ETL-0509

AD-A203 712

Vision-Based Navigation for Autonomous Ground Vehicles First Annual Report

Larry S. Davis

Center for Automation Research
University of Maryland
College Park, Maryland 20742-3411

July 1988

Prepared for:

Defense Advanced Research Projects Agency
1400 Wilson Boulevard
Arlington, Virginia 22209-2308

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

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

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

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

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1a. REPORT SECURITY CLASSIFICATION: UNCLASSIFIED	1b. RESTRICTIVE MARKINGS
2a. SECURITY CLASSIFICATION AUTHORITY	3. DISTRIBUTION/AVAILABILITY OF REPORT: Approved for Public Release; Distribution is Unlimited.
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE	
4. PERFORMING ORGANIZATION REPORT NUMBER(S)	5. MONITORING ORGANIZATION REPORT NUMBER(S): ETL-0509
6a. NAME OF PERFORMING ORGANIZATION: University of Maryland; 6b. OFFICE SYMBOL (If applicable)	7a. NAME OF MONITORING ORGANIZATION: U.S. Army Engineer Topographic Laboratories
6c. ADDRESS (City, State, and ZIP Code): Center for Automation Research, College Park, MD 20742-3411	7b. ADDRESS (City, State, and ZIP Code): Fort Belvoir, Virginia 22060-5546
8a. NAME OF FUNDING/SPONSORING ORGANIZATION: Defense Advanced Research Projects Agency; 8b. OFFICE SYMBOL (If applicable): ISTO	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER: DACA76-84-C-0004
8c. ADDRESS (City, State, and ZIP Code): 1400 Wilson Boulevard, Arlington, VA 22209-2308	10. SOURCE OF FUNDING NUMBERS: PROGRAM ELEMENT NO. 62301E; PROJECT NO.; TASK NO.; WORK UNIT ACCESSION NO

11. TITLE (Include Security Classification)
Vision-Based Navigation for Autonomous Ground Vehicles First Annual Report

12. PERSONAL AUTHOR(S)
Larry S. Davis

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT (Year, Month, Day)	15. PAGE COUNT
Annual	FROM 7/84 TO 7/85	1988, July	23

16. SUPPLEMENTARY NOTATION

17. COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB-GROUP	Autonomous Navigation, Road Following, Computer Vision

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

This is the first annual report for ETL contract DACA76-84-C-0004. Our activities on the project principally involved building an experimental facility for performing research in vision for autonomous navigation of ground vehicles and developing a computational framework for constructing visual navigation systems.

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT: ☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT. ☐ DTIC USERS	21. ABSTRACT SECURITY CLASSIFICATION: UNCLASSIFIED
22a. NAME OF RESPONSIBLE INDIVIDUAL: Linda H. Graff	22b. TELEPHONE (Include Area Code): (202) 355-2700; 22c. OFFICE SYMBOL: CEETL-RI

DD Form 1473, JUN 86 — Previous editions are obsolete. — SECURITY CLASSIFICATION OF THIS PAGE

UNCLASSIFIED

Preface

This is the first annual report for ETL contract DACA76-84-C-0004. Our activities on the project principally involved building an experimental facility for performing research in vision for autonomous navigation of ground vehicles and developing a computational framework for constructing visual navigation systems.

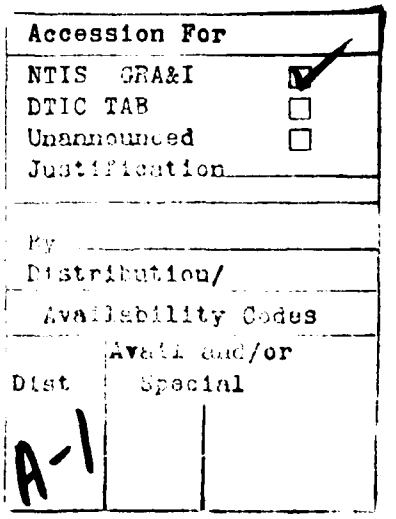

Table of Contents

1. Introduction and Executive Summary 1

2. Facility 3

3. The Maryland Visual Navigation System 5

3.1. System Architecture 7

3.2. System Modules 11

3.2.1. Image Processing 11

3.2.2. Visual Knowledge Base 12

3.2.3. Geometry 13

3.2.4. Vision Executive 14

3.2.5. 3-D Representation 14

3.2.6. Predictor 15

3.2.7. Sensor Control 15

3.2.8. Navigator 16

3.2.9. Pilot 17

3.2.10. Planner 17

4. Technical Reports 18

Figure 1 19

1. Introduction and Executive Summary

This is the annual report for ETL contract DACA76-84-C-0004 "Vision algorithms for autonomous navigation." This was the first contract year of a three year contract, and our activities principally involved building an experimental facility for performing research in vision for autonomous navigation of ground vehicles and developing a computational framework for constructing visual navigation systems.

In Section 2 of this report we describe the facility that we have constructed. The facility includes a variety of computers, robots and sensors that should allow us to develop, in the laboratory, algorithms for visual navigation that can be transferred to the Martin Marietta Corporation for integration onto the Autonomous Land Vehicle. The computers that we have acquired include a VAX 11/785 for general program development. The VAX was chosen because of previous experience that the Laboratory has had with the machine, and the large amount of vision software developed in the Laboratory to run on this machine. The Laboratory also acquired a VICOM image processing system, in exactly the same configuration as the system that will be used to drive the vehicle at Martin Marietta. The VICOM is a MC 68000 based system with special purpose hardware for image convolutions, image arithmetic and table look-ups to approximate arbitrary nonlinear image operations. Todd Kushner, a research scientist supported by the project, has extensive VICOM experience and during the first year of the program provided Martin Marietta with significant technical support in the use of the machine. The Laboratory had, previous to this project, an

American Robot Merlin robot arm which is being used in this project to simulate the ALV. We have purchased a lightweight solid state black and white TV camera and mounted it on the robot arm along with specially designed position encoders that allow us to control the height and the attitude of the camera with respect to the ALV terrain board. The terrain board that we designed and constructed, has nontrivial topography (which allows us to test various road reconstruction algorithms) and has a small set of roads painted on it. We have found this facility to be extremely valuable in designing our visual navigation system. To augment the data that can be acquired with the robot arm, we have assisted Martin Marietta in collecting data sets of road images from their test sites. It is on these images that we have designed and tested our low level vision algorithms.

In Section 3 we present the details of our proposed framework for visual navigation, along with some initial experimental results. This system that we have designed is a highly modular one, with components for image processing, road reconstruction, sensor control and path planning. The design of the system is motivated by a desire to effectively allocate scarce computational resources to the road recovery task. Thus, the system operates by focusing its attention on small windows of the road images that are predicted to contain road features needed both to verify the system's current model of the structure of the road on which it is traveling, as well as to extend that model in the direction of travel. These windows are chosen on the basis of a three dimensional road model that is incrementally constructed as the vehicle moves through the world. This world model, in conjunction with estimates of vehicle motion (available on the ALV

from its internal navigation system and on the robot from its controller), together with the ability to pan and tilt the image sensor, allow us to control the sensor and its processing to efficiently track the road through the world. One of the most important components of our vision system is the three dimensional road reconstruction module. In the absence of either stereo or motion information (which could not be obtained accurately enough on the actual ALV), one must resort to monocular methods for road reconstruction. The simplest of these methods, a flat earth reconstruction, yields accurate road models only to short distances in front of the vehicle. We have designed a sophisticated monocular road inverse perspective algorithm that, based on simple and natural assumptions about road geometry, is able to reconstruct the three dimensional road including road banks. All of these modules are described in detail in Section 3. inally, Section 4 contains a list of reports generated during the year.

2. Facility

The Computer Vision Laboratory has been working closely with Martin Marietta Corp., Denver Colorado, the integrating contractor on DARPA's autonomous land vehicle program. Maryland's role in this relationship is to develop basic algorithms for visual navigation and then to transfer these algorithms to Martin for possible use in demonstrations of the Autonomous Land Vehicle (ALV). In order to support this technology transfer, we have developed an experimental facility at Maryland composed of computer systems, a robot arm carrying TV and other sensors, and a terrain board. These are described briefly below.

During the first year of the program, Martin Marietta plans to navigate the ALV with an onboard VICOM image processing computer. This is a Motoral 68000 based system with special purpose hardware for image arithmetic, convolution, and table lookups. Maryland acquired a VICOM system identical in configuration to the one that will reside on the ALV. Furthermore, Maryland hired Dr. Todd Kushner as a senior programmer on the project; Dr. Kushner worked for VICOM and has unique expertise on much of the image processing software that runs on the VICOM. During the first year of the contract he provided valuable assistance to Martin Marietta in identifying both software and hardware problems that they encountered with the VICOM system.

Maryland also acquired a VAX 11/785 to support algorithm development in the laboratory. This system is similar to the Laboratory's existing VAX 11/785 and allows us to dedicate a powerful computer to the development of algorithms for visual navigation.

Early in the program, Maryland had considered building a small autonomous vehicle for experimentation. Based on both a cost analysis and the need for special engineering expertise for both designing and maintaining such a vehicle, we decided instead to develop an experimental facility in the Laboratory using an existing robot arm. This arm was manufactured by American Robot, has a large workspace and can carry quite substantial payloads. The robot currently carries a lightweight black and white solid state SONY television camera in a mount fitted with three position encoders. These position encoders allow us to control both the height and the attitude of the camera with respect to a terrain board.

We have constructed a terrain board with modest topography (mostly flat, but with a few small hills). On this board we painted a black road network against a green background. The road network consists of one oval road, with a straight road along the major axis of the oval. The straight road is wider than the oval. The TV camera on the robot arm first transmits imagery to the VICOM, and the VICOM can either process the imagery (to test code for delivery to Martin Marietta) or can then transmit the imagery to the VAX for processing by our experimental visual navigation system.

We have found this facility to be extremely useful in developing vision algorithms and our navigation system. While the imagery obtained from the terrain board is quite different quantitatively from the imagery obtained by the color sensor on the ALV, we can still test major components of our vision system—specifically the focus of attention mechanism and the road inverse perspective modules—completely with the terrain board. The image processing algorithms are developed on the more difficult images from the Martin Marietta test site, and then used on the simpler (although still non-trivial) images obtained on the terrain board.

3. The Maryland Visual Navigation System

Our objective is to endow a mobile robot vehicle with the intelligence required to sense and perceive that part of its surroundings necessary to support navigational tasks. The vehicle maintains continuous motion by alternatively "looking ahead" and then "driving blind" for a short distance before taking

another view. While moving blindly, the accepted monocular image is processed to extract features which are then interpreted in three dimensions by a combination of "shape from contour" and geometric reasoning. This provides the data required to form an object-centered representation in the form of a local map. This map is used both for navigation and for focusing attention on selected parts of the visual field as the vehicle continues to move, accepting new images for processing. Though our domain of application has been the visual navigation of roadways, we have attempted to derive some useful principles relevant to visual navigation systems in general.

We have found it useful to distinguish between two different modes of visual processing, and our system can switch between these modes when necessary. Generally, the system begins a task in the *bootstrap mode* which requires processing an entire scene, picking out the objects of interest such as roads or landmarks. Sometimes this mode of processing can be avoided if the system is provided with detailed data concerning the locations of such objects, either from a map or from the analysis of previous sensed data. Once objects of interest are identified, the system switches to a prediction-verification mode called *feed-forward* in which the location of an object as seen from a new vehicle position is estimated, thus focusing attention on a small part of the visual field. This feed-forward capability emerges from the interaction between vehicle dead-reckoning, 3-D world modeling and vision. It has been particularly useful for road following, leading to a computational saving of a factor of ten. However, it is a useful principle in general, and will be applicable to obstacle avoidance as well.

We have implemented our system as a set of concurrent modules running on a time-sharing system (Berkeley UNIX 4.3) with the modules communicating through UNIX system "sockets."

3.1. System Architecture

We have developed and implemented the modular system architecture shown in Figure 1. The architecture shown in Figure 1 consists of a Vision system along with modules for Planning, Navigating and Piloting. The Vision system is decomposed into modules which support low, intermediate and high-level vision.

The vision system operates in two modes, *bootstrap and feed-forward.* This distinction is possible because the system can exploit a "focus of attention" on a portion of the visual field deemed important. The bootstrap mode is employed when the vision system must establish its first view of an object, i.e., it must find it in the visual field. This often requires processing the entire image. (If vehicle position is accurately known and detailed map data is provided, the system can bypass the bootstrap mode and use the map to focus attention.) Once an object is localized, the vision system can predict (to some accuracy) the location of that object in the following view and so restrict future processing to a small portion of the visual field. The system is then operating in the feed-forward mode. This kind of processing, in which a focus of attention is maintained, is particularly important when computational resources are limited, which is really always the case. In fact, it is via the feed-forward mode of operation that the vehicle can achieve continuous movement over an obstacle-free road. By accepting an image

for feed-forward processing, the vehicle can move "blindly" while deriving a new 3-D model of the road from the image. During this travel, the vehicle is dead-reckoning its position relative to the previously derived 3-D model. Depending on the accuracy of the dead-reckoning system, one can extend the distance of blind travel and so achieve greater vehicle speeds. However, this will not be appropriate if obstacles can move into the vehicle's path during the blind travel time. Alternatively, fast, dedicated hardware can be utilized to speed up the computations while keeping the blind travel distance to a minimum.

We shall describe the system architecture in Figure 1 by first explaining the responsibilities of the individual modules in terms of how they transform the data or representations across the interfaces. Then we describe the flow of control through the system in support of the road following task.

The vision modules in our system support low-level (Image Processing), intermediate-level (Geometry) and high-level (Knowledge-Based) vision. These modules are coordinated by a Vision Executive module which also has access to 3-D Representation, Scene Prediction and Sensor Control modules. The Vision Executive also supports interfaces to the Planner and Navigator modules. The Pilot module communicates only with the Navigator. These modules' functional capabilities are listed inside each block in the diagram, and are described in detail in the next subsection.

The vision system as a whole is responsible for perceiving objects of interest (e.g., roads and landmarks) and representing them in an "object centered" reference frame. The Image Processing module is responsible for extracting symbolic

representations from the individual images. These image domain symbolics correspond to significant events in the signal data; they are general features which describe images (e.g., edges, lines, blobs). The transformation from TV signals to symbols represents an enormous reduction in data. Extraction of symbols can be performed either on the entire image, or within a specified window. The module we call the Visual Knowledge Base has several responsibilities. Given the image domain symbolics extracted by the Image Processing module, the Visual Knowledge Base module tries to establish significant groupings of these symbols (e.g., pencils of lines). These groupings are global, corresponding to spatial organizations over large parts of the image, in contrast to the symbols themselves which are typically local groupings of events. This grouping process will then discard those symbols which are not found to belong to any group. The Visual Knowledge Base is also responsible for establishing meaningful groupings from 3-D representations provided by the Geometry module. Given the 3-D data, the Visual Knowledge Base module tries to recognize specific kinds of objects (e.g., roads), and so label important parts of the scene. The Geometry module is responsible for 3-D shape recovery, converting the grouped symbolics (obtained earlier) into surface patches described in a viewer centered reference frame. The Vision Executive is the heart of the vision system; it maintains the "flow of control" through this part of the system, trying to meet the "attentive goals" (such as *find road* or *find landmark* or *find obstacles*) provided by the Planner and Navigator. It is this Executive which triggers the mode of operation (bootstrap or feed-forward). The Vision Executive is aided by several additional sub-

modules which are also shown in Figure 1. Once a 3-D model of the scene has been established in the viewer centered coordinate system, it is converted to an object centered representation by the 3-D Representation module. This representation is more compact than the viewer centered description, and corresponds to a world model organized around the static components of the scene which do, in fact, dominate the scene. In the case of roads, this is the representation passed to the Navigator module for planning a path. This 3-D representation is also used by the Scene Predictor to focus attention on small areas of the visual field in which important objects are located, even after the vehicle has traveled blind; it is the foundation upon which the feed-forward mode operates. Finally, the Vision Executive can control the pointing of the camera via a Sensor Control module. Thus, in seeking to find a landmark or road, for example, the Executive establishes the visual field.

Three additional modules comprise our architecture as it currently stands. A Planner module is responsible for establishing the overall goals of the system and assigning priorities to these goals. As we are concerned with navigation, these goals are typically location goals, either in a map, or relative to something like a landmark located in a map. For road following a goal may be "move to point N on the road map." It is also appropriate that the Planner be responsible for overall resource allocation as this is where the "time sensitivity" of the system resides. Hence, priorities can be established and altered when deemed necessary. The Navigator module is a special purpose planning module. It is responsible for generalized path planning. The Navigator must also track the position of

the vehicle through the 3-D representation as it moves blindly, using "travel" data from the Pilot. Once the Navigator establishes a particular path for some short distance, it passes the path to the Pilot module which interprets this path into steering and motion commands for the vehicle. The Pilot is also responsible for monitoring the dead reckoning (and inertial navigation) unit aboard the vehicle. It should convert wheel shaft encoder readings and gyroscope headings into directional travel since the last time "travel" was requested by the Navigator.

3.2. System Modules

We now consider each module separately.

3.2.1. Image Processing

We have developed algorithms for the extraction of dominant linear features from entire (gray level) images as well as gray and color segmentation routines. These analyses provide independent representations of the information contained in the images in the form of *boundary based* and *region based* descriptions, respectively. These routines are relevant to the bootstrap mode of processing described earlier. Related versions were also developed to support the feed-forward mode. The capabilities we have already installed combine low-level operations with certain grouping procedures to derive particular symbolic descriptors. Thus, our linear feature detector relies on the grouping of pixels with similar gradient orientation, yielding a linear feature with global support (and poor localization) in the bootstrap mode and one with local support (and good localization) in the feed-forward mode. The segmentation procedures are based on edge-

preserving smoothing followed by a connected-components analysis. The color version of the segmentation procedure is far less sensitive to parameters than is the gray version. (It also provides the additional measure of "color" for each segment.) Linear feature extraction is completely automatic. Since the knowledge-based reasoning module in our system cannot currently fuse multiple sources of data (such as both linear features and regions), we utilize only the linear features when running the entire system.

3.2.2. Visual Knowledge Base

This module implements two separate vision tasks–seeking significant groupings of symbols derived from an image and checking consistency of 3-D shape recovery with generic models of objects (e.g., roads). Many types of symbolic groupings can be considered in general, though for purposes of road following we concentrated on the grouping of linear features into pencils of lines.

Pencils are determined by spatial clustering of intersections between pairs of lines in order to suggest a vanishing point. A sequence of vanishing points are adopted when grouping lines from the bottom of an image to the top (corresponding to "near to far" in the world). In the context of road following, these groupings represent hypotheses about road boundaries and markings, and the road geometry itself when more than one grouping is found (for example turns and changes in ground slope).

Once pencils have been grouped and assigned a 3-D interpretation by the Geometry module (see below), the Knowledge Base module attempts to reason

about the consistency of the successive surface patches that comprise the hypothetical road. Changes in surface slope and 3-D symmetries of the road are typical attributes that are considered. If the surface patches and corresponding road segments satisfy the constraints, the interpretation of a "road and its parts" are assigned and associated with a scene model.

3.2.3. Geometry

The Geometry module converts the grouped symbolics in the image domain to a viewer centered 3-D description of objects in the scene. A variety of "shape from" techniques have been suggested for accomplishing this in general (as listed in the diagram), leading to a representation termed the "2.5-D sketch". When several methods are employed, their results must be combined in a kind of "integrated 2.5-D sketch." Currently, our system utilizes several methods for recovering shape from monocular imagery; these are essentially "shape from contour." The first method employs a flat earth assumption, and simply involves backprojecting the images of the road boundary on the ground plane (determined from the vehicle's land navigation system).

Our second method of shape recovery is really model driven. We can invert the perspective transformation of the imaging process if we adopt the following three (road) model-based assumptions:

1. Pencils in the image domain correspond to planar parallels in the world.

2. Continuity in the image domain implies continuity in the world.

3. The camera sits above the first visible ground plane (at the bottom of the image).

Our 3-D reconstruction will then characterize a road of constant width, with turns and changes in slope and bank. The reconstruction process amounts to an integration out from beneath the vehicle into the distance, along local parallels over topography modeled as a sequence of planar surface patches. A simplified version of the method was used by Martin Marietta in their May 1985 demonstration.

3.2.4. Vision Executive

This module does not really perform any computations as yet. It maintains the flow of control through the system in order to exhibit a behavior, such as road following. As the contents of the Knowledge Base are expanded to incorporate other object models, the Vision Executive will have to make decisions about what scene features to search for, which sensors to use in those searches, etc. Once the system modules are mapped to individual nodes of a multiprocessor such as the Butterfly, the Vision Executive will be required to buffer messages between modules as the computations are asynchronous.

3.2.5. 3-D Representation

This module converts the 3-D viewer centered representation of the road scene into an object-centered description. The description is in the form of a file which lists a set of road attributes at each "roadmark" set down. Roadmarks are placed (by this module) along the centerline of the reconstructed road model, at

the beginning of each new planar patch.

3.2.6. Predictor

This module is used to focus the attention of the system on a small part of the visual field. Windows in the field of view are determined inside of which the near part of the road boundaries are located, following the vehicle's blind travel through the 3-D representation created from an earlier image. This is accomplished by essentially transforming the 3-D data used to create the most recent representation according to a rigid body translation and rotation associated with the vehicle's motion (as is familiar in computer graphics). The components of travel used in constructing this transformation are obtained from the dead reckoning (and inertial navigation) system aboard the vehicle. Once the 3-D representation is transformed accordingly, we solve for the intersection of the road boundaries with the periphery of the field of view. Windows of @5 sup {roman o} times 5 sup {roman o}@ are then placed over these points at the lower part of the visual field.

3.2.7. Sensor Control

This module is essentially an interface to the device driver associated with the pan and tilt mechanism of the camera. Computations consist of conversions between relative and absolute pointing angles.

3.2.8. Navigator

This module must provide computational capabilities in support of generalized path planning. Thus, it must be able to establish vehicle position from known landmarks sighted, generate region graphs from visibility/traversability data, generate region sequences between current position and goal, and plan paths down corridors of free space while avoiding obstacles. Our current Navigator is specific to following obstacle free roads (corridors) and so path planning consists of smoothly changing the heading of the vehicle in accordance with the 3-D representation derived from the visual process. This is accomplished by computing cubic arcs as asymptotic paths. Given the current position and orientation of the vehicle, a cubic arc is derived which begins at the vehicle, slopes in the direction of vehicle heading, and terminates 13 meters along the road centerline, in the direction of the centerline. This cubic arc is used as a path for the next 3 meters (about one vehicle length) at which point a new cubic arc is derived which terminates another 13 meters ahead. That is, the vehicle is always steered towards a point which is 13 meters ahead of it, along the centerline. Thus, the vehicle's path will be asymptotic to the road centerline. (If the road has more than one lane, we displace the centerline to the middle of a lane.) This yields paths with smooth changes in heading. When obstacles are introduced onto the road, we must adjust the termination point of a cubic arc so that no part of the arc intersects an obstacle nor crosses the road boundary.

In order to support the focus of attention mechanism in the feed-forward mode, the Navigator must track vehicle position through the 3-D representation

in response to dead reckoning data provided by the Pilot. This information is made available to the Vision Executive at its request.

3.2.9. Pilot

This module converts the cubic arcs obtained from the Navigator into a sequence of conventional steering commands used over the next 3 meters. They decompose a curved path into a set of short, straight line segments. These motion commands must then interface to the motor controls of the vehicle. (In our case, the interface is to the motion control software of a robot arm, as described in the next section.) The Pilot is also responsible for sending dead reckoning data from the vehicle to the Navigator. The Pilot converts the raw data into measured travel, and returns this information to the Navigator several times per second.

3.2.10. Planner

Our current Planner is quite simple; it merely specifies a distance goal, e.g. move to the point 60 meters further down the road. A mature planning module could have arbitrary complexity, specifying high level navigation goals, assigning priorities to these goals, monitoring progress as a function of time and constructing contingency plans. It would also be responsible for allocating computational resources throughout the system. However, until the vehicle can exhibit a variety of behaviors, it seems rather premature to concentrate on issues of high level planning.

4. Technical Reports

Three technical reports were produced during the first year of the contract.

1) Image Processing for Visual Navigation of Roadways by Jacequeline Le Moigne, Allen M. Waxman, Babu Srinivasan and Matti Pietikainen, University of Maryland Center for Automation Research Technical Report No. 138, July 1985.

2) A Visual Navigation System for Autonomous Land Vehicles by Allen M. Waxman, Jacqueline Le Moigne, Larry S. Davis, Eli Liang and Tharakesh Siddalingaiah, University of Maryland Center for Automation Research Technical Report No. 139, July 1985.

3) Road Boundary Detection for Autonomous Vehicle Navigation by Larry S. Davis and Todd Kushner, University of Maryland Center for Automation Research Technical Report No. 140, July 1985.

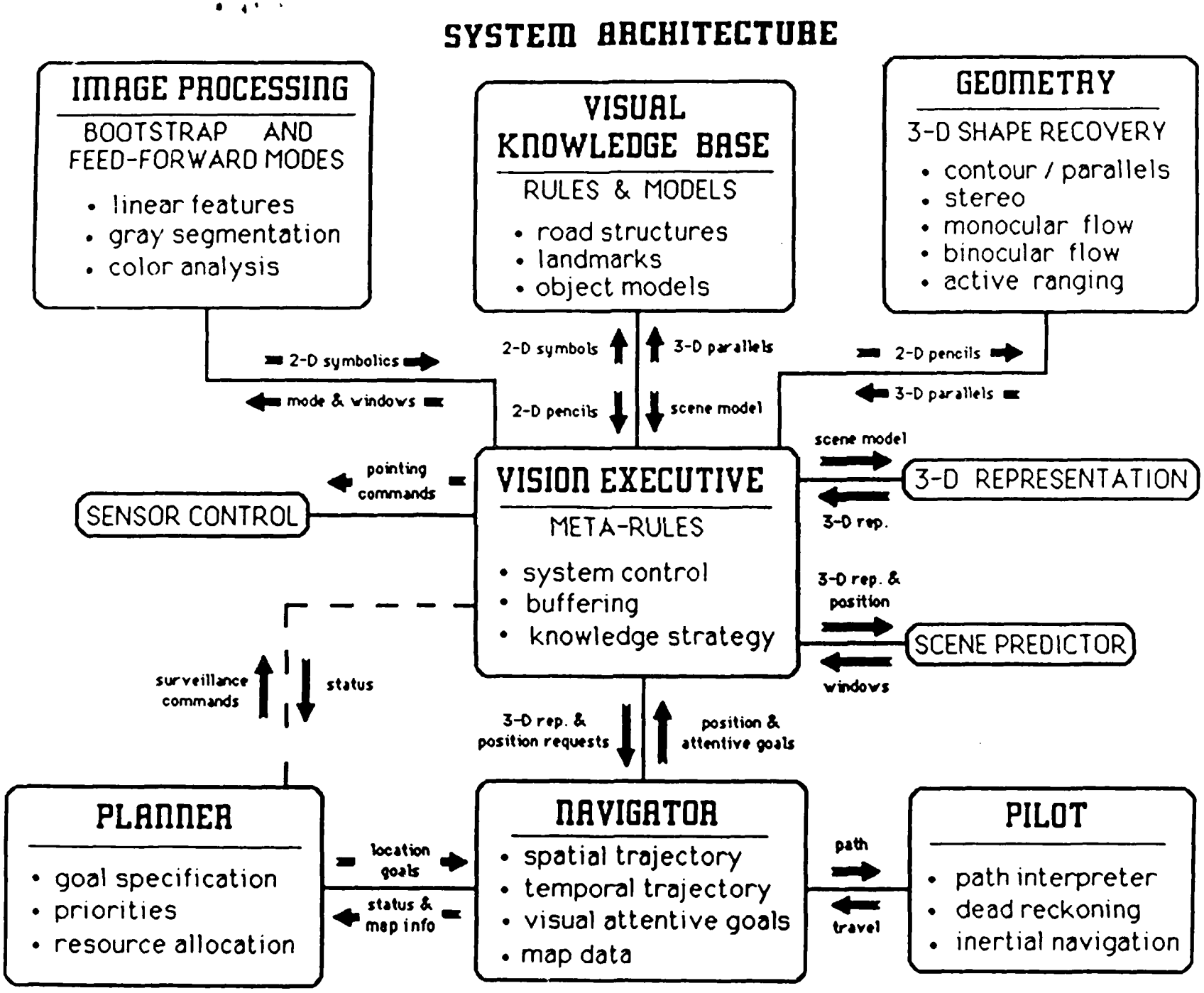

FIGURE 1. SYSTEM ARCHITECTURE AND ROAD FOLLOWING PROCESS CONTROL