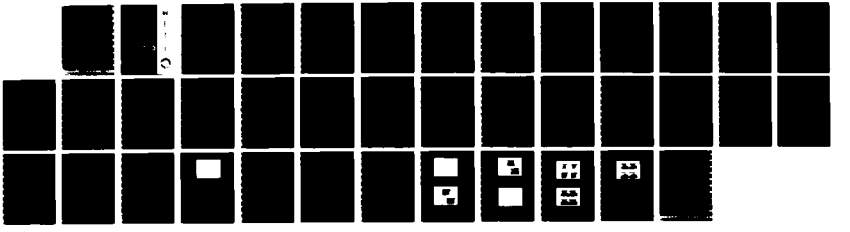
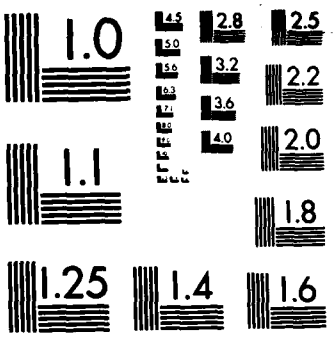
AD-A174 786 DOMAIN-DEPENDENT REASONING FOR VISUAL NAVIGATION OF ROADWAYS(U) MARYLAND UNIV COLLEGE PARK CENTER FOR AUTOMATION RESEARCH J LEMOIGNE OCT 86 CAR-TR-230 ETL-0445 DACA76-84-C-0004 F/G 17/7 NL

1/1

UNCLASSIFIED





ETL-0445

(5)

AD-A174 786

Domain-dependent reasoning for visual navigation of roadways

Jacqueline LeMoigne

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

October 1986

Prepared for
U.S. ARMY CORP OF ENGINEERS
ENGINEER TOPOGRAPHIC LABORATORIES
FORT BELVOIR, VIRGINIA 22060-5546

and
DEFENSE ADVANCED RESEARCH PROJECTS AGENCY
1400 WILSON BOULEVARD
ARLINGTON, VIRGINIA 22209-2308

E T L

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

ADA 174786

REPORT DOCUMENTATION PAGE

Form Approved
OMB No 0704-0188
Exp Date Jun 30, 1986

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) CAR-TR-230 CS-TR-1721	5. MONITORING ORGANIZATION REPORT NUMBER(S) ETL-0445
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	7b. ADDRESS (City, State, and ZIP Code) Research Institute Fort Belvoir, VA 22060-5546
8a. NAME OF FUNDING/SPONSORING ORGANIZATION DARPA 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) Domain-Dependent Reasoning for Visual Navigation of Roadways	
12. PERSONAL AUTHOR(S) Jacqueline LeMoigne	
13a. TYPE OF REPORT Technical 13b. TIME COVERED FROM _____ TO _____	14. DATE OF REPORT (Year, Month, Day) 1986, October 15. PAGE COUNT 37
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
17	07		Road following
17	08		Computer vision

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

A Visual Navigation System for Autonomous Land Vehicles has been designed at the Computer Vision Laboratory of the University of Maryland. This system includes several modules, among them a "Knowledge-Based Reasoning Module" that is described in this report. This module utilizes domain-dependent knowledge (in this case, "road knowledge") in order to analyze and label the visual features extracted from the imagery by the Image Processing Module. Knowledge and general hypotheses are given in section 2. The Reasoning Module itself is described in section 3, and results are presented in section 4. Finally, some conclusions and future extensions are proposed in section 5.

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT ☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT. ☐ DTIC USERS	21. ABSTRACT SECURITY CLASSIFICATION Unclassified
22a. NAME OF RESPONSIBLE INDIVIDUAL Rosalene M. Holecheck	22b. TELEPHONE (Include Area Code) (202) 355-2767 22c. OFFICE SYMBOL ETL-RI-T

DD FORM 1473, 84 MAR 83 APR edition may be used until exhausted. All other editions are obsolete.

SECURITY CLASSIFICATION OF THIS PAGE
UNCLASSIFIED

PREFACE

This document was prepared by the Center for Automation Research at the University of Maryland, College Park, Maryland, under contract number DACA76-84-C-0004 for the U.S. Army Engineer Topographic Laboratories, Fort Belvoir, Virginia, and the Defense Advanced Research Projects Agency, Arlington, Virginia. The Contracting Officer's Representative was Ms. Rosalene M. Holecheck.

CAR-TR-230
CS-TR-1721

DACA76-84-C-0004
October 1986

DOMAIN-DEPENDENT REASONING FOR VISUAL NAVIGATION OF ROADWAYS

Jacqueline Le Moigne

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

ABSTRACT

A Visual Navigation System for Autonomous Land Vehicles has been designed at the Computer Vision Laboratory of the University of Maryland. This system includes several modules, among them a "Knowledge-based Reasoning Module" that is described in this report. This module utilizes domain-dependent knowledge (in this case, "road knowledge") in order to analyze and label the visual features extracted from the imagery by the Image Processing Module. Knowledge and general hypotheses are given in Section 2. The Reasoning Module itself is described in Section 3 and results are presented in Section 4. Finally, some conclusions and future extensions are proposed in Section 5.

The support of the Defense Advanced Research Projects Agency as well as the U.S. Army Engineer Topographic Laboratories under Contract DACA76-84-C-0004 (DARPA Order 5098) is gratefully acknowledged.

1. INTRODUCTION

As described in [1] and [2], a visual navigation system for autonomous land vehicles has been designed at the Computer Vision Laboratory of the University of Maryland. This system, whose architecture is shown in Figure 1, includes several Vision Modules along with Planner, Navigator and Pilot Modules. The Vision Modules are responsible for recognizing objects of interest and constructing an interpretation of the scene. The *Vision Executive* is responsible for the overall vision process control. It is this Module which controls the flow of information between the Vision Modules, selects the mode of operation (bootstrap or feed-forward) and schedules all activities in the vision part of the system. The *Image Processing Module* [2] provides different symbolic descriptions of the images, corresponding to different sets of features (e.g., edges, lines, regions). This extraction of symbols can be performed either on the entire image (*bootstrap mode*) or within a specified window (*feed-forward mode*). These image symbols are then analyzed and interpreted as objects of interest. This analysis is performed, under control of the Vision Executive, by the *Visual Knowledge Base Reasoning Module* which simultaneously utilizes these 2-D symbols and their 3-D representations provided by the *Geometry Module.* The navigation system is described in [1] and details of the Image Processing Module are given in [2]. This paper describes in more detail the Knowledge Base Reasoning Module, as applied to the road following task.

2. DOMAIN-DEPENDENT KNOWLEDGE

When an image is acquired by the Image Processing Module, one of the two modes of operation (bootstrap or feed-forward) is chosen by the Vision Executive, as well as the type of image processing procedure (i.e., linear feature extraction, thresholding, etc.) to be applied. This report will only describe the interpretation and labeling of the linear features extracted from the images.

The Knowledge Base Reasoning Module utilizes domain-dependent knowledge to reason about these extracted features. In both modes, bootstrap and feed-forward, the reasoning is performed from the bottom of the image to the top, corresponding to near to far in the world. The knowledge-based reasoning has two responsibilities: identifying significant groupings of image symbols, and checking the consistency of 3-D shape recovery with models of the objects of interest (roads in this case). This section describes the knowledge utilized for these two different tasks.

2.1. Finding the significant groupings of image symbols

For the road following task, linear features are grouped into pencils of lines. A *pencil* is defined as a set of at least two line segments which converge in the image from bottom to top. A special case of a pencil is a *set of parallels* which might correspond to a road perpendicular to the line of sight. The main assumption which leads us to this choice of symbolic groupings is that many road images can be decomposed into several pieces, where each piece of the road is represented by a pencil which converges from bottom to top in the image. Figure

2 shows some examples of ideal road scenes represented by pencils. In particular, Figure 2(a) represents an ideal straight road with one pencil, and Figure 2(c) shows an intersecting road with a set of parallels.

For purposes of road following, these groupings are computed by utilizing assumptions about road boundaries and markings, and the road geometry. Several sets of assumptions are currently used.

The first set of assumptions concerns the road geometry. Some of these assumptions are used for grouping the line segments into converging pencils and choosing the successive pencils. A pencil is constructed by determining its vanishing point, based on the spatial clustering of intersections between pairs of image lines. The clustering algorithm is very simple.

(1) We consider all pairs of intersections. The first sufficiently close pair determines the first pencil; its vanishing point is the average location of the two intersections.

(2) For every other pair of intersections, the distance between the two intersection points is computed. If this distance is below a given threshold, three different cases may occur:

    (i) Neither of the two intersections already belongs to a pencil; a new pencil is created.

    (ii) One of the two intersections already belongs to a pencil; the other intersection is added to this pencil.

    (iii) The two intersections belong to two different pencils;

the two pencils are merged together.

(3) For each intersection not included in a cluster (i.e., too far from all other intersections), a pencil is created containing only that intersection.

(4) Create all "degenerate" pencils containing only one segment.

To summarize, a pencil may contain one line, two lines or a maximum number of lines corresponding to the same *intersection cluster.*

The second set of assumptions concerns the location of the vehicle relative to the road; in particular, for the Bootstrap mode, it is assumed that the vehicle does not start in the middle of a curve or an intersection and that the camera (and the vehicle) are pointing approximately towards the road. If the vehicle is off the road, the distance between the vehicle and the road is assumed to be small. This set of assumptions supports the incremental interpretation of the pencils from the bottom of the image to the top and guides the choice of the first pencil at the bottom of the image; it will be the pencil whose elements lie in the lower portion of the picture and whose general direction is the "closest to the vertical direction". This set of assumptions could be less restrictive if other pieces of evidence, obtained from complementary descriptions, could be combined with this first boundary-based description of the image. Other descriptions can be computed from different types of Image Processing, either on the same image or on different views of the same scene. The different types of image descriptions which may be obtained will be discussed in Section 5. Through control of the camera by the Vision Executive, different views from a single point of the same scene can

be obtained and processed: one could utilize such a *panoramic view* to accumulate evidence relevant to the choice and the labeling of the different image features. For example, assuming that the vehicle starts in the middle of a curve and that the camera is pointing straight ahead, only some of the significant road segments are visible in the current image; by controlling the pan of the camera, one could search for the road around these first features which were found in the initial view of the scene.

In the bootstrap mode, the first pencil in the image is chosen based on the assumptions described above, while the successive pencils are chosen by minimizing a function depending both on the distance to the previous pencil and on the consistency in direction with this previous pencil. For example, in Figure 2(b), the successively chosen pencils are (1,2,3,4,5,6) and then (7,8,9,10,11,12). In the feed-forward mode, the choice of the successive pencils is simplified by the image processing itself; all the line segments are computed successively on each side of the road and are given with their order to the Reasoning Module. In this case, the choice of the successive pencils follows this order. For example, in Figure 3, assuming that the line segments are exactly in symmetric correspondence on each side of the road, the successively chosen pencils are (1,2), (3,4), (5,6), etc.

2.2. Labeling the image symbols

Other assumptions about the road geometry are utilized for checking the consistency between the 3-D shape recovery and the model of the road.

We first describe our implementation of a monocular inverse perspective algorithm for reconstructing the three dimensional geometry of the road, and then describe how that three dimensional description is interpreted in the context of both generic knowledge about road structure and specific knowledge about the road being followed (such specific knowledge may be derived from either a map or analysis of previous images of the road). The inverse perspective technique [1] is based on the following three 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).

This technique builds a 3-D model of the road which includes turns, slopes and banks. Details of this Module are given in [1]. It returns to the Visual Knowledge Base the equation of the plane defined by the given pencil and the 3-D coordinates of all the line segments which form this pencil. For the first road patch, the 3-D reconstruction utilizes assumption (3); the two camera parameters of height and tilt determine the 3-D coordinates of the given line segments. For the following road patches, we utilize assumptions (1) and (2). By the first assumption, the 2-D and 3-D coordinates of the vanishing point lead to a single constraint on the parameters of the next patch. By the second assumption (*continuity assumption*), the 2-D and 3-D coordinates of two *continuity points*, which belong to the previous patch and must also belong to the new one, give us two

additional constraints. These three constraints allow us to determine the three parameters of the surface plane. Figure 4 shows an example where continuity and vanishing points are indicated. In order to utilize assumption (2), the two continuity points must lie on a line in the plane of the road and perpendicular to the direction of the road; Figures 5(a) and 5(b) represent such an example of 2-D segments with their 3-D reconstructions. The continuity points are usually end-points of two segments of the previous patch. However, if the two end-points of the previous segments do not satisfy this property, the longest 3-D segment must be *cut* and the continuity point considered for the next patch is the end-point defined by this cut. For example, in Figure 5(c), 3-D segments (MN) and (OP) have been reconstructed from the 2-D segments (mn) and (op), but the points N and P do not lie on a line (L) perpendicular to the road direction; therefore segment OP, which is the longest, is cut and the continuity points which will be considered for the next patch are the points N and Q (whose 2-D corresponding points are n and q).

Given this 3-D reconstruction, the system next reasons about the consistency of the successive surface patches that comprise the hypothetical road. The typical attributes which are considered are;

(1) Changes in surface slope between successive surface patches.

(2) Width of the road which must be included in an "acceptable" interval.

(3) Symmetries between *couples* (see below) of segments that define the locations of lane markers.

This reasoning process is described in more detail in Section 3. Below we discuss the grouping of linear segments into *couples.* Two kinds of couples may be defined;

(1) *road-shoulder* couples

(2) *lane marker* couples

For example, in Figure 2(a), we would define three couples:

(1,2) represents the left road-shoulder couple

(3,4) represents the center line couple

(5,6) represents the right road-shoulder couple.

These couples are determined by computing 3-D distances between segments and grouping together the segments with distances smaller than the minimum arbitrary widths for shoulders or lane markers (such information is given a priori).

It may happen that some segments are *isolated*—i.e., the distances to their neighbors are above some minimum arbitrary width. In this case, we create a *degenerate couple* which contains one *actual* segment (x) and one *virtual missing segment.* We will denote such couples by (x,–1) or (–1,x), where –1 represents the virtual missing segment. In the example of Figure 6, three couples are built, including a *degenerate* one; they are (1,–1), (2,3) and (4,5). Decomposing pencils into couples simplifies the interpretation process; whenever a couple contains one or two actual elements, all computations of distances and symmetries are performed directly on the couples and the couples are first interpreted as road-shoulder couples or lane marker couples following these measurements. If one

element in a couple is missing, the single line segment which represents the couple is initially assumed to be:

- the border of the shoulder in the case "road-shoulder" (cf. Figure 7),
- the midline of the lane marker in the case "lane marker" (cf. Figure 7).

The notion of *missing element* in a couple is particularly useful when this element appears in the next patch; it can be integrated into the labeling without rebuilding the complete model of the scene. Figures 8(a) and 8(b) show an example of two successive frames in which three and four line segments, respectively, have been found. The right border of the road is not found in the first frame (Figure 8(a)); (1,2) represents the left road-shoulder couple and (–1,3) represents the right road-shoulder couple, where the right border of the road is a missing element and segment 3 represents the right shoulder of the road. In the second frame (Figure 8(b)), (4,5) represents the left road-shoulder couple and is connected to the couple (1,2); (6,7) represents the right road-shoulder couple and is connected to the couple (–1,3); segment 6 represents the right border of the road which appears in this frame and segment 7 is the continuation of the right shoulder.

3. THREE-DIMENSIONAL ANALYSIS OF ROAD SCENES

This section contains a detailed description of the algorithm utilized in the Reasoning Module of our system.

The algorithm can be divided into five main tasks:

(1) Choice of the next *best* pencil.

(2) Checking the consistency of the new pencil.

(3) 3-D interpretation and labeling of this pencil.

(4) Finding *missing* segments

(5) Computation of the temporary scene model.

A description of each of the five different tasks is given below.

(1) Choice of the next best pencil

Assuming that all the pencils have been computed by the method defined in Section 2.1, the choice of the initial pencil depends on the distance to the bottom of the image and the verticality of the pencil in the image, which are computed as follows.

(1) The distance of a pencil to the bottom of the image is given by the height of its center of gravity above the bottom border of the image.

(2) The verticality of a pencil is the angle between the orientation of the pencil and the "vertical image direction". If V is the pencil's vanishing point and G is its center of gravity , then the orientation of the pencil is the

vector $\vec{VG}$.

If there is a pencil whose distance to the bottom of the image is significantly smaller than all the others, this pencil is chosen as the initial pencil. Otherwise, if no pencil is obviously the lowest in the image, we choose as the initial pencil the most vertical of the sufficiently low pencils.

For the subsequent pencils, each segment of the candidate pencil is associated with one segment of the previous pencil. Thus, the choice function computes:

(1) The distance between the previous pencil and a candidate pencil, that is, the sum of all the minimum 2-D distances between associated segments.

(2) The consistency of the candidate pencil with the previous one, which minimizes the proportionality of the 2-D distances between every two segments of the previous pencil and the same distances in the candidate pencil.

The *choice function* is the sum of these two measurements. Then, we choose as the next pencil, the pencil which minimizes this function.

For example, in Figure 9 the first pencil P_0 is constructed with the segments (1,2,3,4). Consider some of the candidate pencils for the next pencil; if P_1=(5,6,7) is a candidate pencil, the closest corresponding segments of the two pencils are (5,1), (6,2) and (7,3). The distance D_1 from P_1 to P_0 is

$$D_1 = d(1,5) + d(2,6) + d(3,7)$$

and the consistency of P_1 relative to P_0 is measured by

$$C_1 = \max\left[\frac{d(1,2)}{d(5,6)}, \frac{d(1,3)}{d(5,7)}, \frac{d(2,3)}{d(6,7)}\right].$$

Similarly, if P_2=(6,7,8,9) is another candidate pencil, the closest corresponding segments are (6,1), (7,2), (8,3) and (9,4),

$$D_2 = d(1,6) + d(2,7) + d(3,8) + d(4,9)$$

$$C_2 = \max\left[\frac{d(1,2)}{d(6,7)}, \frac{d(1,3)}{d(6,8)}, \frac{d(1,4)}{d(6,9)}, \frac{d(2,3)}{d(7,8)}, \frac{d(2,4)}{d(7,9)}, \frac{d(3,4)}{d(8,9)}\right].$$

Furthermore a constant A is added to the choice function for each line segment in either one of the two pencils which has no corresponding segment in the other pencil. For the two pencils P_1 and P_2, the choice functions are;

$$F_1 = D_1 + C_1 + A$$

$$F_2 = D_2 + C_2$$

Finally, $F_2 = min(F_1, F_2)$ and the pencil P_2 is chosen.

(2) Checking the consistency of a new pencil

If the pencil is the first one processed, no consistency is checked. Otherwise, the consistency measure is computed utilizing the two *extreme* segments of the candidate pencil. When a pencil is computed, all the segments are ordered inside the pencil from right to left "looking from the vanishing point". The first and last segments are called *extreme segments* in this ordering. The 3-D interpretation of these two extreme segments is computed and the difference in angle between the previous and the new plane furnishes the consistency measure.

The inconsistency may occur, for example, when one of the extreme segments corresponds to the border of a line of bushes along the road instead of the

border of the shoulder or the border of the road. Another example is the one shown in Figure 10 where one of the extreme segments happens to be the horizon line.

If the two planes are not consistent, we may decide to suppress one of the extreme segments in the pencil, for example if one of the extreme segments is not closely connected with one of the segments of the previous pencil. By considering the whole pencil, one may also utilize the 3-D distances between all segments of the pencil to suppress the extreme segment whose distance to the closest segment is above a given threshold. If no such segments can be suppressed, we may choose an alternative new pencil.

(3) 3-D interpretation and labeling of the complete pencil

The entire pencil is next sent to the Geometry Module which computes the equation of the plane and the 3-D coordinates of each segment. An important special case occurs when the chosen pencil contains only one segment, and not enough information is available to compute the equation of the plane. In this case, *a flat earth assumption* is applied and the single segment's 3-D structure is computed based on the last computed plane. The label of this segment is the one of the closest 3-D segment in the previous pencil (*3-D connection*). The second segment of the left shoulder in Figure 13 is an example of such a case.

When the pencil contains at least two segments, a general labeling process, described below, is applied. First, the couples are computed from the 3-D information given by the Geometry Module (see Section 2.2). If there is only one cou-

ple, it is labeled by 3-D connections. Otherwise, we compute the width of the road. Two thresholds, which define the minimal and the maximal widths for a road, are given. Therefore, two cases can occur:

(1) The width is not included in the permissible interval. If the width is too small, we attempt to relax the constraints on the formation of the pencil; therefore, one or several segments may be added to the pencil and the width of the road is computed again. If the width of the road is too large, we attempt to suppress some of the segments of the pencil which seem inconsistent with the other segments, in particular by considering the 3-D distances. Figure 10 shows an example of such a situation. Segments 1 and 2 represent the road and segment 3 may be, for example, the horizon line. Unfortunately, segments 1,2 and 3 belong to the same pencil; but when the 3-D distances are computed with the assumption that the three segments are parallel, the distance of segment 3 to the two segments 1 and 2 is very large in comparison to the distance between 1 and 2 and also in comparison to a given maximum road width. In this case, segment 3 will be suppressed. Once this segment has been suppressed, a new attempt at labeling is made. An example of such a situation can be seen in Figure 11.

(2) The width of the road is included in the given bounds. The labeling process begins; the extreme couples are labeled as *shoulders* or *road borders* and the other *inside* couples are labeled as *lane markers* or *discarded* depending on 3-D distances and symmetries.

(4) Finding the missing segments

If, by the grouping of segments into couples or by the comparison between the previous labeled pencil and the new labeled pencil, some segments are declared *missing*, we try to find them by relaxing the constraints on the formation of a pencil; the cluster corresponding to a vanishing point is enlarged. Then if another pencil can be grouped with the current one to form a larger pencil, the missing elements are searched for among the new segments belonging to this other pencil.

(5) Construction of the temporary scene model

Each new labeled pencil becomes a new patch in our temporary scene model which includes all the successive road patches computed in an image, defined by the equations of the corresponding surface planes and the 3-D coordinates of the segments representing the road shoulders and the road borders; this scene model is relative to the vehicle coordinate system. Once the entire image is processed, this scene model will be given to the Representation Module, which computes a representation in the world coordinate system; for more details see [1].

Finally, we turn to the criteria for terminating the analysis of an image. Most frequently, termination occurs when there are no segments in the current pencil. This may occur either during the interpretation of a pencil (due to suppression of segments), or when we are choosing the *next best* pencil. In such cases, although we could go back to the Image Processing Module and ask for more information in the neighborhood "following" the end of the last labeled

segment, or backtrack to the previous segments, we currently terminate processing of the current image.

4. RESULTS

Figures 11 to 16 show some results of this reasoning process for the bootstrap and the feed-forward modes; the interpretation is labeled **LS** for *left shoulder*, **LR** for *left road*, **CL** for *center line*, **RR** for *right road*, **RS** for *right shoulder* and **D** for *discarded.*

Figure 11 shows the set of segments extracted from the image of a straight road for which the left road and left shoulder as well as right shoulder have been extracted. In this case, only one pencil is found and all the other lines are discarded.

Figure 12 shows the image of two intersecting roads. Since, currently, the "intersection model" is not yet included in our Knowledge Base, the lines corresponding to the intersection are discarded. For the other lines, one pencil is formed with the first left segment and part of the right segment (which is cut by the Geometry Module) and then a second pencil is created with the second left segment and the second part of the right segment. This is represented in our 3-D model by a sequence of two patches corresponding to the decreasing slope of the road.

The bootstrap results of Figure 13 are grouped into two pencils which represent approximately the turn in the road; the horizon line is discarded.

The feed-forward results in Figures 14 to 16 are interpreted as a sequence of many pencils; first, the two bottom segments of the left and right windows determine the first pencil. As explained previously (Section 2.1), the Geometry

Module sends back all the segments with their far end-points belonging to the same line, perpendicular (in 3-D) to the direction of the road. Therefore, most of the time, one of these two first segments is cut by the Geometry Module; then, the next pencil which is chosen includes this part which was cut and the next segment of the other side, and so on. In this case, the reasoning involves mainly checking the consistency of a new patch with the previous ones. These pencils, constructed with feed-forward results, give a better approximation of the road geometry and the structure of the terrain than the ones constructed using the bootstrap results.

5. FUTURE EXTENSIONS AND CONCLUSION

We have described in this report the Reasoning Module of our navigation system, as applied to the road following task. Several extensions to this first version are being planned.

One of the extensions is to be able to define several interpretations with a confidence value assigned to each of them. This capability implies the possibility of going back to the Image Processing Module to ask for partial processing of a particular region of the image; new Image Processing results may increase the confidence of one interpretation compared to another.

This last extension would be even more useful if it could be combined with the ability to fuse independent symbolic descriptions extracted by the Image Processing module; this would represent a major extension to this reasoning process. In particular, the boundary-based and region-based descriptions are complementary descriptions [4], as illustrated in the lower right quadrant of Figure 17. For example, we can use the grouping of lines into pencils to select parameters for the segmentation process; we could then utilize the segmentation results to construct a model of the road out to a much greater distance (some 100 meters). Other independent symbolic descriptions given by stereo vision or active ranging may also be combined with the boundary-based and region-based descriptions. In general, combining evidence from several complementary descriptions also leads to a greater confidence in the interpretation of the scene. This extension would be relevant to recognition of shadows, patchy road surfaces, etc.

This increased flexibility in the scheduling of the vision and reasoning activities is useful not only in the bootstrap mode, as was described previously, but also in the feed-forward mode. In this mode, currently, Image Processing, groupings into pencils, 3-D interpretation and 3-D reasoning are sequential operations; so, for example, the reasoning cannot proceed until the image domain symbolics are extracted from the entire image. The next version of the system will ask for two segments at a time, compute their 3-D interpretation and then reason about the consistency of the new surface patch and road edges relative to the 3-D model built from the previous pencils. The 3-D model of the road includes some information such as changes in slope and orientation of the road or width of the road. It can be updated by utilizing a terrain data base, and in this way predict "events" such as intersections or sharp turns. If the new segments are consistent with the previous model, a new model will be built and new segments analyzed; if not, other segments can be obtained in the same windows or new windows can be defined from the 3-D model.

The Vision Executive represents the centralized source of control of the vision part of the system. Therefore, in order to integrate all these different visual capabilities, both the Vision Executive and Knowledge Base must be capable of evolving incrementally. The implementation of the Visual Knowledge Base as a rule-based system or a frame-based system, including rules and defining hierarchies of objects with their attributes and an inference mechanism, seems at this point the best choice for a knowledge-based system [3–7].

I thank Drs. A. Waxman and L. Davis for all the useful discussions I had with them, T. Siddalingaiah for his help in testing the entire system, and Dr. L. Davis for his many valuable comments on early drafts of this paper.

REFERENCES

[1] A.M. Waxman, J. Le Moigne, L.S. Davis, E. Liang and T. Siddalingaiah, "A Visual Navigation System For Autonomous Land Vehicles", University of Maryland, Center for Automation Research Technical Report 139, July 1985.

[2] J. Le Moigne, A.M. Waxman, B. Srinivasan and M. Pietikainen, "Image Processing for Visual Navigation of Roadways", University of Maryland, Center for Automation Research Technical Report 138, July 1985.

[3] G. Reynolds, N. Irwin, A. Hanson, E. Riseman, "Hierarchical Knowledge-Directed Object Extraction Using a Combined Region and Line Representation", *Proceedings DARPA Image Understanding Workshop*, October 1984, 195–204.

[4] T.O. Binford, "Survey of Model-Based Image Analysis Systems", *International Journal of Robotics Research*, Vol. 1, No. 1, Spring 1982, 18–64.

[5] R. Fikes and T. Kehler, "The Role of Frame-Based Representation in Reasoning", *Communications of the ACM*, Vol. 28, No. 9, Sept. 1985, 904–920.

[6] F. Hayes-Roth, "Rule-Based Systems", *Communications of the ACM*, Vol. 28, No. 9, Sept. 1985, 921–941.

[7] M. Minsky, "A Framework for Representing Knowledge", in *The Psychology of Computer Vision*, P. Winston, Ed., McGraw-Hill, New York, 1975, 211–277.

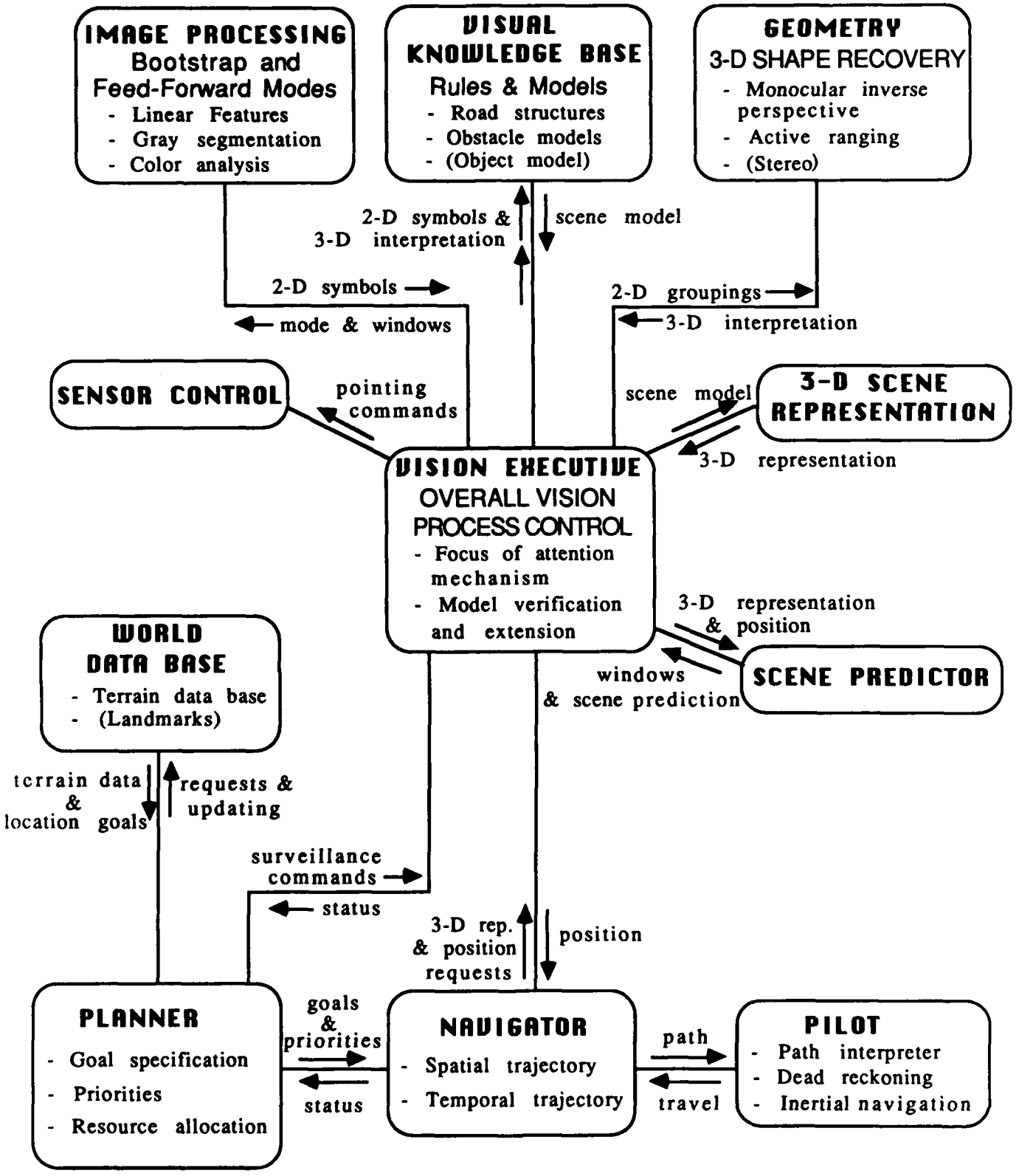
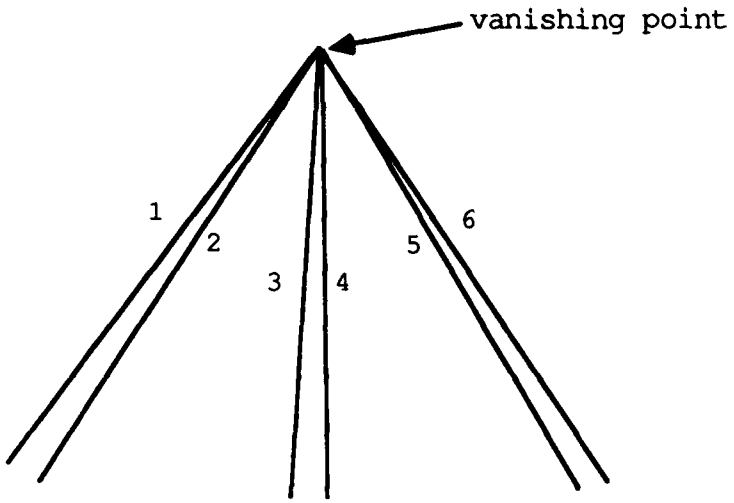
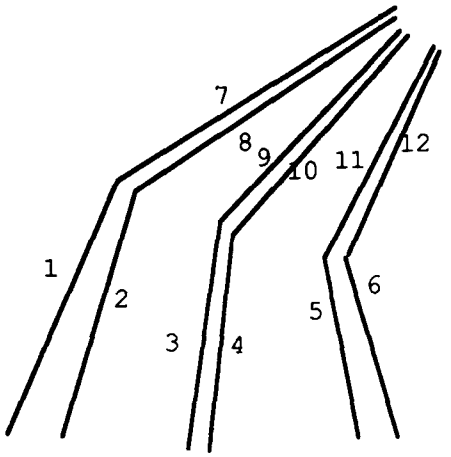


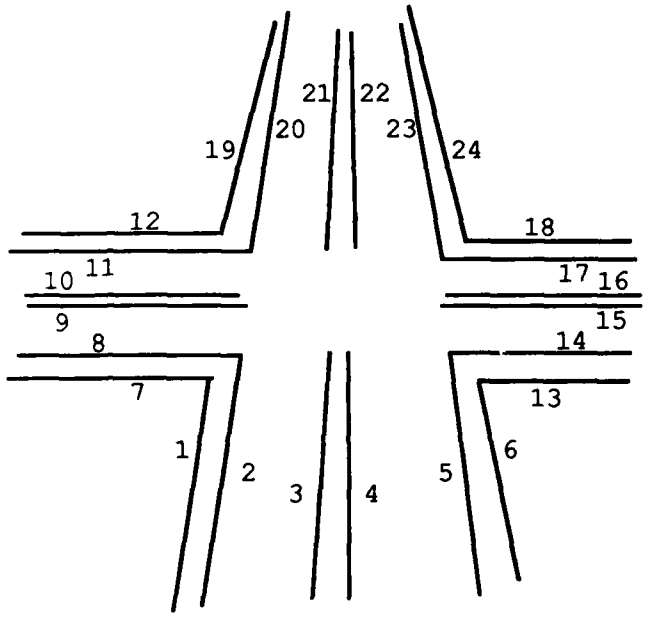
FIGURE 1.
System Architecture



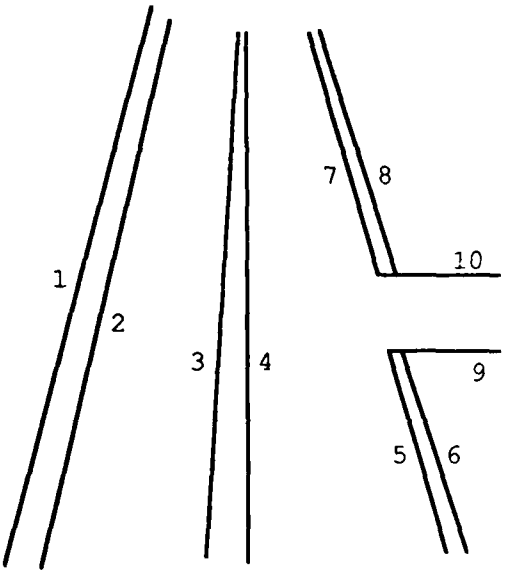
(a) simple road (1 pencil)



(b) turn and/or slope (2 pencils)



(c) intersection with/without slope
(1 pencil, paralllels)



(d) intersection on one side
(1 pencil, parallel)

FIGURE 2.
Different Examples of Ideal Pencils
(case 2 lanes at most)

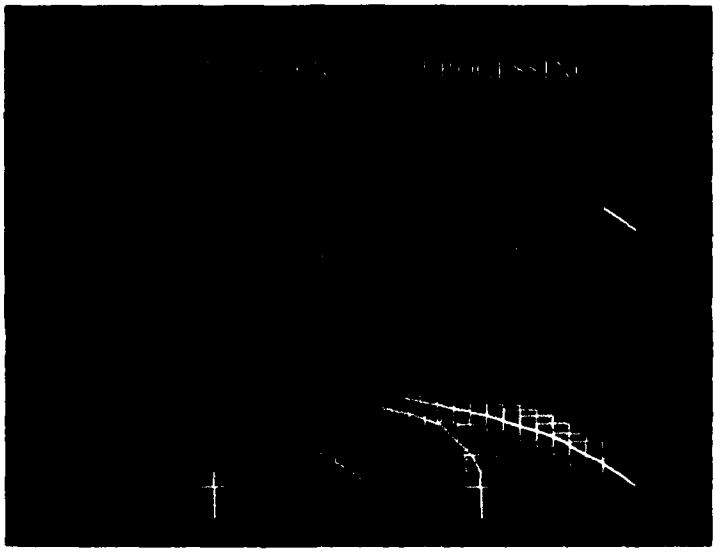


FIGURE 3.
Feed-forward Image Processing Results

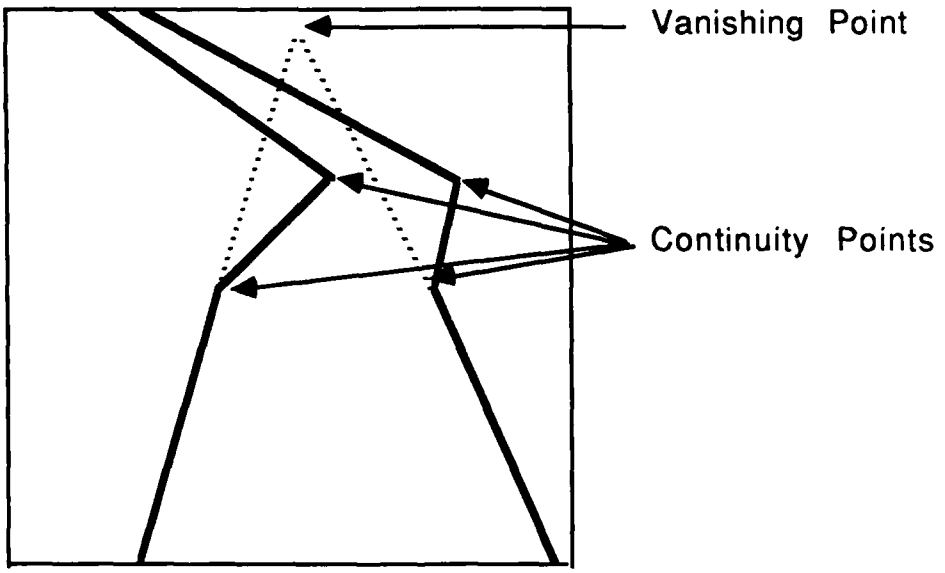


FIGURE 4.
Example of continuity points and vanishing point

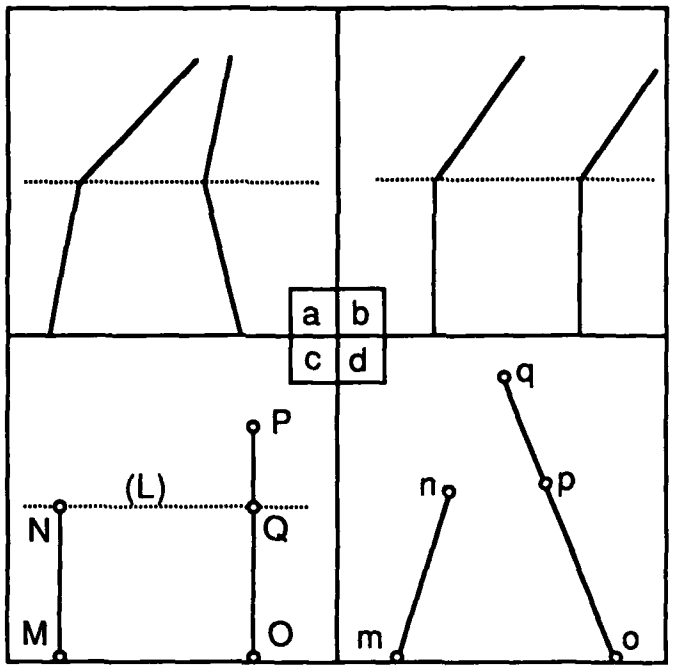


FIGURE 5.
3-D reconstruction of road geometry
a. 2-D pencils
b. 3-D segments reconstructed from 2-D pencils of a/
c. Example of a "cut" segment
d. 2-D pencils leading to 3-D lines of c/

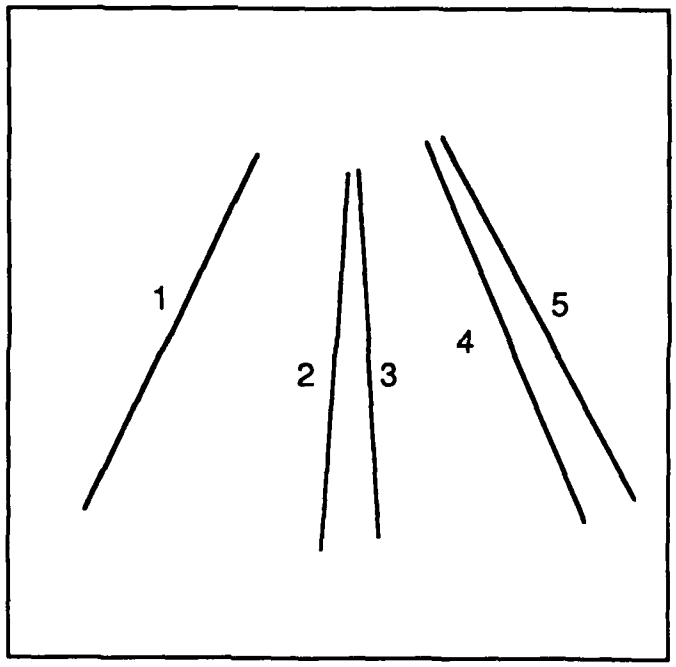


FIGURE 6.
Example of couples

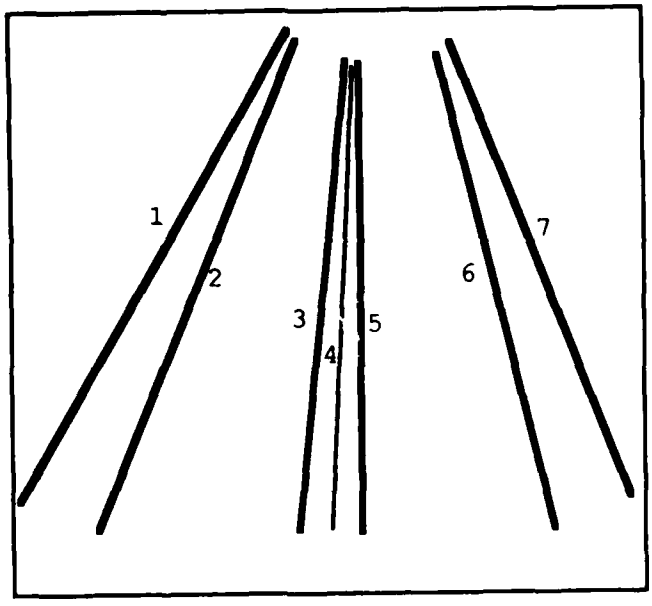


FIGURE 7.
Line Segments Terminology
1- border of the left shoulder
2- left border of the road
3- left border of the lane marker
4- midline of the lane marker
5- right border of the lane marker
6- right border of the road
7- border of the right shoulder

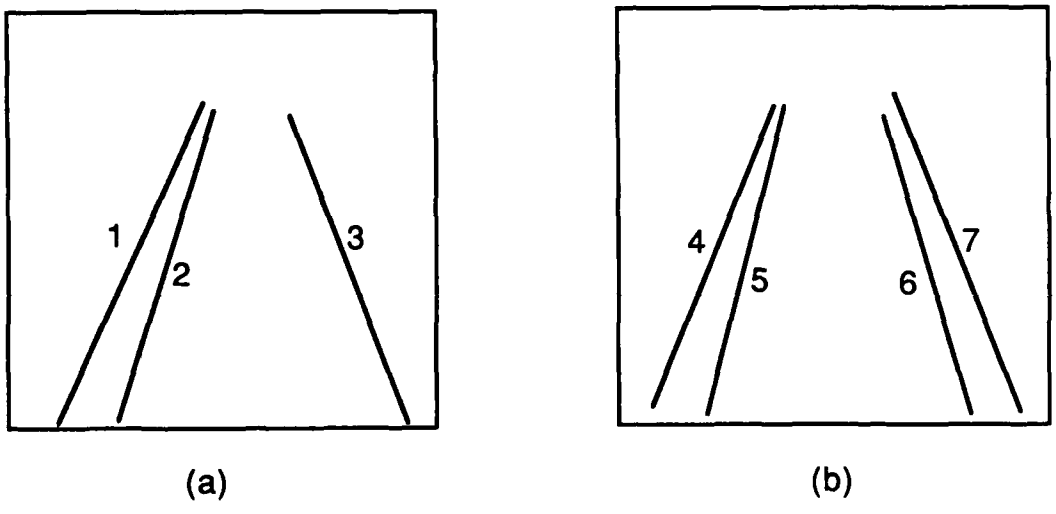


FIGURE 8.
Example of a "missing element"
a. First frame
b. Second frame

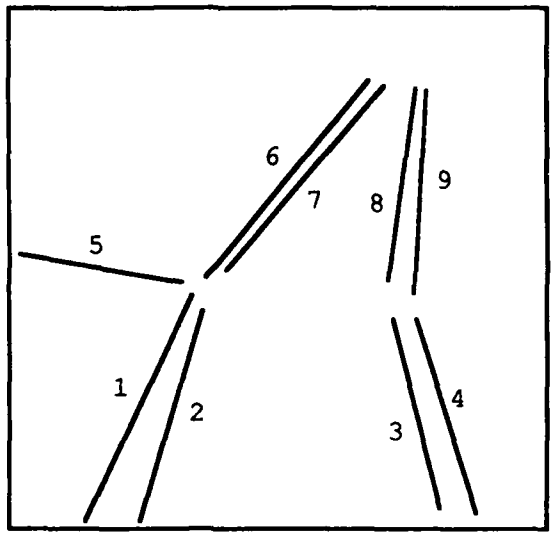


FIGURE 9.
Definition of the Choice Function

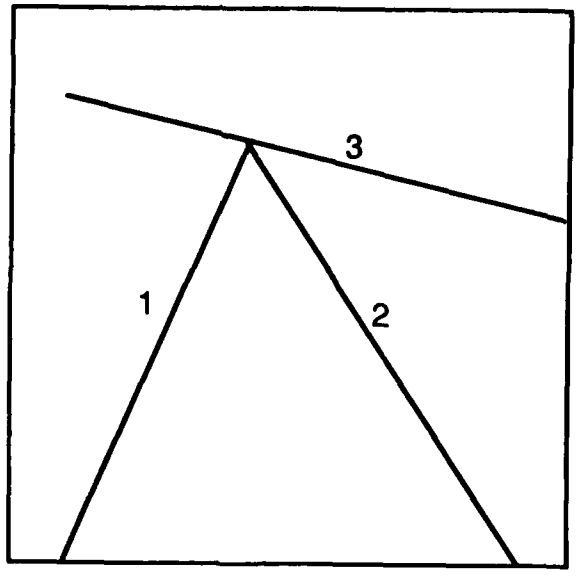


FIGURE 10.
Example where a line of the pencil has to be suppressed

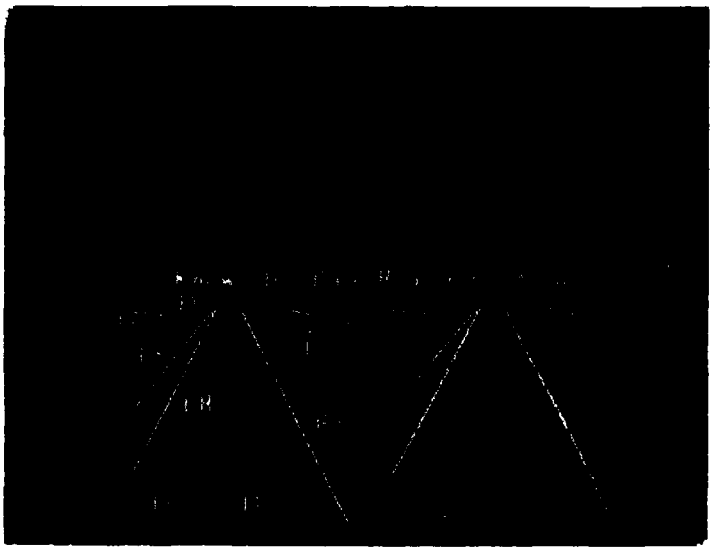


FIGURE 11.
"Straight Road"; Bootstrap Reasoning Results
a. Original image
b. Extracted lines with labeling
c. Superposition original and lines

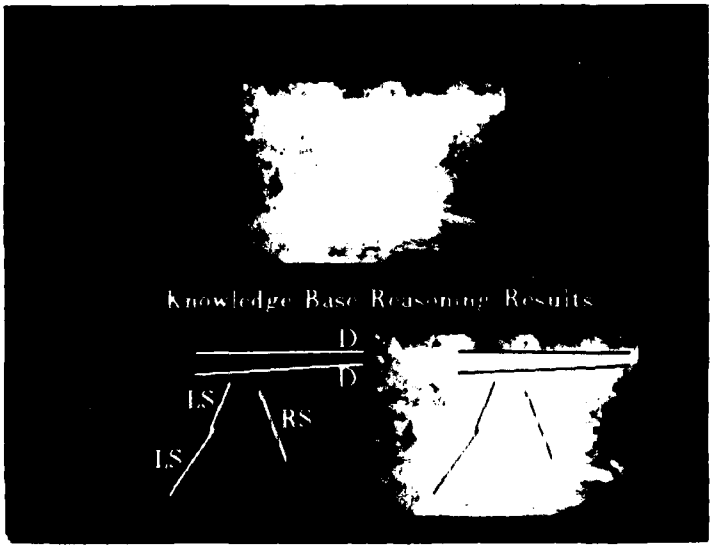


FIGURE 12.
"Intersecting Road"; Bootstrap Reasoning Results
a. Original image
b. Extracted lines with labeling
c. Superposition original and lines

FIGURE 13.
"Turn"; Bootstrap Reasoning Results
a. Original image
b. Extracted lines with labeling
c. Superposition original and lines

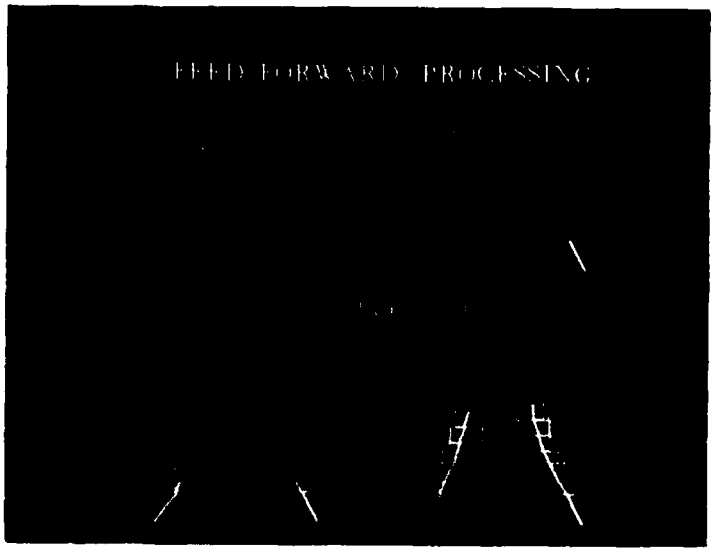


FIGURE 14.
Simulator image; "Slope"; Feed-forward Results

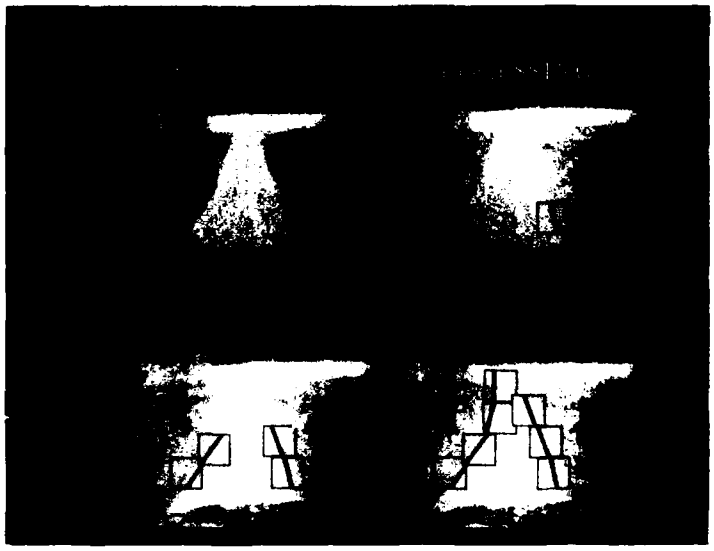


FIGURE 15.
"Intersecting Road"; Feed-forward Results

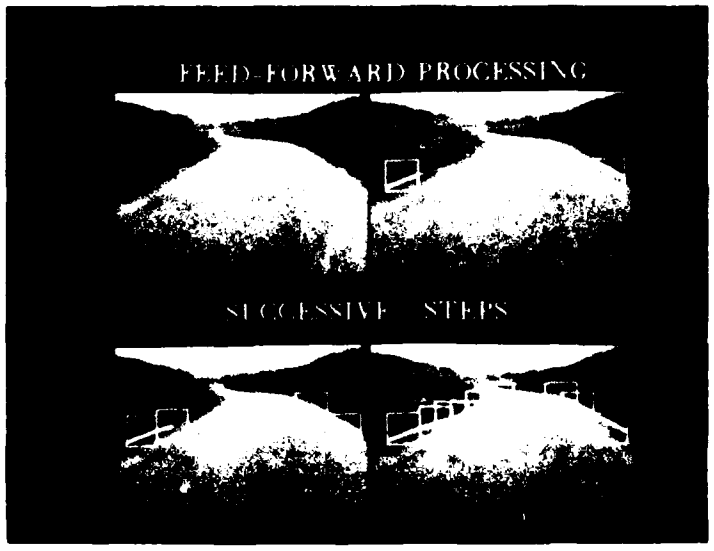


FIGURE 16.
"Bending Road"; Feed-forward Results

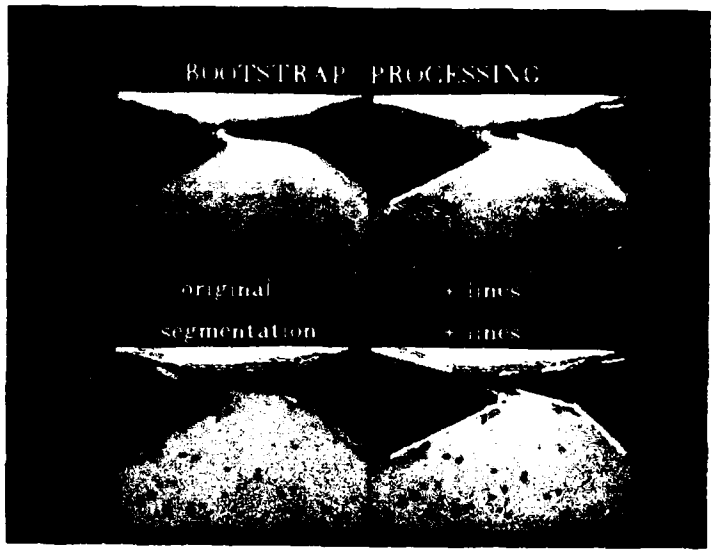


FIGURE 17.
Bootstrap Image Processing Results

END

1-87

DTIC