RSD-TR-18-85

DETECTING ROAD EDGES USING HYPOTHESIZED VANISHING POINTS[1]

Shih-Ping Liou

Ramesh C. Jain

Department of Electrical Engineering and Computer Science
The University of Michigan
Ann Arbor, Michigan 48109

November 1985

CENTER FOR RESEARCH ON INTEGRATED MANUFACTURING

Robot Systems Division

COLLEGE OF ENGINEERING
THE UNIVERSITY OF MICHIGAN
ANN ARBOR, MICHIGAN 48109-1109

[1] This research was supported by General Dynamics under contract #DEY-600483.

TABLE OF CONTENTS

ABSTRACT

In this paper, we propose a parallel algorithm called Vanishing-Point Road-Edge Detector (VPRE Detector), which uses the vanishing points to extract the roadway boundaries. The determination of vanishing point as well as the detection of roadway boundaries is done simultaneously. Several experimental results for real scenes are given. Our experiments with real-world road images demonstrates the efficacy of the proposed approach.

Index Terms

Autonomous navigation, edge detection, machine vision, vanishing point, visual navigation.

1. Introduction

Autonomous vehicles are receiving growing attention from computer vision and artificial intelligence researchers [WMS85, TYA85, Mor81, Mor83, Nil69, GSC79, Tsu84, YIT83, IMB84, TMM85, DaK85, WST85, GiS85, PLN86]. The roadway boundary[1] location problem has become an important problem in visual navigation tasks. In a constrained environment this problem is easy, but for real roads in different real world situations, this has been a formidable task. The difficulties are due to variable contrast in images, a wide range of intensity values of the road surfaces, and presence of intersection and other confusing artifacts for the algorithm. To make this problem harder, usually, the processing speed of the algorithm should be satisfactory for driving a vehicle above a specified minimum speed.

One of the earliest autonomous navigation systems was the Stanford Cart and Robot Rover described in [Mor81, Mor83]. In that system, there was no need to detect roadway boundaries. An interest operator[Mor81] was used for both obstacle identification and path planning. Neither the Cart nor the Rover could operate in real time; the Cart moved $1m$ every 10 to 15 minutes in lurches, whereas the Rover moved up to $1m$ every 10 minutes.

A heuristic route planning systems, capable of forming the planning foundation of an autonomous ground vehicle, was described by Gilmore and

[1] The term roadway boundary and road edges will be used interchangeably in this paper.

Semeco[GiS85]. They believe that communication between the knowledge-based subsystem, in charge of the vision, planning, and conflict resolution aspects was required to make autonomous vehicles functional in a real world environment. Another expert system called LES was developed for interpreting aerial photographs[PLN86]. The contextual information was used to improve the performance on regions which were difficult to classify.

A visual navigation system is being built at the University of Maryland to enable a vehicle to follow roads. Their approach comprises a *boot-strap* phase [WMS85] and a *feed-forward* phase [DaK85]. the methods they use in the former phase are all from the conventional image processing techniques such as filtering, sobel edge detector, histogram analysis, segmenting, shrinking and expanding, and Hough transform. The initial finding of the road edges needed 120 CPU seconds on a VAX 780. In the latter phase, they used the *pivot* point, the endpoint of the line in the previous window, to constrain the search space as well as to eliminate the use of line fitting techniques. Like the idea presented in [WST85], they selected windows instead of processing the entire image.

A mobile robot system designed in Japan was described in [TYA85]. The knowledge of the environment, the abundance of vertical edges in the scene, and the flatness of the floor, were arranged in constraints for the dynamic scene analysis. This robot moved at a speed of 0.3m/sec in a building containing moving objects.

Another system FIDO, based on the Stanford Cart and Robot Rover and improved by using parallel hardware, reduced the run time from 15 minutes to less than a minute per step [TMM85]. Inigo *et al.* [IMB84] proposed a set of algorithms, implementable in real time, for roadway boundary location and obstacle detection. However, since only left road edge was detected in their algorithms, this method would face difficulties not only in the estimation of roadway boundaries but in the determination of moving direction also.

The first complete vision system for an autonomous land vehicle was presented by Wallace *et al.*[WST85]. Not only low-level motor drivers but top-level control loop and user interface were both covered in their system; however, after several mechanisms were tried to extract lines and edges, none of them worked all the time. The following are the assumptions they made:

(a) The road is straight; thus, predictions can be made by linearly extending the road edges.

(b) The camera's calibration, roadway, and orientation with respect to the road are known.

(c) The ground is locally level and all candidate edges arise from ground feature.

This system, essentially a feed-back system, used the model edges to generate two small subimage rectangles in which to search for the left and right roadway boundaries. After obtaining all edges in subimage rectangles, the system selected a pair of extracted edges as the roadway boundaries. However, without further information about the direction of the roadway boundaries, the system will probably lead to a poor performance especially in handling badcontrast pictures.

By simplifying the problem and using some ideas of the vanishing point analysis, we propose an algorithm called Vanishing-Point Road-Edge Detector to extract the roadway boundaries. In order to avoid using any data-driven line fitting or approximation technique in the algorithm, a set of convergent lines is used to speed up the process. Several performance measure functions are developed for determining the position of vanishing point.

Our approach has some similarities with the mechanism used in the *feed-forward* phase developed at the University of Maryland. We reduce the computational complexity by selecting windows based on the location of the vanishing point. The road continuity assumption and the feed-back idea appear to be omnipresent, and our system is no exception.

In the following sections, image analysis techniques and our proposed algorithm will be described first. A more detailed description of the implementation as well as the experimental results are presented next. Finally, future directions of this proposed technique are described in our conclusion.

2. Image Analysis

In this section, both camera and road models will be described first. Following that, review of the research on vanishing point analysis is given. Finally, the requirements as well as the analysis of the roadway boundary location problem will be addressed.

2.1. Camera Model

A camera mounted on the front of the vehicle acquires the image of the roadway that has been distorted by perspective. Using this image to find the roadway or measure distance requires an explicit camera model. The camera models have been extensively discussed in [DuH73, NeS79]. The version we use in our paper is based on [DuH73] and is illustrated in Fig. 1. This camera is translated from the origin, panned through an angle θ, and tilted through an angle ϕ. Also, we use the two coordinate systems--global coordinate system formed by (X,Y,Z), and picture coordinate system represented by (X' ,Z')— to rectify the camera centric situation. In order to use linear matrix operator to represent a nonlinear projection, the homogeneous coordinates are commonly used to represent both object and picture points. For convenience, we write the offset l, the vector from gimbal center to the lens center, as $(l_{1,}l_2+f\ ,l_3)^t$.

Applying the system geometry stated above to the direct perspective transformation equations presented in [DuH73] gives the coordinates of a point (x,y,z) in an image as

$$\begin{aligned} x &= f\frac{(x-x_0)\cos\theta+(y-y_0)\sin\theta-l_1}{-(x-x_0)\cos\phi\sin\theta+(y-y_0)\cos\phi\cos\theta+(z-z_0)\sin\phi-l_2} \\ z' &= f\frac{(x-x_0)\sin\phi\sin\theta-(y-y_0)\sin\phi\cos\theta+(z-z_0)\cos\phi-l_3}{-(x-x_0)\cos\phi\sin\theta+(y-y_0)\cos\phi\cos\theta+(z-z_0)\sin\phi-l_2} \end{aligned} \tag{1}$$

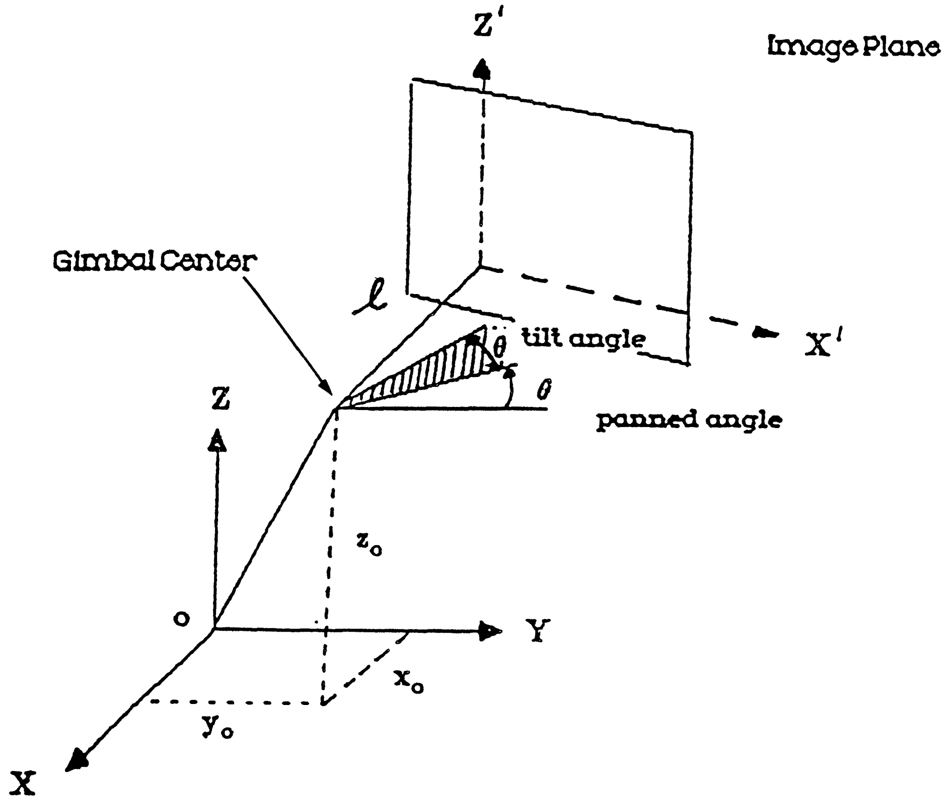

Fig. 1 Camera Model

2.2. Road Model

According to the assumption we made and the camera model shown in Fig. 1, the roadway boundary can be modeled as:

$$y = mx + b \tag{2}$$

where m is the slope of roadway boundaries and b is a constant.

2.3. Vanishing Point Analysis

Since the perspective projection of any set of parallel lines which are not parallel to the image plane will converge to a vanishing point [FoD82], computing these points can allow us to constrain the possible positions of roadway boundaries and other known structures in an image. Determination of the vanishing points has, therefore, attracted the attention of many researchers [Bar82, TYA85, MaA84, Ken79, Bad74].

Kender [Ken79] presented two approaches to determining vanishing points; one of them maps line segments with a common vanishing point into circles that pass through the origin, and the other transforms lines with the same vanishing point into a common line in the involute space. Badler [Bad74] discussed a method based only on cross products of line segments to extract vanishing points. A computationally inexpensive algorithm which is similar to [Bad74] was described by Magee and Aggarwal [MaA84] for the determination of vanishing points, once line segments in an image had been determined. The relationship between vanishing points, vanishing lines, and the gradient space was explored by Shafer *et al.* [SKK82]. Tsuji, Yagi, and Asada [TYA85] used the property of invariance which vanishing points had from translation to make the estimation of the rotational component. In addition, since the vertical edges in their scenes were strong and reliable, the tilt angle of camera could be used to provide an approximate location of the vanishing point and, therefore, search of the input image for these lines was rather easy.

Most researchers proposed methods to either determine vanishing points from the given line segments or to search line segments which are converged to the given vanishing point. However, none of them gives the solution to both determining vanishing point and finding line segments at the same time. In this paper, we will present an efficient parallel algorithm to detect the roadway boundaries as well as vanishing point simultaneously.

2.4. Problem Requirements and Analysis

To find road edges, one approach is to detect all edge points in a frame and then interpret these points. Conventional edge detectors do not provide accurate solutions to the roadway boundary location problem without some kind of line fitting operation after edge detection. Since most techniques for line fitting are slow, the task of roadway boundary location in real time is usually difficult.

To simplify the problem, low curvature roadway boundaries and locally level ground driving are assumed. In that case, the tangent direction to the roadway edges, at any time instant, can provide sufficient information for vehicle guidance. Let us define the road vanishing point at time t, denoted by RVP_t, to be a vanishing point at which the tangents to the current roadway boundaries at time t will converge.

Now, since the location of the vanishing point does not change much between two contiguous frames, we can assume that its location in the current frame will be same as in the previous frame. This fact allows combining road edge detection and vanishing point determination in one step. By assuming a search window around the previous-frame position of the vanishing point, we may find the *best* vanishing point in the current frame using a measure based on edges and their characteristics in the frame.

If the exact tilt angle is detected by some physical instruments as δ, the search space of RVP at time t is easily determined as (See Fig. 2)

$$S_t = \left\{ (x,z) \| \; |x - RVP\_X_{t-1}| \le w \;\; \text{and} \;\; z = \frac{-f \sin\delta}{\cos\delta} \right\} \tag{3}$$

Where w is a constant and RVP_X is the x coordinate of RVP.

$$z = \frac{-f \sin\delta}{\cos\delta} \tag{4}$$

is the so-called vanishing line [SKK82].

Since the RVP is the point, where the extensions of two most obvious straight edges, i.e. the left and right roadway boundaries, meet, our performance measure should try to detect two best converging straight lines. Due to noise and other problems, parts of lines may not be present. The factors influencing the performance measure should be based on

1) length of the line,

2) average gradient magnitude of all the points on the line, and

3) consistency of the direction of each point on the line that converge to the hypothesized vanishing point. Besides, the position of the line also plays vital role in determining road edges. We decided to use the following four factors which dominates the RVP determination problem:

(a) the length of the detected line formed by continuous line segments associated with a vanishing point.

(b) the positions of these detected lines.

(c) the average gradient magnitude of the points on the detected lines.

(d) the consistency of those line segments on a detected line.

Unlike most approaches of controlling the vehicle by finding the road center, the proposed technique may even simplify this computation. The turning angle can be obtained by a simple lookup table, in which each entry indicates the value we got from RVP_X_t - RVP_X_{t-1}. Since this paper is only concerned with the solution to the roadway boundary location problem, we will not discuss the control problem in more details.

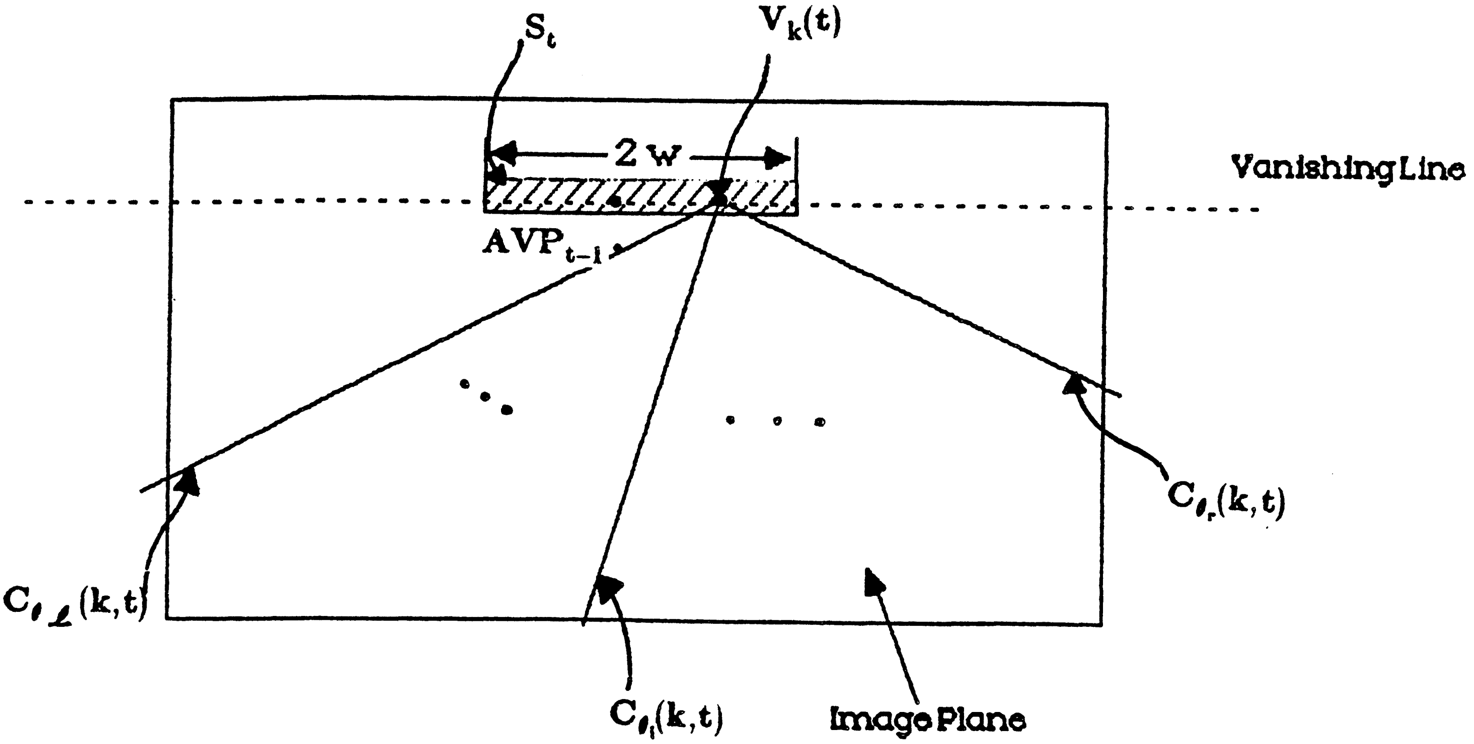

Fig. 2 Convergent Lines

3. Proposed Vanishing-Point Road-edge Detector

The proposed algorithm can be implemented on multiprocessors such that each processor essentially works using a hypothesized vanishing point.

According to Eq. (3) and within a preset tolerance α, the coordinates of all possible vanishing points can be calculated as:

$$\begin{aligned} x_k &= RVP\_X_{t-1} - w + k\,\alpha \\ z_k &= \frac{-f\ \sin\delta}{\cos\delta} \quad aaa \quad k = 0, \frac{2w}{\alpha} \end{aligned} \tag{5}$$

Let us denote the kth vanishing point at time t as $V_k(t)$ and define the *Convergent Line* with an angle s, represented by $C_s(k,t)$, to be a line segment that crosses the image plane and its extension passes the vanishing point $V_k(t)$.

If we assume that both roadway boundaries are located in the range from θ_l to θ_r, the number of convergent lines associated with any vanishing point, for the given resolution θ_α, is

$$\frac{\theta_l - \theta_r}{\theta_\alpha} + 1$$

And the coordinates of the points which are located on these different convergent lines can also be calculated before the vehicle begins moving.

Let us define *Pixel Element* $(d,m)_{i,j}$ to be a pair of data derived from the estimate of the gradient magnitude m and edge orientation d at pixel location (i,j) and let ι_r be the adjustment ratio used for the safety purpose.

According to Eq(5), for the VPRE detector algorithm, $\frac{2w}{\alpha}+1$ processors are used to achieve the parallel performance. This algorithm is stated as

1. For processor k ,($k=0,\frac{2w}{\alpha}$), we start from $V_k(t)$ at time t

    * produce the corresponding pixel elements
    * scan the image and get the next convergent line $C_s(k,t)$ as illustrated in Fig. 2.
    * Compute the performance measure of each convergent line.
    * Divide all the line sets into two categories. One is the set of all lines with positive slopes, and the other is the one with negative slopes.
    * For each category,
        * Find the line set i with the maximum performance measure m_i of all line sets.
        * If there exists a line set j with performance measure m_j such ; that

$$|m_{set\_found} - m_j| \leq t_r * m_{set\_found} \qquad (6)$$

        Set *set_found* to i, *value_found* to m_i.

        * Line set *set_found* is the roadway boundary in this category.
    * Set the performance measure of $V_k(t)$ to be the minimum of the performance measures in these two categories.

2. Select the vanishing point which has the maximum performance measure as the *RVP* and its corresponding detected lines in each category as the road edges.

4. Implementation Details

Due to good computational efficiency and reasonable performance, the Sobel edge finder is used to compute the direction and magnitude of the gradient at a point *i,j*, where the intensity is $f_{i,j}$. Thus we define the partial derivative estimates

$$S_x(i,j) = f_{i+1,j+1}+2f_{i+1,j}+f_{i+1,j-1}-f_{i-1,j+1}-2f_{i-1,j}-f_{i-1,j-1} \tag{7}$$

$$S_y(i,j) = f_{i-1,j+1}+2f_{i,j+1}+f_{i+1,j+1}-f_{i-1,j-1}-2f_{i,j-1}-f_{i+1,j-1} \tag{8}$$

The S_x and S_y partial derivative estimates are then combined to form an estimate of the gradient magnitude *m*, and direction *d* by

$$\begin{aligned} m &= |S_x|+|S_y| \\ d &= \tan^{-1}\left[\frac{S_y}{S_x}\right] \end{aligned} \tag{9}$$

The magnitude and direction maps are both quantized to eight bits to form the pixel element.

Now let us define the *Edge Point* to be a point whose *pixel element* is $(d,\ m)$ such that for a convergent line $C_s(k,t)$

$$|d - s| \leq \iota_\theta \text{ and } m \geq \iota_s \tag{10}$$

where t_θ and t_s are both threshold values.

Note that we are interested only in those edge points that are on convergent lines. Other points will not enter into considerations irrespective of their strength. Next, we form line segments out of these edge points. We use ι_d to represent the allowed maximum length of continuous non-edge points, and ι_l to mean the minimum length of continuous edge points to form a line segment.

From all points along $C_s(k,t)$ for some s and $V_k(t)$, we extract all the qualified line segments. Finally, a decision is made over all convergent lines by a measure of the system's confidence in how well the lines are formed.

For finding good lines, we used the following performance measure functions with normalized value in a range from 0 to 1.

A. The length of the detected line formed by continuous line segments associated with a vanishing point"

$$Leng(x,k) = \frac{D_{xk}}{D_{Max}} \tag{11}$$

Where D_{xk} =total length of all line segments detected along the convergent line $C_x(k,t)$, and D_{Max} is the maximum length value for all possibilities.

B. The position of these detected lines

$$Close(x,k) = \frac{1}{1+q_y} \tag{12}$$

Where q_y is the y coordinate of the points which is nearest to the X axis and also located on the detected lines.

C. The average gradient magnitude of the points on the detected lines"

$$Mag(x,k) = \frac{\sum_{y \in E_x} S_y}{255 N(E_x)} \tag{13}$$

Where S_y means the gradient magnitude component of the pixel element $(d,s)_y$ and $N(E_x)$ means the amount of all edge points found along the convergent line $C_x(k,t)$.

D. The consistency of those line segments on a detected line whose direction is θ_x"

$$Cons(x,k) = \frac{\sum_{y \in E_x} \frac{\pi - 2|d_y - \theta_x|}{\pi}}{N(E_x)} \tag{14}$$

Where d_y means the direction component of the pixel element $(d,m)_y$

The performance measure $\lambda(x, k)$ is then defined as

$$\lambda(x,k) \tag{15}$$

Where $w_{i,} i = 1..4$ are the weights and the value of x and k are associated with a convergent line $C_x(k,t)$ at time instant t.

And, by using λ^+ and λ^- to represent the measure in the positive-slope and negative-slope category separately, the purpose is to find x and y such that

$$\begin{aligned} \lambda^+(x,k) &= MAX \left\{ \lambda^+(i,k) \right\} \quad \forall \; i \in A \\ \lambda^-(y,k) &= MAX \left\{ \lambda^-(i,k) \right\} \quad \forall \; i \in B \end{aligned} \tag{16}$$

Where A and B indicate the the set of convergent lines in the positive-slope and negative-slope category separately.

and define χ as

$$\chi(k) = MIN \left\{ \lambda^+(x,k), \lambda^-(y,k) \right\} \tag{17}$$

By combining the information from RVP_{t-1} as well as χ, the process will make the final decision on the RVP_t position. The decision is made by selecting the vanishing point $V_{l(t)}$ such that

$$\chi(l) = MAX \left\{ \chi(k) \right\} \quad \forall \quad V_k(t) \in S_t \tag{18}$$

4.1. Experimental Results

The performance of the Vanishing-Point Road-Edge Detector was tested using several real-world road image sequences. The results of part of these sequences are briefly illustrated in Table 1. The first column in this table shows the number of the picture sequence, while column two shows the number of recorded frames as well as the number of sampled pictures. Following that, the search window S_t, as given in Eq(3), is described in the next field by using the coordinates of the left-bottom and right-top points. The coordinate of the

RVP found using the described approach are given in the last column.

Figures 3-8 show the original pictures and the road edges with the vanishing points found by our approach. For the first four sequences short-interval frames were used, whereas for the remaining sequences we used long-interval frames. In each Figure, (a) is the original picture. (b) and (c) are the direction and magnitude maps derived from the Sobel edge detector. The output picture with the lines showing the roadway boundaries and the box illustrating the search space is presented in (d). We found that the resulting edges are very good even for the poor contrast pictures (Figure 3, 7 and 8). In addition, the results for the Sequence 5 and 6 show that it is not necessary to work with short-interval frames in a sequence. Thus the proposed method offers a robust method for detecting road edges for visual navigation tasks.

5. Conclusion

This paper presents an approach for locating the road edges. Since the direct application of the conventional image processing techniques in the past does not provide us with efficient as well as reliable roadway boundary location, we utilized some general assumptions to simplify the analysis, and introduced the idea of vanishing point analysis to speed this approach. Our approach finds the best vanishing point for the road edges using a criterion based on the quality of line segments detected in an image. This step combines the task of locating a vanishing point and finding road edges. In this respect this approach may be considered similar to approaches proposed for finding the focus of expansion by pivot point position in any two continuous frames is changed more significantly than the vanishing point position is. Secondly, since we assume the vanishing point might move to other place, the algorithm we proposed has more flexibility. Finally, in their case the selected window has to cover the left as well as right roadway boundaries in the following frame, the selection of windows is hence not as easy as ours.

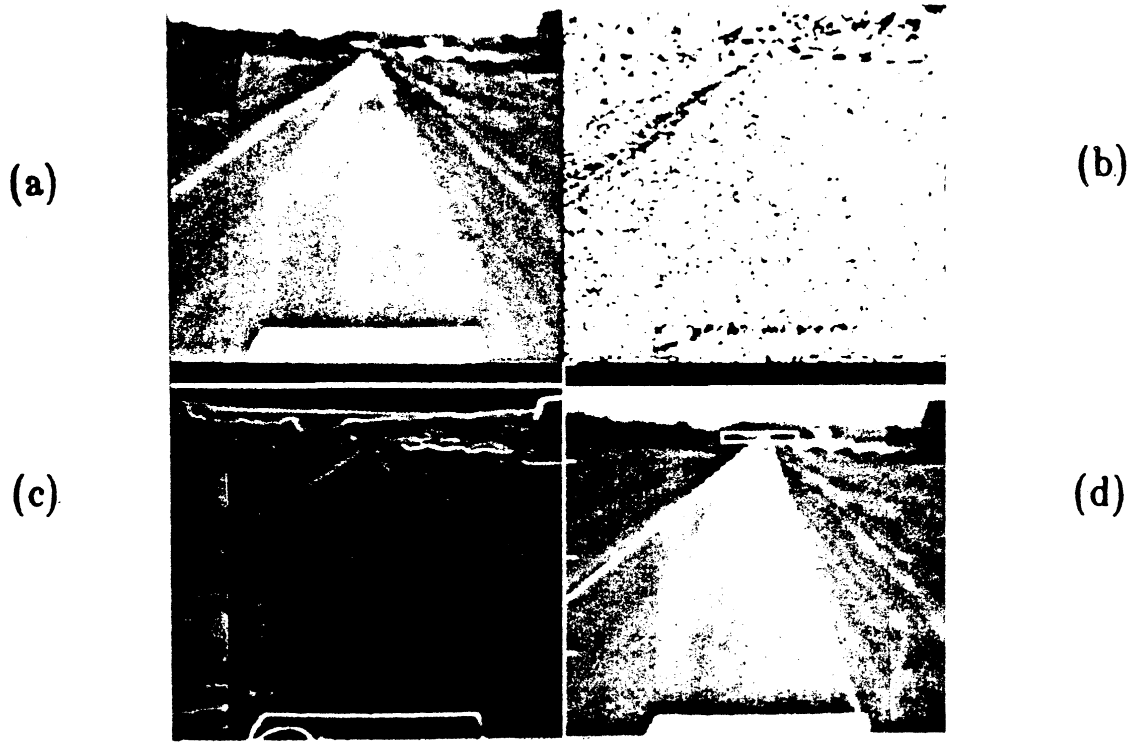

Fig 3. The first frame in Picture Sequence 1
a) Original Picture b) Direction Map c) Magnitude Map d) Output Picture

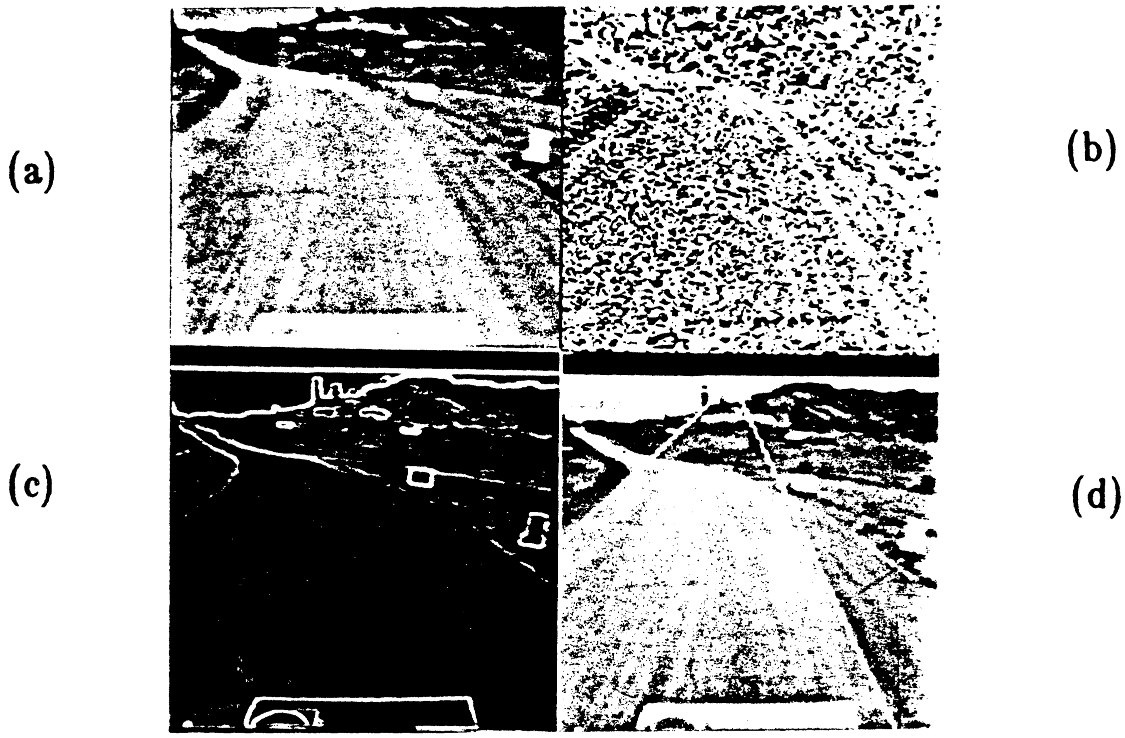

Fig 4. The first frame in Picture Sequence 2
a) Original Picture b) Direction Map c) Magnitude Map d) Output Picture

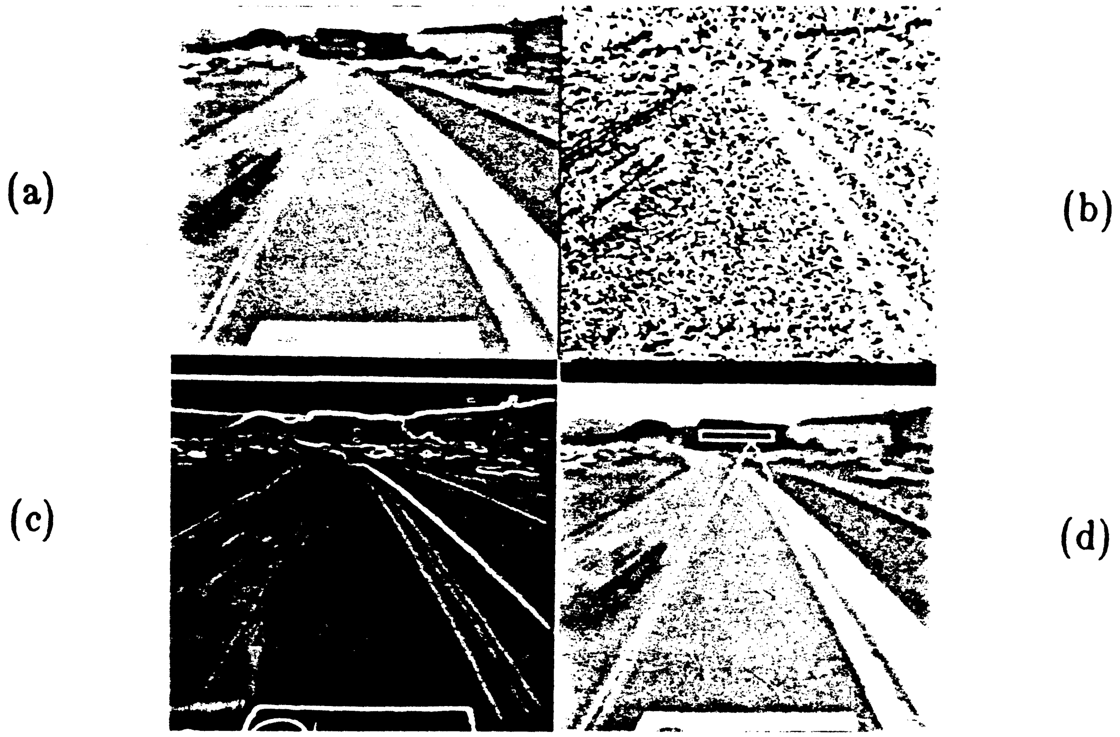

Fig 5. The first frame in Picture Sequence 3
a) Original Picture b) Direction Map c) Magnitude Map d) Output Picture

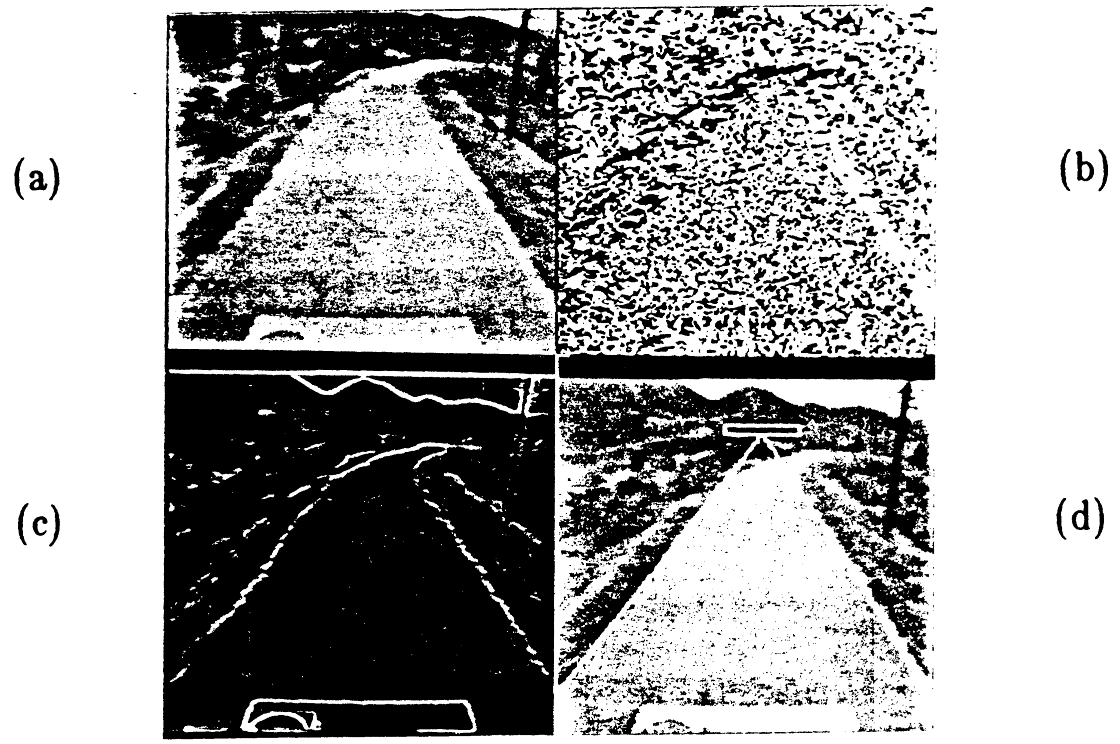

Fig 6. The first frame in Picture Sequence 4
a) Original Picture b) Direction Map c) Magnitude Map d) Output Picture

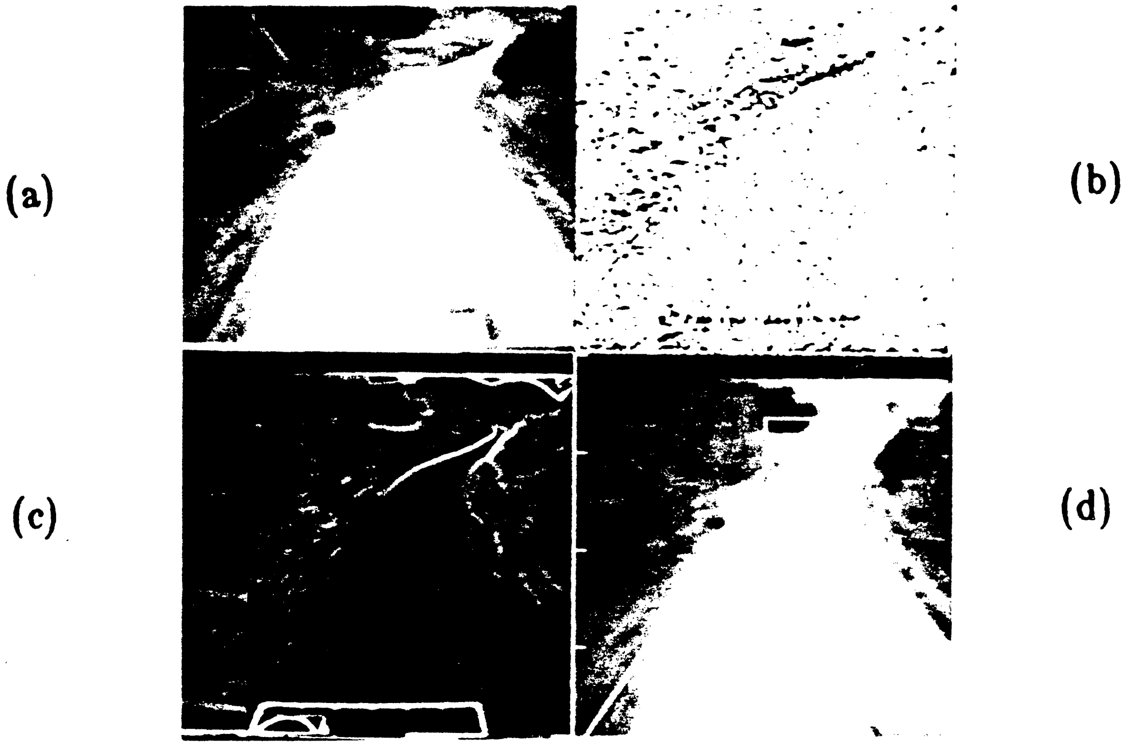

Fig 7. The first frame in Picture Sequence 5
a) Original Picture b) Direction Map c) Magnitude Map d) Output Picture

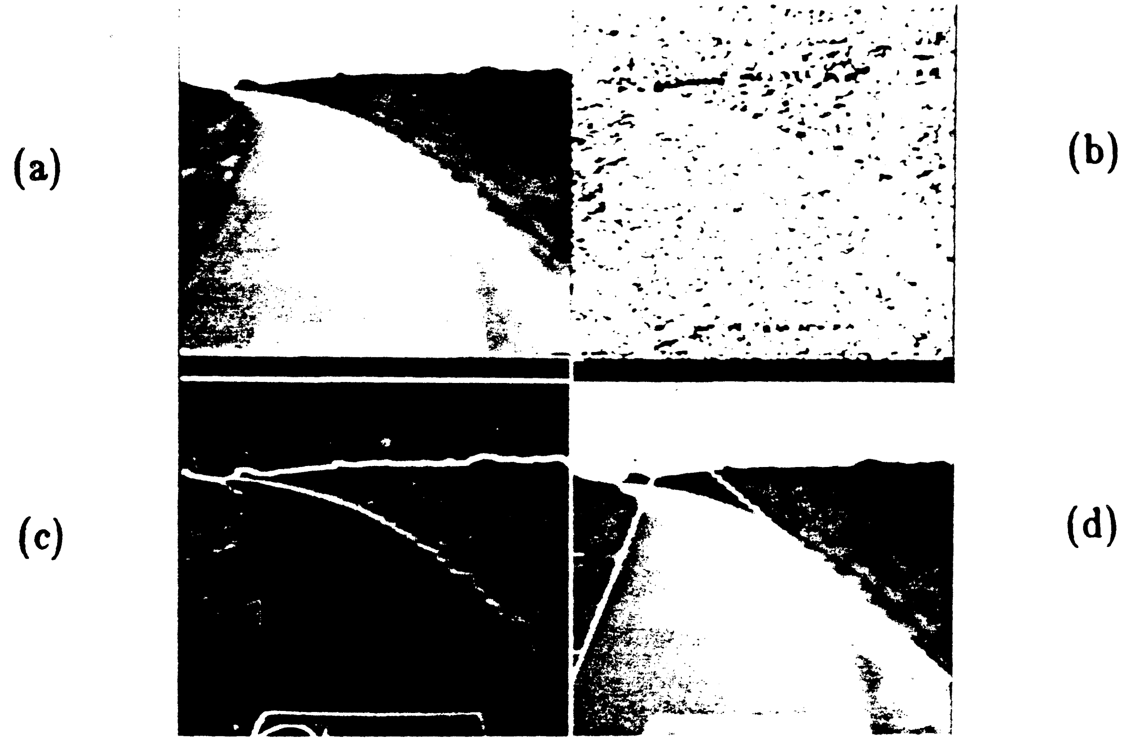

Fig 8. The first frame in Picture Sequence 6
a) Original Picture b) Direction Map c) Magnitude Map d) Output Picture

References

[Bad74]

Badler, N., "Three-Dimensional Motion from Two-Dimensional Picture Sequence", *Proc IJCPR,* Copenhagen, Denmark, Aug 13-15, 1974, 157-161.

[Bar82]

Barnard, S. T., "Methods for interpreting perspective images", *Proc. of the Image Understanding Workshop,* Stanford University, Palo Alto, Calif., 1982, 193-203.

[DaK85]

Davis, L. S., and T. R. Kushner, "Road Boundary Detection for Autonomous Vehicle Navigation", *CS-TR-1538, Center for Automation Research,* University of Maryland, July 1985.

[DuH73]

Duda, R. O. and P. E. Hart, *Pattern classification and scene analysis,* Wiley, New York, 1973.

[FoD82]

Foley, J. D. and A. Van Dam, *Fundamentals of interactive computer graphics,* Addison-Wesley, Reading, Mass., 1982.

[GiS85]

Gilmore, J. F., and A. C. Semeco, "Terrain Navigation through

Knowledge-Based Route Planning", *Proc of the 9th IJCAI*, Los Angeles, CA, Aug 18-23, 1985, 1086-1088.

[GSC79]

Giralt, G., R. Sobek, and A. Chatila, "A multi-level planning and navigation system for a mobile robot", *Proc. 6th IJCAI*, Tokyo, 1979, 335-337.

[IMB84]

Inigo, R. M., E. S. McVey, B. J. Berger, and M. J. Wirtz, "Machine vision applied to vehicle guidance", *IEEE trans. on Pattern Analysis and Machine Intelligence*, 6, No 6, 1984, 820-826.

[JeJ84]

Jerian, C. and R. Jain, "Determining motion parameters for scene with translation and rotation", *IEEE trans. on PAMI*, Vol 6, No 4, July 1984, 523-530.

[Ken79]

Kender, J., "Shape from texture: An aggregation transform that maps a class of textures into surface orientation", *Proc IJCAI*, Tokyo, Japan, Aug 20-23, 1979.

[Law84]

Lawton, D. T., "Processing Dynamic Image Sequence from a Moving Sensor", *Ph. D. Dissertation (TR 84-05)*, Computer and Information Science Department, University of Mass., Amheret, 1984.

[MaA84]

Magee, M. J. and J. K. Aggarwal, "Determining vanishing points from perspective images", *Computer Vision, Graphics, and Image Processing,* 26, 1984, 256-267.

[Mor81]

Moravec, H. P., *Robot rover visual navigation*, UMI Press, Ann Arbor, MI., 1981.

[Mor83]

Moravec, Hans P., "The Stanford Cart and the CMU Rover", *Proc of the IEEE*, 71, July 1983, 872-884.

[NeS79]

Newman, W. M. and R. F. Sproul, *Principles of interactive computer graphics*, McGraw-Hill, New York, 1979, 333-365.

[Nil69]

Nillson, N. G., "A mobile automation", *Proc. 1st IJCAI*, 1969, 509-520.

[PLN86]

Perkins, W. A., T. J. Laffey, and T. A. Nguyen, "Rule-based Interpreting of Aerial Photographs Using LES", To appear in *Optical Engineering,* Feb., 1986.

[SKK82]

Shafer, S. A., T. Kanade, and J. R. Kender, "Gradient Space under Orthography and Perspective", *IEEE 1982 Workshop on Computer Vision :*

Representation and Control, Franklin Pierce College, Rindge, New Hampshire, Aug 23-25, 1982, 26-34.

[TMM85]

Thorpe, C., L. Matthies, and H. Moravec, "Experiments and Thoughts on Visual Navigation", *Proc 1985 IEEE Intern. Conf. on Robotics and Automation*, St. Louis, March 1985, 830-835.

[Tsu84]

Tsuji, S., "Monitoring of a building environment by a mobile robot", *Proc. 2nd Int. Symp. on Robotics Research*, Kyoto, 1984.

[TYA85]

Tsuji, Saburo, Yasushi Yagi, and Minoru Asada, "Dynamic scene analysis for a mobile robot in a man-made environment", *Proc 1985 IEEE Intern. Conf. on Robotics and Automation*, St. Louis, March 1985, 850-855.

[WMS85]

Waxman, Allen M., Jacqueline Le Moirne, and Babu Srinivasan, "Visual navigation of roadways", *Proc 1985 IEEE Intern. Conf. on Robotics and Automation*, St. Louis, March 1985, 862-867.

[WST85]

Wallace, R., A. Stentz, C. Thorpe, H. Moravec, W. Whittaker, and T. Kanade, "First Results in Robot Road-Following", *Proc of the 9th IJCAI*, Los Angeles, CA, Aug 18-23, 1985, 1089-1095.

[YIT83]

Yachida, M., T. Ichinose, and S. Tsuji, "Model-guided monitoring of a building environment by a mobile robot", *Proc. 8th IJCAI*, Munich, 1983, 1125-1127.