Proceedings of the SPIE 1825 on Intelligent Robots and Computer Vision XI, November 16-18, 1992, Boston, MA.

Obstacle Detection and Terrain Characterization Using Optical Flow Without 3-D Reconstruction

Gin-Shu Young[1,2], Tsai-Hong Hong[1], Martin Herman[1], Jackson C. S. Yang[2]
[1]National Institute of Standards and Technology (NIST), Bldg 220, Rm B124, Gaithersburg, MD 20899
[2]Robotics Laboratory, Department of Mechanical Engineering, University of Maryland, College Park, MD 20742

Abstract

For many applications in computer vision, it is important to recover range, 3-D motion, and/or scene geometry from a sequence of images. However, there are many robot tasks which can be achieved by extracting relevant 2-D information from the imagery and using this information directly, without recovery of the 3-D information. In this paper, we focus on two tasks, obstacle avoidance and terrain navigation. A novel approach to these two tasks has been developed without 3-D reconstruction. This approach is often called purposive active vision. A linear relationship, plotted as a line and called a reference flow line, is determined. The difference between a plotted line and the reference flow line can be used to detect discrete obstacles above or below the reference terrain. For terrain characterization, slopes of surface regions can be calculated directly from optical flow. Some error analysis is also done. The main features of this approach are that (1) discrete obstacles are detected directly from 2-D optical flow; no 3-D reconstruction is performed; (2) terrain slopes are also calculated from 2-D optical flow; (3) knowledge about the terrain model, camera-to-ground coordinate transformation, or vehicle (or camera) motion is not required; (4) the error sources involved are reduced to a minimum, since the only information required is a component of optical flow. An initial experiment using noisy synthetic data is also included to demonstrate the applicability and robustness of the method.

1 INTRODUCTION

For many applications in computer vision, it is important to recover range, 3-D motion, and/or scene geometry from a sequence of images [19] [8]. However, there are many robot tasks which can be achieved without recovery of such information [14]. These tasks can be achieved by extracting relevant 2-D information from the imagery and using this information directly, without 3-D reconstruction, for controling the task of the robot. One such task is obstacle avoidance, while another is terrain navigation. In this paper, we focus on these two tasks, and show how obstacle detection and terrain characterization can be achieved without recovering 3-D scene information. This approach is often called *purposive active vision* [1].

To operate autonomous vehicles safely, obstacles must be detected and the terrain should be analyzed before path planning and obstacle avoidance activity is undertaken. Obstacles are defined as any region in space which a vehicle should not or cannot traverse, such as protrusions (objects lying on top of the terrain), depressions (potholes, ruts, gullies in the terrain), or steep terrain shown in figure 1a. The goal of this work is to develop a simple, robust, and general method for obstacle detection and terrain characterization for ground vehicles or air vehicle landing. The method developed allows the ground vehicle to navigate through man-made roadways or natural outdoor terrain. The method also allows the aircraft to land on known or unknown terrain.

Most existing methods perform obstacle detection based on *range information (or 3-D scene geometry)* obtained either from active sensors (such as laser scanners, radars, or ultrasonics) or passive sensors (cameras) [2] [3] [4] [5] [7] [9] [17] [20].

In our approach, optical flow is used without being converted to range, 3-D motion, and/or scene geometry. A straight line is used as a detector in an appropriate image-based space, i.e. a space whose coordinate axes represent parameter values extracted from the image domain. For example, suppose that the parameters x and $\dot{y}$ denote the x-component of image position and the y-component of optical flow, respectively. Then, if a region in the x, $\dot{y}$ image-based space is not a straight line (figure 1b), the region is potentially regarded as an obstacle. For terrain characterization, slopes of surface regions can also be calculated directly from $\dot{y}$(x,y).

In the following, previous work is summarized first, then our method is described. The equations derived for the method are discussed and the errors of the terrain slopes analyzed. Finally, experimental results are presented.

2 PREVIOUS WORK

A number of obstacle detection methods have been developed in the past (e.g. [2] [3] [4] [5] [6] [7] [9] [12] [15] [16] [17] [18] [20]). Range information is often employed to solve this problem. This information may be obtained from active sensors, stereo cameras, optical flow, etc. *A priori* knowledge is required by most existing methods. Such knowledge may include sensor-to-ground coordinate transformations, sensor motion, model optical flow fields, road models (or maps), etc. Errors in *a priori* knowledge result in errors in the output.

Many obstacle detection methods based on optical flow have been developed. Sridhar *et al* [17] at NASA Ames Research Center investigated the methodology for obstacle detection for rotorcraft low altitude flight. The obstacle detection problem is posed as the problem of finding range to all objects in the field of view. Range information is obtained by the use of optical flow in their method. Bhanu et al [2] presented an inertial sensor integrated optical flow technique for motion analysis, in which range information is extracted from optical flow for obstacle detection. The method by Hoff and Sklair [9] detected landing hazards for a descending spacecraft, in which an algorithm using range information retrieved from optical flow with known camera motion was developed. Enkelmann [6] detected obstacles by evaluating the difference between calculated optical flow and estimated model flow. The estimated model requires knowledge about the focus of expansion (FOE), the transformation matrix between the camera and vehicle coordinate systems, and the camera motion. In addition, this method works only with a camera translating on a planar surface. Raviv [13] detected obstacles from an optical-flow-based invariant with an assumption of observer undergoing only translational motion in parallel to a planar surface. Mallot *et al* [11] detected discrete obstacles by the use of inverse perspective mapping. This method requires a coordinate transform and has the limitation of the observer (i.e., the camera) moving in the horizontal plane under pure translation.

The work described above is characterized by the following:
(1) range information extracted from optical flow, stereo, or active range sensors is often employed to detect obstacles,
(2) terrain slope, a useful feature for determining traversability, is not usually calculated,
(3) *a priori* knowledge such as coordinate transformations, sensor motion, model optical flow fields, or road models (maps) is often required,
(4) the observer's motion is often limited to translational motion in detecting obstacles directly from optical flow.

3 METHOD

For translational motion, if a line in space is projected into the image, then the component of optical flow perpendicular to this image line, as a function of position along the line, will result in a linear relationship. In figure 2a, if the image line $\overline{ab}$ arises from the line $\overline{AB}$ in space, the component $\dot{y}$ of optical flow (which is defined to be in a direction perpendicular to this image line) vs. image position x is a line shown in figure 2b. The y coordinate of image line $\overline{ab}$ is y^* and the coordinates of the image points a and b in figure 2a are (x_o, y^*) and (x_n, y^*), respectively. The coordinates of the two points F_A and F_B in figure 2b are $(x_o, \dot{y}_o)$ and $(x_n, \dot{y}_n)$, respectively. The line $\overline{F_A F_B}$ in a plot of $\dot{y}$ vs. x is called a *flow line*. A proof of this property is found in section 4. The flow line has enough information to indicate
(1) discontinuities in the flow field along the line,
(2) protrusions or depressions of 3-D objects relative to the line,
(3) slopes of terrain relative to the line.

Although the discussion so far has been concerned only with single image lines, many lines in the image plane can be processed in parallel to analyze the full terrain ahead of the vehicle. An extension of this approach to general motion (translation and rotation) has also been developed [21].

Obstacle detection

To apply the above properties to obstacle detection, three steps are involved.
Step 1: Estimation of the reference flow line.
This line can be obtained from the observed optical flow corresponding to regions on the ground surface near the vehicle. In principle, only two points are required to estimate the reference flow line.
Step 2: Computation of the difference.
The difference between the reference flow line obtained in **Step 1** and the observed flow corresponding to objects projected into the image line is computed. Note that the "objects" here can be discrete obstacles or smooth or uneven terrain in the visible environment.
Step 3: Identification of obstacles.
The computed difference in **step 2** is used to detect obstacles. If the difference is positive, the observed point is considered to be a protrusion relative to the reference line (see figure 3). If the difference is negative, the observed point is regarded as a depression relative to the reference line.

Terrain characterization

For terrain characterization, slopes of surface regions can be computed *directly* from the optical flow without recovering range, 3-D motion, and/or scene geometry. The slopes are functions of the flow values $\dot{y}(x, y)$. The formulas for calculating slope are derived in section 4. Errors in the estimation of terrain slopes due to the uncertainty of the camera center are analyzed in section 5.

Main features

The method has several features:
(1) Minimal a *priori* knowledge - Information about the pose of the camera relative to the ground is not required. Knowledge of the velocity of the camera (direction and magnitude) is not required. Road or terrain models are not required. The major assumption is that the camera motion is a general translation.
(2) Simple - Only one component (e.g., $\dot{y}$) of the optical flow is needed. In principle, normal flow, the component of the optical flow along the local gradient direction, can be used. 2-D visual information (i.e., $\dot{y}$(x,y)) is used directly to detect obstacles and calculate slopes; no 3-D reconstruction is performed. The feature used to detect discrete obstacles is simple - a straight line.
(3) Fast - The method is simple; therefore computation is fast. Image lines can be processed in parallel.
(4) Robust - Since the only required information is $\dot{y}$(x,y), the error sources involved are reduced to a minimum. The reference flow line is estimated from the observed data so that the reference and observed flows share the same error source.

4 DERIVATION

In this section, it is proved that for an image line arising from a line in space, the relationship between one component of the optical flow and the image position is linear when the motion of the objects relative to the camera is translational. This linear relationship is used to denote a reference flow line. Also, the equations for calculation of terrain slope are introduced.

A flow line

Consider the arbitrary line $\overline{AB}$ in space which is projected into the image line $\overline{ab}$. Without loss of generality, let us choose the image x-axis to be parallel to line $\overline{ab}$, and the image y-axis to be perpendicular to $\overline{ab}$. In figure 4, a coordinate frame c attached to the camera is chosen as follows:
(1) Let the camera focal point be the origin O_c.
(2) Let the optical axis be the Z_c axis.

(3) Choose X_c and Y_c to be parallel to the image axes x and y, respectively.

A coordinate frame b is then affixed to the line $\overline{AB}$ as follows:
(1) Let the origin O_b be the point lying on the extended line $\overline{AB}$ the shortest distance from the point O_c.
(2) Let the Z_b axis be along the line $\overline{AB}$.
(3) Choose X_b and Y_b arbitrarily as long as the right hand rule is obeyed.

A point P in the scene can be transformed from frame b to frame c by the equation:

$$\begin{Bmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{Bmatrix} = H_b^c * \begin{Bmatrix} X_b \\ Y_b \\ Z_b \\ 1 \end{Bmatrix} \tag{1}$$

where (X_c, Y_c, Z_c) and (X_b, Y_b, Z_b) are the coordinates of point P in frames c and b, respectively, and

$$H_b^c \equiv \begin{bmatrix} h_{11} & h_{12} & h_{13} & h_{14} \\ h_{21} & h_{22} & h_{23} & h_{24} \\ h_{31} & h_{32} & h_{33} & h_{34} \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2}$$

represents a 4x4 transform matrix from frame b to frame c. Note that, for each instant of time, H_b^c is the same for all points on line $\overline{AB}$. From the pinhole camera model with focal length unity, the image position x is

$$x = \frac{X_c}{Z_c} \tag{3}$$

The equations for optical flow due to translational motion are as follows:

$$\dot{x} = \frac{1}{Z_c}(-T_X + xT_Z) \tag{4}$$

$$\dot{y} = \frac{1}{Z_c}(-T_Y + yT_Z) \tag{5}$$

where $(\dot{x}, \dot{y})$ are the components of optical flow, Z_c is the depth of the object relative to the camera, (x,y) are the components of the image position, and (T_X, T_Y, T_Z) is the translational motion of the object relative to the camera. Note that, for each instant of time, (T_X, T_Y, T_Z) are constants for all points lying on a rigid object. As defined earlier, line $\overline{AB}$ coincides with the Z_b axis, therefore any points lying on line $\overline{AB}$ always have

$$X_b = Y_b = 0 \tag{6}$$

With equations (1)(2)(3) and (6), the following linear relationship can be obtained from equation (5) for all image points lying on line $\overline{ab}$ that arise from points in the scene lying on line $\overline{AB}$:

$$\dot{y} = a_1 + a_2 x \tag{7}$$

where

$$\begin{aligned} a_1 &= \frac{h_{13}}{a_3} \\ a_2 &= \frac{-h_{33}}{a_3} \\ a_3 &= \frac{(h_{34}h_{13} - h_{14}h_{33})}{(-T_Y + yT_Z)} \end{aligned} \tag{8}$$

Equation (7) represents a reference flow line corresponding to one line in space. For each instant of time, the values a_1, a_2, and a_3 are constants for all points on this line in space. Note that the reference flow line can be estimated from two points in principle. This means that specific knowledge of the transformation matrix and camera motion is not required.

Calculation of terrain slope

The terrain slope can be obtained *directly* from the optical flow without recovering range, 3-D motion, and/or scene geometry. In figure 5, the heavy solid lines ($\overline{P_1P_2}$ and $\overline{Q_1Q_2}$) denote the observed terrain from one image line (y=constant). The slope $\tan\alpha$ denotes the observed terrain $\overline{Q_1Q_2}$ relative to the reference terrain line $\overline{P_1P_2}$. The slopes $\tan\theta_1$ and $\tan\theta_2$ denote the lines $\overline{P_1P_2}$ and $\overline{Q_1Q_2}$, respectively relative to the line $\overline{O_cE}$. Here O_c and E are the camera focal point and the image point with the image coordinate (0, y). Therefore, the terrain slope angle α can be obtained from the difference between θ_2 and θ_1. The calculation of the terrain slope is described below.

Let the focal length be unity. The camera-centered coordinates of points O_c and E are (0,0,0) and (0,y,1). Let the image coordinates of object points P_1 and P_2 be (x_{p_1}, y) and (x_{p_2}, y), respectively. From the pinhole camera model, the camera-centered coordinates of points P_1 and P_2 are ($x_{p1}Z_{p1}$, yZ_{p1}, Z_{p1}) and ($x_{p2}Z_{p2}$, yZ_{p2}, Z_{p2}). Here Z_{p_1} and Z_{p_2} denote the depth of points P_1 and P_2, respectively. Therefore, the components in the camera frame of the vectors $\overline{P_1P_2}$ and $\overline{O_cE}$ are ($x_{p2}Z_{p2}$-$x_{p1}Z_{p1}$, y(Z_{p2}-Z_{p1}), Z_{p2}-Z_{p1}) and (0, y, 1), respectively. The angle θ_1 between the two vectors $\overline{P_1P_2}$ and $\overline{O_cE}$ can be expressed as

$$\cos\theta_1 = \frac{(\overline{P_1P_2}\cdot\overline{O_cE})}{(|\overline{P_1P_2}|\,|\overline{O_cE}|)} = \frac{1}{(\sqrt{1+\tan^2\theta_1})} \tag{9}$$

With the components of the two vectors, the slope $\tan\theta_1$ for the line $\overline{P_1P_2}$ relative to $\overline{O_cE}$ can be found as

$$\tan\theta_1 = \frac{(x_{p2}K_p - x_{p1})}{(\sqrt{y^2+1}(K_p-1))} \tag{10}$$

where

$$K_p = \frac{Z_{p2}}{Z_{p1}} \tag{11}$$

Combining equations (5) and (11), we obtain

$$K_p = \frac{\dot{y}_{p1}}{\dot{y}_{p2}} \tag{12}$$

where $\dot{y}_{p_1}$, $\dot{y}_{p_2}$ denote the y components of the optical flow at image positions x_{p_1} and x_{p_2}, respectively.

Similarly, the slope $\tan\theta_2$ for the line $\overline{Q_1Q_2}$ relative to $\overline{O_cE}$ can be found as

$$\tan\theta_2 = \frac{(x_{q2}K_q - x_{q1})}{(\sqrt{y^2+1}(K_q-1))} \tag{13}$$

where

$$K_q = \frac{Z_{q2}}{Z_{q1}} = \frac{\dot{y}_{q1}}{\dot{y}_{q2}} \tag{14}$$

and $\dot{y}_{q_1}$, $\dot{y}_{q_2}$ denote the y components of the optical flow at image positions x_{q_1} and x_{q_2}, respectively.

Finally, the terrain slope angle α can be obtained as

$$\alpha = \theta_2 - \theta_1 \tag{15}$$

This terrain slope angle is a function only of image parameters x, y (the image position) and $\dot{y}$ (a component of optical flow) of various image points. No 3-D reconstruction is needed.

5 ERROR ANALYSIS

It is a common practice to choose the center of the image as the camera center. However it was reported in [10] that the camera center could be off by as many as 25 pixels for three cameras tested. Uncertainties in image positions (x,y) relative to the camera center in the image may cause errors in calculating terrain slopes. The following presents an error analysis for the terrain slopes.

In this analysis, the terrain slope $\tan\theta$ (equation (10) or (13)) is considered to be the terrain line relative to $\overline{O_cE}$ shown in figure 5. Errors in the camera center are considered separately in the x and y directions. The measured image position (x_{mea},y_{mea}) relative to the camera center can be expressed as

$$x_{mea} = x_{act} + \epsilon_x \tag{16}$$

$$y_{mea} = y_{act} + \epsilon_y \tag{17}$$

where (x_{act},y_{act}) denotes the actual image position and (ϵ_x, ϵ_y) denotes the errors of the camera center in the x and y directions. Upper bounds on the errors due to ϵ_x and ϵ_y are derived as follows:

Error due to ϵ_x

The measured terrain slope can be represented as

$$\tan\theta_{mea} = \frac{((x_{act2} + \epsilon_x)K - (x_{act1} + \epsilon_x))}{(\sqrt{y^2+1}(K-1))} = \tan\theta_{act} + \frac{\epsilon_x}{(\sqrt{y^2+1})} \tag{18}$$

Therefore, the upper bounds on the errors can be obtained as

$$|\tan\theta_{mea} - \tan\theta_{act}| <= \epsilon_x \tag{19}$$

In our experiments, we have used a SONY XC-57 camera with a 16 mm focal length. The picture elements for this camera are 510(H) x 492(V) and the sensing area is 8.8mm x 6.6mm. With a 25-pixel displacement from the actual camera center, the upper bounds on the errors for varying measured slope angles θ_{mea} are listed in Table 1. The maximum upper bound on the errors for the range of angles indicated in the Table is only 1.53^o.

Error due to ϵ_y

The ratio of the actual terrain slope to the measured slope can be expressed as

$$\frac{\tan\theta_{act}}{\tan\theta_{mea}} = \sqrt{\frac{((y_{act} + \epsilon_y)^2 + 1)}{(y_{act}^2 + 1)}} = \sqrt{1 + \frac{\epsilon_y^2}{(y_{act}^2+1)} + \frac{(2y_{act}\epsilon_y)}{(y_{act}^2+1)}} \tag{20}$$

Therefore, the upper bounds on the errors can be obtained as

$$\frac{\tan\theta_{act}}{\tan\theta_{mea}} <= 1 + \epsilon_y + \sqrt{\frac{(2y_{max}\epsilon_y)}{(y_{max}^2+1)}} \tag{21}$$

where

$$y_{max} = \frac{6.6mm}{2} \cdot \frac{1}{f} \quad and\ f = 16mm \tag{22}$$

With 25 pixels off the actual camera center, the upper bounds on the errors for varying measured slope angles θ_{mea} are listed in Table 2. The maximum upper bound on the errors is approximately 2.79^o.

6 EXPERIMENTS AND RESULTS

Two initial experiments using noisy synthetic data have been performed. The experiment simulates a camera mounted on top of the vehicle 2 m above the ground (see figures 6 and 7). The first experiment involves detecting a bump a distance of 5 m from the head of the vehicle and a height of 0.3 m above the smooth terrain (see figure 6). The second experiment involves detecting a pothole with a depth of 0.6 m in the terrain, as well as a ramp (see figure 7). The pothole is 6 m away from the vehicle and the beginning of the ramp is 12 m from the vehicle. The ramp was tested with various slope angles (9^o, 18^o, and 27^o).

In the first experiment, two kinds of noise were added to the optical flow field: 5%, and 10%. The results are presented in figures 8a, and 8b. The region above the horizontal line results from a protrusion above the terrain. The gap in the graphs at the end of the protrusion is due to occlusion. In the second experiment, for each slope angle (9^o, 18^o, and 27^o), 10% noise was added to the optical flow field. The results are shown in figures 9a through 9c.

The region below the horizontal line results from a depression in the terrain. The gap in the graphs at the beginning of the depression is due to occlusion. The region with a smoothly increasing value relative to the horizontal line denotes a ramp. The data points collected from the region were fit by a line. The slope of the line represents the slope of the ramp. The actual and calculated slopes of the ramp are presented in table 3 (for the 10% noise case). The performance of this method depends on the quality of the input data. For example, this method may not work if the uncertainty in the range due to the optical flow value is larger than the size of the bump or pothole.

7 CONCLUSIONS

For obstacle detection and terrain characterization, a novel method using optical flow without recovering range, 3-D motion, and/or scene geometry has been developed. This method can be used for ground vehicles navigating in man-made roadways or natural outdoor terrain or for air vehicles landing in known or unknown terrain. The method has several features:
(1) Simple - Only 2-D visual information (e.g., one component of optical flow $\dot{y}$(x,y)) is needed to detect obstacles and calculate terrain slopes.
(2) Fast - The method is simple and therefore computationally fast. Image lines can be processed in parallel.
(3) Robust - The error sources involved are reduced to a minimum since the only required information is $\dot{y}$(x,y).
The initial experiments included in the paper suggest that the approach is effective and robust.

8 ACKNOWLEDGEMENTS

The authors would like to thank Dr. James Albus and Dr. David Coombs for important comments on this work.

References

[1] J. Aloimonos, "Purposive and Qualitative Active Vision," *Proc. Image Understanding Workshop*, 1990.

[2] B. Bhanu, B. Roberts, and J. Ming, "Inertial Navigation Sensor Integrated Motion Analysis for Obstacle Detection" *Proc. IEEE Int'l Conf. on Robotics and Automation*, 1990.

[3] M. J. Daily, J. G. Harris, and K. Reiser, "Detecting Obstacles in Range Imagery," *Proc. Image Understanding Workshop*, 1987.

[4] M. J. Daily, J. G. Harris, and K. Reiser, "An Operational Perception System for Cross-country Navigation," *Proc. Image Understanding Workshop*, 1988.

[5] R. T. Dunlay and D. G. Morgenthaler, "Obstacle avoidance on roadways using range data," *SPIE Mobile Robots*, Vol. 727, 1986.

[6] W. Enkelmann, "Obstacle Detection by Evaluation of Optical Flow Fields," *First European Conf. on Computer Vision*, 1990.

[7] M. Hebert and T. Kanade, "3D Vision for Outdoor Navigation by an Autonomous Vehicle," *Proc. Image Understanding Workshop*, 1988.

[8] D. Heeger and A. D. Jepson, "Subspace Methods for Recovering Rigid Motion I: Algorithm and Implementation," *International Journal of Computer Vision*, in press.

[9] W. Hoff and C. Sklair, "Planetary Terminal Descent Hazard Avoidance Using Optical Flow," *Proc. IEEE Int'l Conf. on Robotics and Automation*, 1990.

[10] R. K. Lenz and R. Y. Tsai, "Techniques for Calibration of the Scale Factor and Image Center for High Accuracy 3-D Machine Vision Metrology," *IEEE Transactions on PAMI*, Vol. 10, No. 5, Sep. 1988.

[11] H. A. Mallot et al., "Inverse perspective mapping simplifies optical flow computation and obstacle detection," *Biological Cybernetics*, Vol. 64, pp177-185, 1991.

[12] D. N. Oskard, T. H. Hong, and C. A. Shaffer, "Real-Time Algorithms and Data Structures for Underwater Mapping," *IEEE Trans. on Systems, Man, and Cybernetics*, Vol. 20, No. 6, 1990.

[13] D. Raviv, "Flat Surfaces: A Visual Invariant," *NISTIR 4794*, NIST, Gaithersburg, MD, Mar. 1992.

[14] D. Raviv and M. Herman, "Visual Servoing for Robot Vehicles Using Relevant 2-D Image Cues," in *Active Vision*, Y. Aloimonos, ed., Lawrence Erlbaum, 1992, to appear.

[15] S. Singh and P. Keller, "Obstacle Detection for High Speed Autonomous Navigation," *Proc. IEEE Int'l Conf. on Robotics and Automation*, 1991.

[16] U. Solder and V. Graefe, "Object Detection in Real Time," *SPIE Vol. 1388 Mobile Robots V*, 1990.

[17] B. Sridhar, R. Suorsa and P. Smith, "Vision Based Techniques for Rotorcraft Low Altitude Flight," *SPIE* , Vol. 1571, 1991.

[18] K. Storjohann, Th. Zielke, H. A. Mallot and W. Von Seelen, "Visual Obstacle Detection for Automatically Guided Vehicles," *Proc. IEEE Int'l Conf. on Robotics and Automation*, 1990.

[19] C. Tomasi and T. Kanade, "The Factorization Method for the Recovery of Shape and Motion from Image Streams," *Proc. Image Understanding Workshop*, 1992.

[20] P. A. Veatch and L. S. Davis, "Range Imagery Algorithms for the Detection of Obstacles by Autonomous Vehicles," *Center for Automation Research Technical Report*, CAR-TR-309, July 1987.

[21] G.-S. Young, T.-H. Hong, M. Herman, and J. C. S. Yang, "New Visual Invariants for Obstacle Detection Using Optical Flow Induced from General Motion," IEEE Workshop on Applications of Computer Vision, Nov. 30 - Dec. 2, 1992, Palm Springs, California.

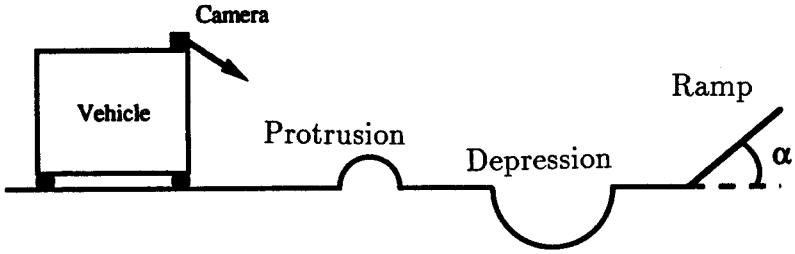

Fig. 1a Terrain with obstacles

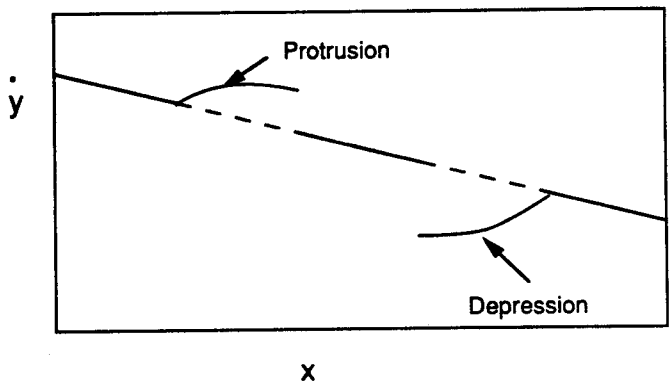

Fig. 1b Detection of discrete obstacles in an image-based space

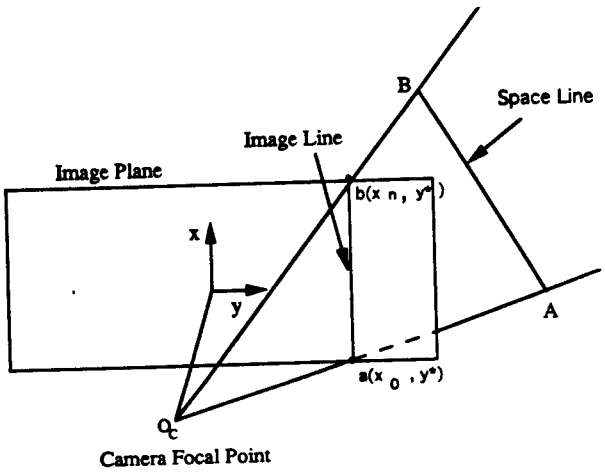

Fig 2a Space line and image line

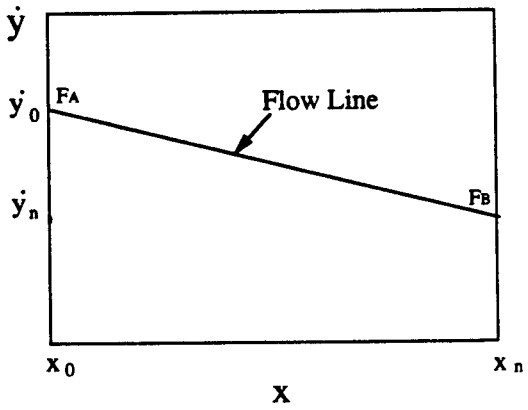

Fig. 2b Flow line in an image-based space

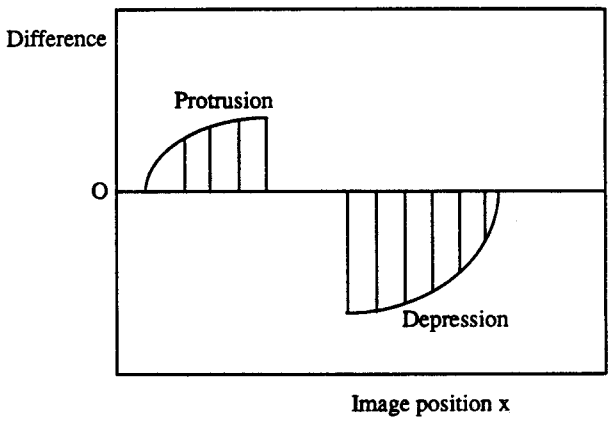

Fig. 3 Detection of discrete obstacles

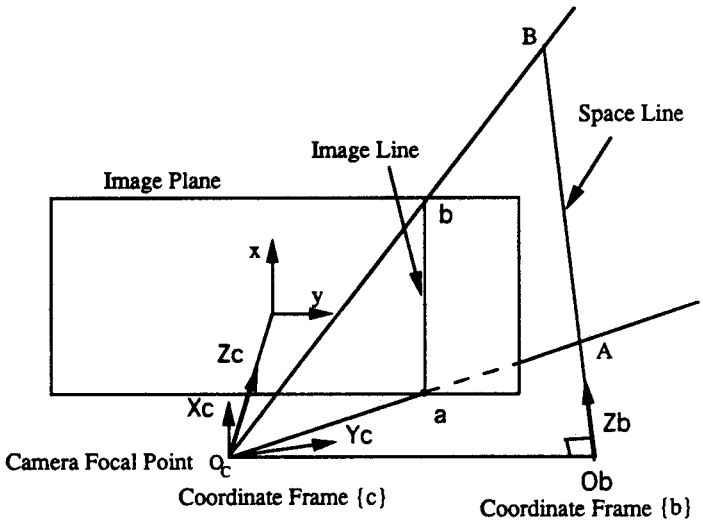

Fig. 4 Definition of two coordinate frames

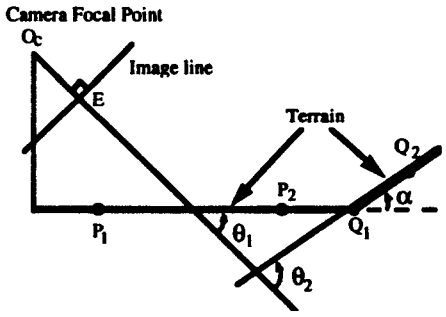

Fig. 5 Terrain slope in side view

θ_{mea} (in degree)	5	10	20	30	45	60	75	89
Upper bound errors (in degree)	1.53	1.49	1.35	1.14	0.76	0.38	0.10	0.00

Table 1 Upper bound errors due to ϵ_x

θ_{mea} (in degree)	5	10	20	30	45	60	75	89
Upper bound errors (in degree)	0.51	1.00	1.86	2.48	2.79	2.36	1.33	0.09

Table 2 Upper bound errors due to ϵ_y

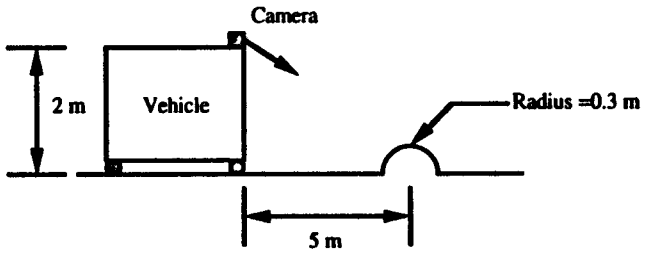

Fig. 6 Side view of terrain with a bump

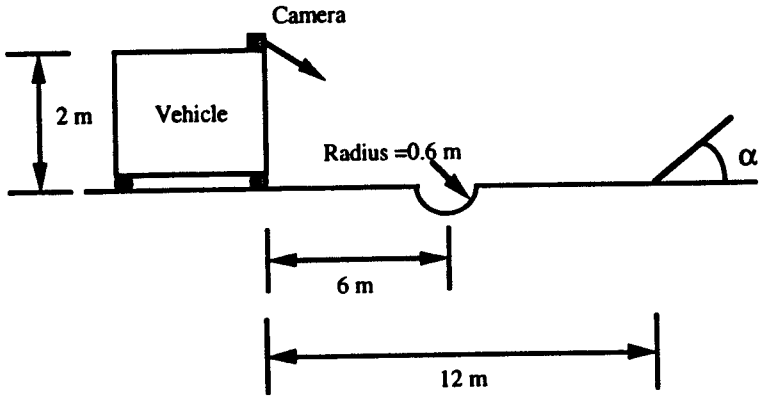

Fig. 7 Side view of terrain with a pothole and a ramp

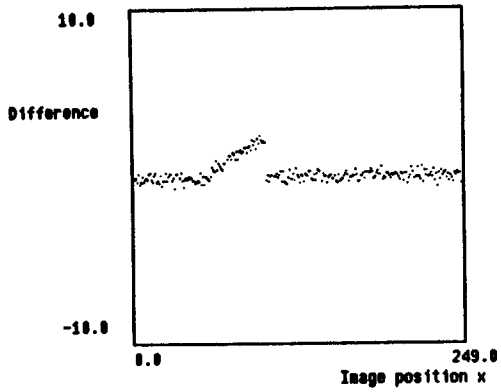

Fig. 8a Detection of a bump (5 % noise)

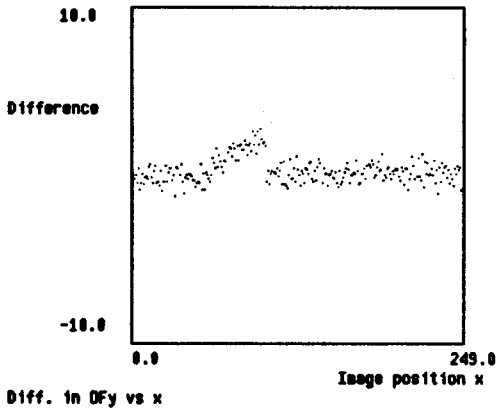

Fig. 8b Detection of a bump (10 % noise)

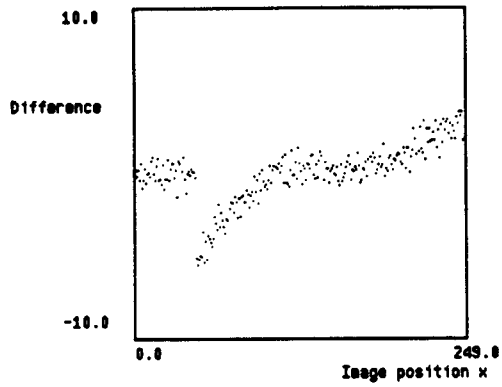

Fig. 9a Detection of a pothole and a 9^o ramp (10 % noise)

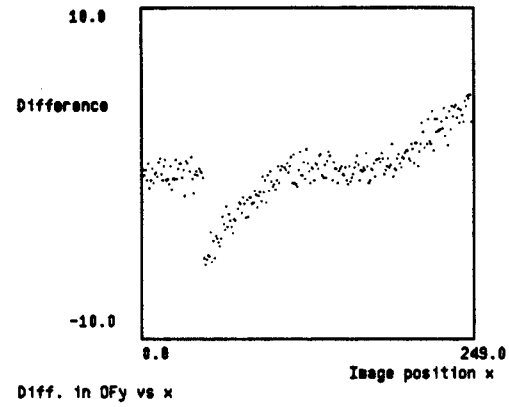

Fig. 9b Detection of a pothole and a 18^o ramp(10 % noise)

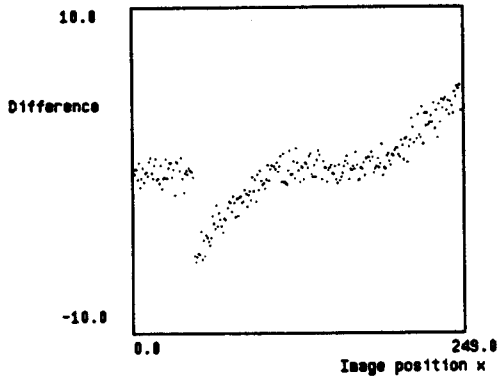

Fig. 9c Detection of a pothole and a 27^o ramp(10 % noise)

Actual slope angles	9°	18°	27°
Estimated slope angles	8.27°	15.9°	23.13°

Table 3 Results of ramp detection with 10 % noise