail contained in the underlying database can be subjectively measured in terms of the quality of the generated scene.

WASH3D[32] is an interactive graphics system that uses the MAPS system to integrate a digital terrain database, a cultural feature database, and the CONCEPTMAP database to allow a user to generate cartographically accurate 3D scenes for human visual analysis. WASH3D uses the coarse resolution DLMS database described in Section 6.4 to generate a baseline thematic map. The thematic map is a 2D image which is produced by scan conversion of the DLMS digital feature analysis database (DFAD) polygon database. We assign a color to each region polygon using the DFAD surface material code-- forest and park (green), water (blue), residential (yellow), and high-density urban (brown). DLMS terrain elevation data (DTED) is interpolated to determine ground elevations at each point in the 2D image. Since the resolution of the DFAD data is coarse, comparable to map scales of 1:250,000 to 1:100,000, we use the CONCEPTMAP database to provide high resolution 3D feature descriptions of buildings, roads, bridges, residential and commercial areas. The CONCEPTMAP database is derived from imagery with resolutions between 1:12000 and 1:36000, and the addition of these features effectively intensifies the perceived level of detail in the simulated scene, even though the base map is at a coarse resolution.

Lukes[33] describes the utility of selective database intensification for tailoring standard database products to custom applications and for time-critical applications which cannot be handled by normal production schedules. Figure 13 shows the interactive process by which users can specify an area of interest for 3D scene generation. Figures 14 and 15 show two 3D scenes of the Washington D.C. area generated by WASH3D.

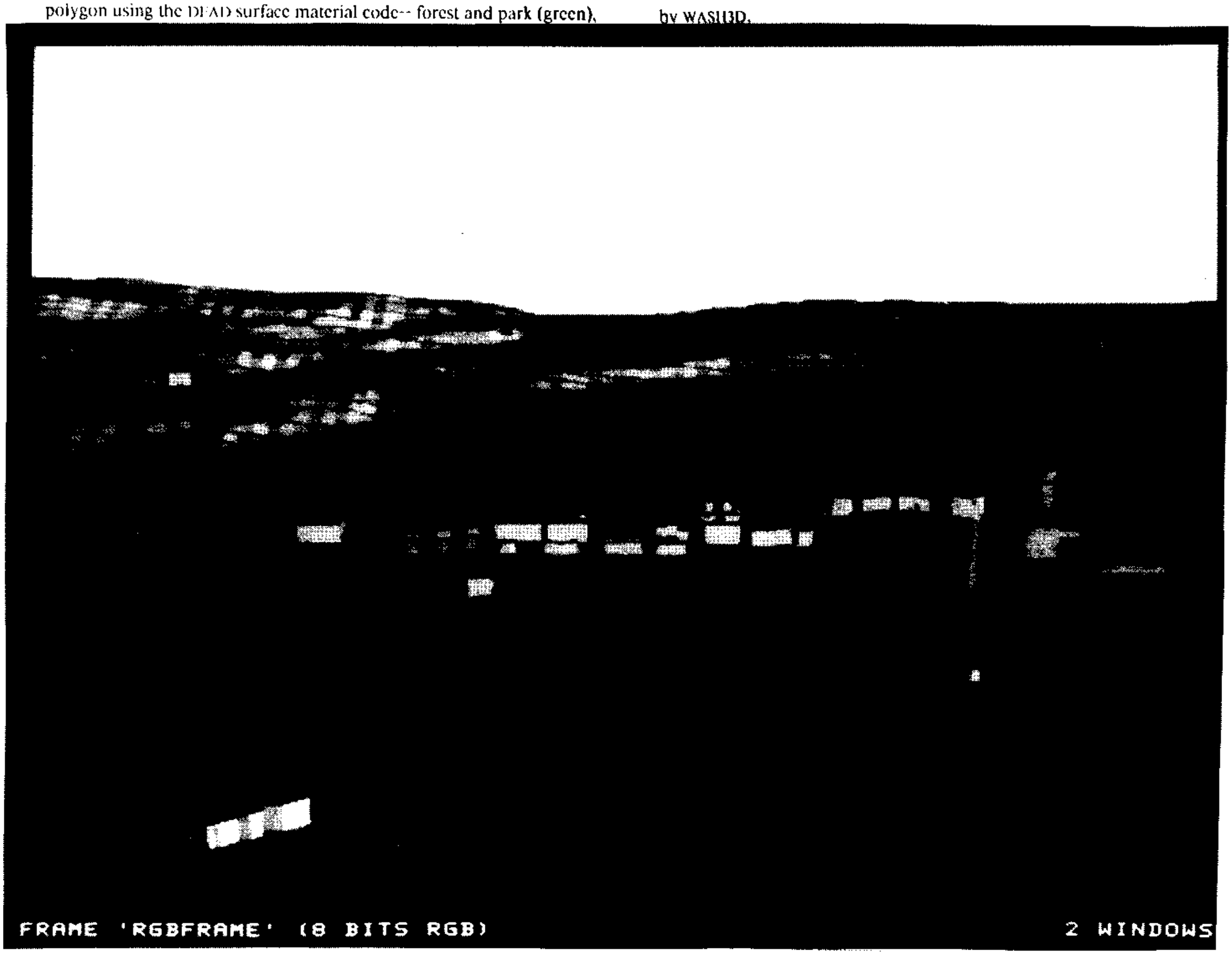


Figure 15: WASH3D: Northwest Washington From Above National Airport

### 7.0.2. MACHINESEG: Map-Guided Machine Segmentation

The second application of the MAPS database is in the area of map-guided machine segmentation. Users may specify a map feature from the CONCEPTMAP database or interactively generate a feature description using the SEGMENT program. In the case of a map database feature, MACHINESEG uses an existing image coverage [IEC] file (see Section 6.5.2) that specifies in which images the feature is found, and the feature location in the image. For interactive specification, an [IEC] file is created dynamically by image-to-map correspondence using the image database.

For each image, a high resolution window containing the database feature is extracted and displayed. We expand the size of the image window to contain an area of uncertainty around the feature location. The expansion is currently based on the size of the feature, but we plan to incorporate correspondence error measures based on the quality of the camera model associated with each image. The image window is smoothed, and a segmentation is performed using a region-growing technique[34] which combines an edge strength metric and region merge acceptability based on spectral similarity to control region growing.

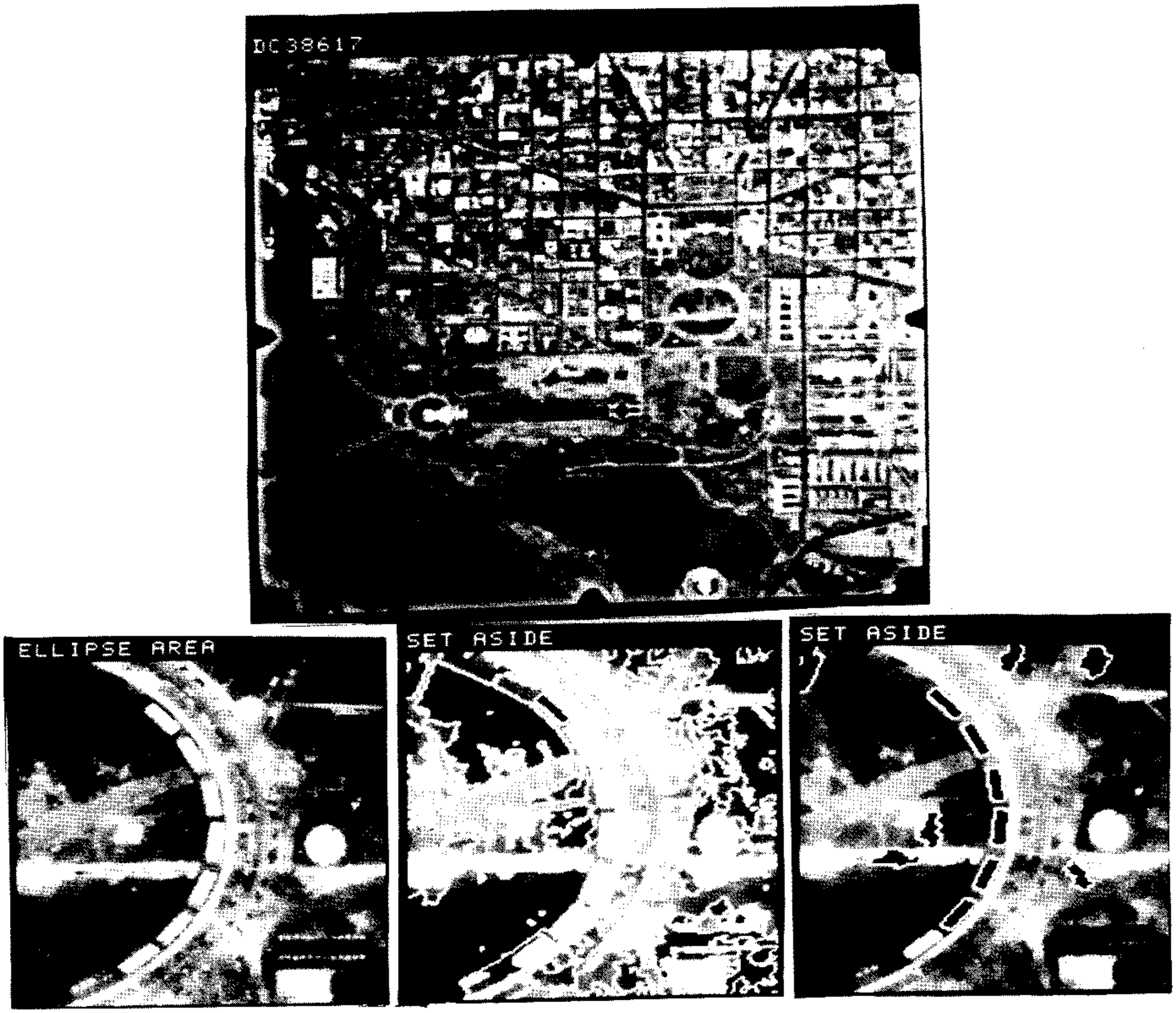


Figure 16: MACHINESEG: Segmentation using MAPS System

Figure 16 shows the segmentation of several low-elevation buildings along the perimeter of the Washington Ellipse. The uppermost building is added to the CONCEPTMAP database in the standard manner described in Section 6.5. The user specifies the image, DC38617, to perform the segmentation and the MACHINESEG system automatically displays a reduced resolution window of the image (*dc38617*), and a high resolution window (*ellipse area*) containing the database area. MACHINESEG creates a copy of the high resolution window as a work area (*set aside*) for the image processing routines. An image smoothing operation is followed by the generation of seed regions using a conservative similarity measure to insure that potentially matchable regions are not prematurely merged. The initial seed regions are overlaid on the image using graphics overlays. Any seed regions that satisfy the shape criteria for the database feature are extracted and marked. In this example, the database feature itself was marked in the initial seed region matching. As regions are merged based on weak edge boundaries and high spectral compatibility, the resulting region is evaluated with respect to a list of shape and spectral criteria. If the region satisfies the criteria, it is marked, and further merging is allowed only if the proposed merge improves the overall region score. Criteria include fractional fill, area, linearity, perimeter, compactness, and spectral measures.

The final results are shown in the second window labeled *set aside*. Five buildings similar to the map database feature were correctly identified while one building was omitted. Six segments were incorrectly identified. Had we made use of spectral information in this particular segmentation-- that the building roofs were bright features-- we probably could have excluded 5 of the 6 errors. However, we are more concerned with using weak knowledge, and one cannot expect better performance without more sophisticated analysis. MACHINESEG allows the user to delete erroneous segments and generates map descriptions of each extracted feature. These descriptions can then be used to search for these features in other database imagery.

The significance of MACHINESEG is that it can search systematically for features in a database of images, an operation that is fundamental for change detection applications. It directly uses the map database description as an evaluation tool for image segmentation and interpretation. It also uses very general image processing tools to perform both segmentation and evaluation and is amenable to supporting other approaches to image segmentation and feature recovery. A further application of the MACHINESEG system is discussed in the following section.

### 7.0.3. SPAM: Rule-based System for Airport Interpretation

The third application of the MAPS system is in the investigation of rule-based systems for the control of image processing and interpretation with respect to a world model.

In photo-interpretation, knowledge can range from stereotypical information about man-made and natural features found in various situations (airports, manufacturing, industrial installations, power plants etc.) to particular instantiations of these situations in frequently monitored sites. It is crucial for photo-interpretation applications that the metrics used be defined in a cartographic coordinate system, such as *<latitude/longitude/elevation>*, rather than an image-based coordinate system. Descriptions such as "the runway has area 12000 pixels" or "houses are between 212 and 345 pixels" are useless except for (perhaps) the analysis of one image. It is the case, however, that to operationalize metric knowledge one must relate the world model to the image under analysis. This should be done through image-to-map correspondence using camera models which is the method used in our system.

We have begun to build SPAM[35] to test our ideas in the use of the combination of a map database, task independent low-level image processing tools, and a rule-based system.

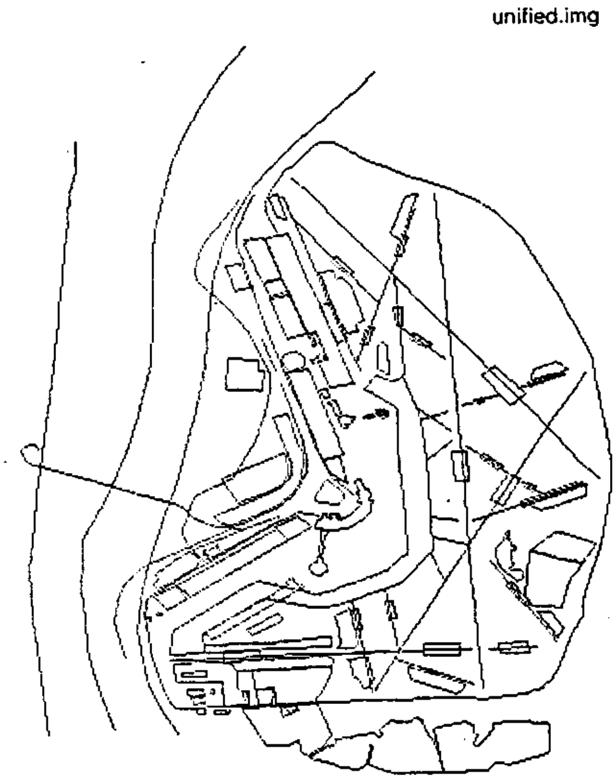
SPAM uses the MAPS database to store facts about man-made or natural feature existence and location, and to perform geometric computation in *map space* rather than *image space*. Differences in scale, orientation, and viewpoint can be handled in a consistent manner using a simple camera model. The MAPS database facility also maintains a partial model of interpretation, separate from, but in the same representation as, the map feature database.

The image processing component is based on the MACHINESEG program described in the previous section. It performs low level and intermediate level feature extraction. Processing primitives are based on linear feature extraction and region extraction using edge-based and region-growing techniques. It identifies islands of interest and extends those islands constrained by the geometric model provided by MAPS and model-based goals established by the rule-based component.

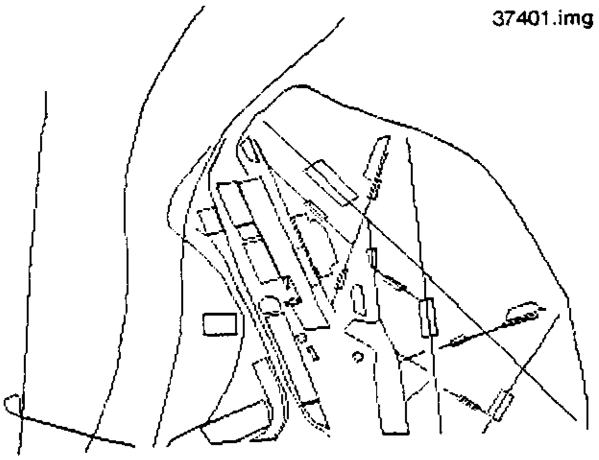
The rule-based component provides the image processing system with the best next task based on the strength/promise of expectations and with constraints from the image/map database system. It also guides the scene interpretation by generating successively more specific expectations based on image processing results.

We are in the preliminary stages of development for the SPAM system and have begun to build a detailed map model of National Airport. Figure 17 gives an example of the ability of the MAPS database to use image-to-map correspondence to generate unified spatial models from partial information. The line drawing labeled 37401.IMG contains the northern section of National Airport; 36809.IMG is a partially overlapping southern section of National Airport. Line segments represent point, line, and areal features corresponding to runways, terminal buildings, access roads, and hangars, interactively specified

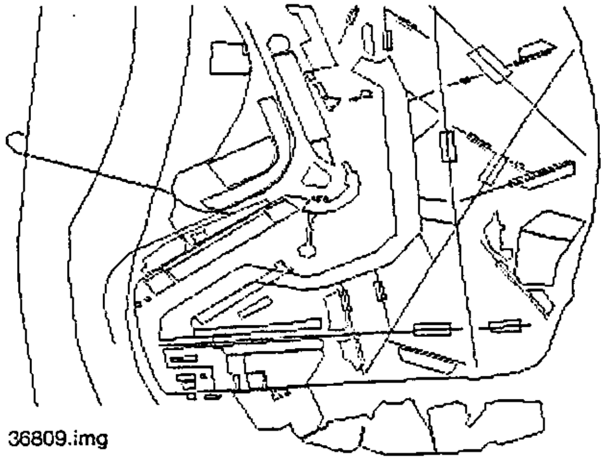
**Figure 17:** SPAM: National Airport Spatial Model



Unified scene of Washington National Airport



Northern section of Washington National Airport



Southern section of Washington National Airport

using the CONCEPTMAP representation. For those features that appear in both images, the concept *role* mechanism (see Section 6.5.2) is used to specify multiple *<latitude/longitude/elevation>* descriptions. A unified map description is created by matching corresponding line segments using the overlapping image areas (in map space) to constrain search. The result of unification is the line drawing labeled AIRPORT.IMG.

## 8. Future Work

Our future work will be directed toward two research topics. First, we have only begun to explore the use of MAPS as a component of an image interpretation system. We will continue our work in the airport scene interpretation task, using the SPAM system as a testbed for integration of a rule-based system with the MAPS system. Second, there is much to do in expanding the CONCEPTMAP database to include more complex 3D descriptions, and in attendant issues of scaling and sizing to larger databases. Other tasks we will pursue are the evaluation of our hierarchical spatial representation to constrain search in large databases, general solutions to complex spatial queries for situation assessment applications, and the application of spatial knowledge to navigate through a map database.

In discussing future work it is important to understand the strengths and limitations of the current research. The strengths of this work lie in several unique features of the MAPS system. First, we have constructed a system of moderate complexity which has significant capabilites in each area of our Image/Map Database model. The system integrates map knowledge from diverse sources and performs several tasks that require synthesis of this knowledge. We have the ability to represent complex map features in a uniform cartographic coordinate system and can compute new spatial relationships directly from the map data.

The major limitation in the MAPS system is the current method for performing image-to-map correspondence.[********] From the standpoint of the state of the art in photogrammetry, we make simplistic planemetric assumptions in our correspondence algorithm, but they do give reasonable results for several reasons. First, all of our photographs are vertical aerial mapping imagery, and efforts are taken to minimize camera tilt. Second, we have very high resolution photographs, each of which covers a relatively small area, and due to the relatively local level terrain in Washington D. C., our polynomial correspondence functions are reasonably accurate.

The issue is not how to recover camera information from the imagery, since in cartography and manual photo-interpretation the sensor models and ephemeral data are well known and modeled, but to use existing photogrammetric tools for basic data acquisition. Therefore, in this limitation we see an opportunity to investigate how MAPS could be interfaced to a photogrammetric frontend which would directly provide *<latitude/longitude/elevation>* data from a stereo model.[*********] The frontend should have a landmark database and interactive display tools to guide the stereo model setup in a manner similar to our current implementation. Nothing in the current MAPS implementation precludes such an interface since we maintain a 3D map feature representation throughout the database using the USGS terrain database. The building of such tools should be the common objective both to cartographers and to computer scientists.

## 9. Acknowledgements

The MAPS system is a result of effort by several people. Jerry Denlinger implemented the BROWSE system. Wilson Harvey implemented the hierarchical tree generation and supervised much of the landmark database generation. Mike Matsko implemented WASH3D, the 3D scene generation system. Dave Springer is responsible for image database generation and maintenance. Takeo Kanade and Raj Reddy have provided encouragement, support, and guidance for this work.

## References

1. McKeown, D. M., Jr. and Reddy, D. R., "A Hierarchical Symbolic Representation for an Image Database," *Proceedings of IEEE Workshop on Picture Data Description and Management*, April 1977, pp. 40-44.

2. McKeown, D. M., Jr., and Reddy, D. R., "The MIDAS Sensor Database and its Use in Performance Evaluation," *Proceedings ARPA Image Understanding Workshop*, Palo Alto, Cal., Oct. 1977, pp. 47-51.

3. Rubin, S. M., *The ARGOS Image Understanding System*, PhD dissertation, Carnegie-Mellon University, Computer Science Department, November 1978.

4. Rubin, S. M., "Natural Scene Recognition Using LOCUS Search," *Computer Graphics and Image Processing*, Vol. 13, 1980, pp. 298-333.

5. Barrow, H., et. al., "Interactive Aids for Cartography and Photo Interpretation: Progress Report, October 1977," *Proceedings: DARPA Image Understanding Workshop*, Oct. 1977, pp. 111-127.

6. Fischler, M., et. al., "The SRI Road Expert: An Overview," *Proceedings: DARPA Image Understanding Workshop*, Nov. 1978, pp. 13-19.

7. Quam, L., "Road Tracking and Anomaly Detection," *Proceedings: DARPA Image Understanding Workshop*, May 1978, pp. 51-55.

8. Nagy, G., and Wagle, S., "Geographic Data Processing," *Computing Surveys*, Vol. 11, No. 2, June 1979, pp. 139-182.

9. *IEEE Workshop on Picture Data Description and Management*, IEEE Computer Society, Asilomar, Ca., August 1980.

10. *IEEE Computer Magazine, Special Issue on Pictorial Information Systems*, IEEE Computer Society, November 1981.

11. Kondo, T., Shinoda, H. Sawada, N., Numagami, H. and Kidode, M., "A Map-Guided Image Database System For Remotely Sensed Data," *Proceedings of Pattern Recognition and Image Processing Conference*, June 1982.

12. Chang, N. S., and Fu, K. S., "A Query Language For Relational Image Database Systems," *Proceedings of the Workshop on Picture Data Description and Management*, IEEE, August 1980, pp. 68-73.

13. Chang, N. S., and Fu, K. S., "An Integrated Image Analysis and Image Database Management System," Tech. report TR-EE-80-20, Electrical Engineering Department, Purdue University, May 1980.

14. Chang, N. S., *Image Analysis and Image Database Management*, UMI Research Press, Ann Arbor, Michigan, 1981.

15. Samet, H., and Webber, R. E., "Using Quadtrees to Represent Polygonal Maps," *Proceedings of the IEEE Computer Vision and Pattern Recognition Conference*, June 1983, to appear

16. Rosenfeld, A., Samet, H., Shaffer, C., and Webber, R. E., "Application of Hierarchical Data Structures to Geographical Information Systems," Tech. report TR-1197, University of Maryland, 1983.

---

[********] For tasks such as photo-interpretation and situation assessment, the current level of accuracy using high resolution photographs is not as critical and may be sufficient.

[*********] The CAPIR system[33] is one example of such a system.

17. Matsuyama, T., Hao, L. V., and Nagao, M., "A File Organization for Geographic Information Systems," *Proceedings of 6ICPR*, IEEE, Oct 1982, pp. 83-87.

18. Bullock, B.L., et. al., "Image Understanding Application Project: Status Report," *Proceedings: DARPA Image Understanding Workshop*, Sept. 1982, pp. 29-41.

19. Brooks, R. A., "Symbolic Reasoning among 3-D Models and 2-D Images," *Artificial Intelligence*, Vol. 17, 1981, pp. 285-349.

20. Matsuyama, T., "A Structural Analysis of Complex Aerial Photographs," Tech. report, Department of Electrical Engineering, April 1980, Ph.D Thesis

21. Nagao, M., Matsuyama, T., and Mori, H., "Structural Analysis of Complex Aerial Photographs," *Proceedings of Sixth International Conference on Artificial Intelligence*, IJCAI, Tokyo, Japan, August 1979, pp. 610-616.

22. Selfridge, P. G., "Reasoning About Success and Failure in Aerial Image Understanding," Tech. report 103, University of Rochester, May 1982, Ph.D Thesis

23. Herman, M., Kanade, T. and Kuroe, S., "Incremental Acquisition of a Three-dimensional Scene Model from Images," Tech. report CMU-CS-82-139, Computer Science Department, Carnegie-Mellon University, October 1982.

24. McKeown, D. M. and T. Kanade, "Database Support for Automated Photo Interpretation," *Techniques and Applications of Image Understanding III*, Society of Photo-Optical Instrumentation Engineers, Washington, D.C., April 1981, pp. 192-198.

25. McKeown, D.M., "Concept Maps," *Proceedings: DARPA Image Understanding Workshop*, Sept. 1982, pp. 142-153, Also available as Technical Report CMU-CS-83-117

26. McKeown, D. M., and J. L. Denlinger, "Graphical Tools for Interactive Image Interpretation," *Computer Graphics*, Vol. 16, No. 3, July 1982, pp. 189-198.

27. Teitelman, W., "A Display Oriented Programmer's Assistant," Tech. report CSL77-3, Xerox Palo Alto Research Center, 1977.

28. Sproull, R. F., "Raster Graphics for Interactive Programming Environments," Tech. report CSL79-6, Xerox Palo Alto Research Center, 1979.

29. Defense Mapping Agency, *Product Specifications for Digital Landmass System (DLMS) Database*, St. Louis, Missouri, 1977.

30. Faintich, M. B., "Digital Sensor Simulation at the Defense Mapping Agency Aerospace Center," *Proceedings of the National Aerospace and Electronics Conference (NAECON)*, May 1979, pp. 1242-1246.

31. Faintich, M. B., "Sensor Image Simulator Application Studies," *Proceedings: International Conference on Simulators*, Sept. 1983, to appear

32. McKeown, D. M., and Matsko, M. S., "Urban Scene Generation for Cartographic Applications," Tech. report, Carnegie-Mellon University, in preparation, 1983.

33. Lukes, G. E., "Computer-assisted photo interpretation research at United States Army Engineer Topographic Laboratories (USAETL)," *Techniques and Applications of Image Understanding III*, Society of Photo-Optical Instrumentation Engineers, Washington, D.C., April 1981, pp. 85-94.

34. Yoram Yakimovsky, "Boundary and Object Detection in Real World Images," *Journal of the ACM*, Vol. 23, No. 4, 1976, .

35. McKeown, D.M., and McDermott, J., "Toward Expert Systems for Photo Interpretation," *IEEE Trends and Applications '83*, May 1983.

# MAPS System Major Components

This Appendix contains a list of the major program modules which compose the MAPS system.

| NAME | SIZE (bytes) | COMMENTS |
|---|---|---|
| Browse | | |
| browse | 305500 | interactive image display facility |
| picpac | 530762 | interactive image processing facility |
| Corres | | |
| corres | 286042 | interactive image-map correspondence |
| checkcorres | 49523 | check correspondence errors |
| cormain | 52893 | correspondence algorithm |
| corpairs | 75176 | edit correspondence pairs file |
| creatsdf | 50601 | create a scene description file |
| dumpcoef | 19649 | dump a coefficients file |
| dumpcor | 23547 | dump a correspondence file |
| dumpsdf | 25398 | dump a scene description file |
| hypcorpairs | 82380 | generate hypothesized landmarks |
| updatesdf | 59099 | update a scene description file |
| Landmark | | |
| landmark | 194953 | interactive landmark extraction |
| creatldm | 23557 | create binary landmark file |
| etytod3 | 50217 | make a .d3 file from an .ety file |
| etytoldm | 19948 | create landmark file from .ety files |
| ldescribe | 43695 | give landmark descriptions |
| ldmrlprt | 38275 | dump all info about a landmark |
| ldmtest | 28696 | find landmarks within geodetic area |
| Segment | | |
| segment | 170230 | hand segmentation program |
| mkidf | 10537 | create ascii file from binary seg file |
| segrename | 39045 | edit segmentation region names |
| Machineseg | | |
| machineseg | 290222 | machine segmentation program |
| Conceptmap | | |
| conceptmap | 665710 | associate conceptual and map data |
| buildsegmap | 98301 | build composite segmentations |
| coetrack | 125241 | track points using map correspondence |
| congeoall | 213278 | generate geometric database |
| d3dump | 24629 | dump a d3 file |
| d3entcor | 93936 | create corres entry from .d3 file |
| d3fdump | 31039 | dump a d3 feature file |
| d3tod3f | 15826 | convert a .d3 file to a feature file |
| d3toimg | 44710 | generate binary image from .d3 files |
| dlmsseg | 128324 | create DLMS overlay for geodetic area |
| dmaextract | 31544 | extract features from DLMS .fea files |
| dumpql | 207962 | dump a querylist file |
| dumpsdf | 25398 | dump a scene description file |
| ecdump | 9425 | dump the contents of a coverage file |
| ecshow | 137700 | display manager for coverage files |
| ecsort | 26624 | sort coverage files by keys |
| ectoseg | 18173 | create .seg file from coverage file |
| hierarchy | 486262 | build and access hierarchical database |
| hiertrack | 321869 | track and display pts using hierarchy |
| idhier | 254739 | identify points using hierarchy |
| imagetoec | 34283 | associate image with coverage file |
| imagetomap | 54092 | <generic><row><col> => <lat/lon/elev> |
| photo | 299710 | interactive image photogrammetry |
| segtod3 | 57034 | convert .seg file to .d3 data structure |
| segtoimg | 32785 | convert .seg regions to binary image |
| stereoshow | 153125 | show stereo image pairs |
| unifyseg | 107603 | unify segmentation regions |
| Wash3d | | |
| wash3d | 764517 | 3d scene generation from MAPS database |
| dfeaprt | 45013 | print DLMS feature given dlms code |
| dispfea | 137335 | display a DLMS map feature file |
| dlms | 53134 | create dlms index file |
| dlmsbin | 34604 | convert ascii feature files to binary |
| dlmsfind | 45419 | find a DLMS feature based on attributes |
| feadumper | 45267 | dump a DLMS feature file |
| Terrain | | |
| elevation | 24097 | access terrain data images |