Experimental Investigation of Obstacle-Avoiding Mobile Robots without Image Processing

Experimental Investigation of Obstacle-Avoiding Mobile Robots without Image Processing

Hiroshi MASUDA*, Yoshio MIMURA*, and Yasuhisa OMURA*

(Received October 2, 2007)

Abstract

A simple method to detect step height, slope angle and trench width using four PSD range sensors (GP2D12) is proposed, and the reproducibility and accuracy of characteristic parameter detection are examined. The detection error of upward slope angle is within 2.5 degrees, while that of downward slope angle exceeded 20 degrees very large. In order to reduce such errors, a range sensor that has better range-voltage performance must be introduced, or we will have to increase trial frequency to prevent detection delay. Step height can be identified with an error of ± 1.5 mm. Trench width cannot be reliably measured at this time. It is suggested that an additional method is needed if we are to advance the field of obstacle detection.

Index Terms — obstacle classification, PSD range sensor, IR sensor, trench, slope, step

1. Introduction

In the last decade, autonomous mobile robots have been attracting a lot of attention and technical levels have dramatically advanced (see, for instance,[1]). Many robots providing entertainment, room cleaning and other services have already been developed[2]. To be truly practical, robots must be able to acquire environmental events and/or spatial information about its environment. Some robots for entertainment have optical sensors, ultrasonic sensors, touch sensors and other configurations. To create more autonomous robots that suit future applications, a 2-D range sensor[3] and a CMOS-imager camera[4] are being studied extensively. In these studies, sensor down-sizing is an on-going concern. However, the newly-developed sensors are still expensive, and computing overheads are apt to increase. This is a fundamental problem with the present research roadmap.

2-D path planning for mobile robots has also been studied extensively[5] ; it is thought that combining the path planning method[6] with the potential-field method[7] or a mapping technique would be a promising approach. These techniques are also needed for a future generation of self-learning robots.

This paper investigates how to detect and classify obstacles in front of a robot without a camera. The purpose of this paper is (1) to realize a robot that can detect the differences between step, slope and trench, (2) to form arithmetic procedures to estimate characteristic

* Graduate School of Science and Eng., Kansai University, 3-3-35 Yamate-cho, Suita, Osaka 564-8680, Japan.

values (step height, slope angle and trench width), and (3) to propose algorithms that yield reliable judgments. Four PSD IR range sensors are applied to the robot. Experiments on the robot challenge its sensor functions with steps, slopes and/or trenches.

The electrical or mechanical configuration of the robot is described in Section 2. Section 3 describes the measurement accuracy of the PSD range sensors used. Section 4 proposes algorithms that allow the robot to detect obstacles and estimate characteristic values. Section 5 describes the results of an obstacle-detection test and the reliability of obstacle recognition. Finally, some remaining issues are summarized.

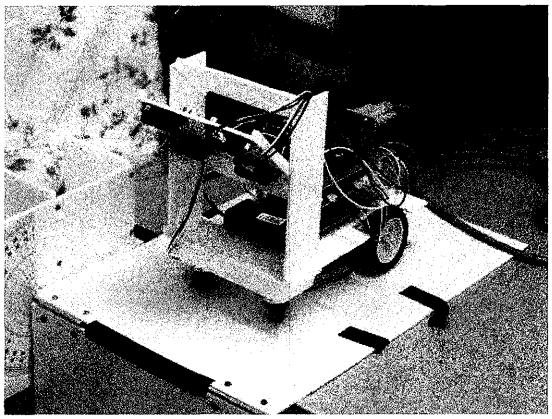

Fig. 1 Photo of an assembled robot for testing.

2. Mechanical and Electrical Architecture of the Robot

A picture of the prototype robot tested here is shown in Fig. 1. The robot has two non-driven caster wheels at the front and two motor-driven wheels at the back whose rotation speeds are controlled by a motor-drive circuit. The motor-driven wheels have four rotation modes (brake, stop, forward, and back). Since these four functions are implemented on the wheels independently, the robot can move in any direction. Four range sensors are placed on the front of the robot (PSD1L, PSD1R, PSD2L and PSD2R, respectively) to detect obstacles in front of the robot (See Fig. 2). These four sensors detect distances from the sensor to the floor, and the micro-controller calculates characteristic values for example, the slope angle θ when the obstacle is a slope.

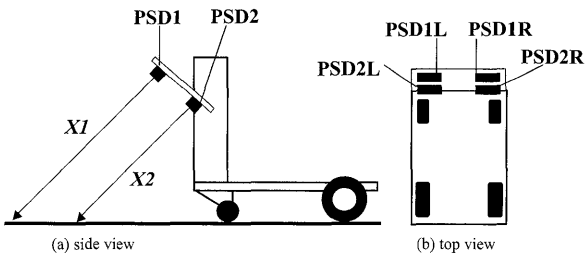

Fig. 2 Schematic of sensors' layout.

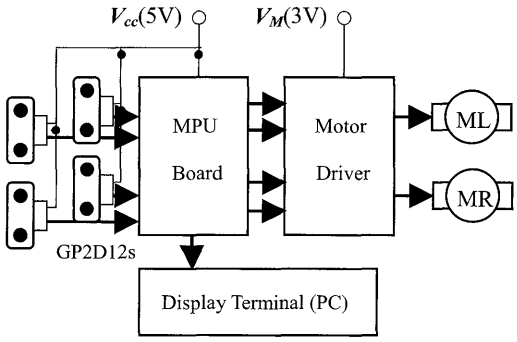

Fig. 3 Electronic control system for motor driver and others.

The electronic architecture of the robot is shown in Fig. 3. The circuit-mounted board includes a micro-controller (ADuC7026[8] produced by Analog Devices Corp.) to give the robot a data processing function. The micro-controller has input terminals for up to 12 single-ended A/D converters and other analog processing functions. The micro-controller receives analog signals from sensors through its built-in A/D converters, logically assigns the environment to one of the obstacles or no obstacle and finally outputs the characteristic value of the obstacle (slope angle θ for the slope, step height $h1$ for the step, and so on). The micro-controller on the MPU board calculates the obstacle's dimensions, and transfers the data to a PC via the RC-232 interface.

3. Accuracy and Reproducibility of Output Signal of PSD Sensor

We started by evaluating the potential of the IR range sensor (GP2D12[9] produced by SHARP Corp.) used to detect obstacles; we focus here on sensor performance attributes not described in the commercial data sheet. This sensor has the following features:

(1) The distance detection range (sensor to object) is 10 to 80 cm.
(2) The IR source signal of one sensor interferes very little with the functioning of the other sensors.
(3) The sensor is basically insensitive to object color and reflectivity.
(4) The sensor is basically insensitive to room light.
(5) Distance from the sensor to the floor can be detected even when the object surface is tilted. However, the variation in range is significant when the tilt angle is large.
(6) Low cost and small.

As just described, the PSD sensor has many advantages over other sensors. In some cases, There is a significant amount of electrical noise in the output signal, however, when we consider some applications that demand the detection of slope angle. This suggests that we have to examine how accurately the sensor detects distance from the sensor to the object (denoted by $X1$ and $X2$ for two sets of sensors, respectively) before we can design an accurate sensor circuit.

As an example, we show range data created by transforming the analog signals of the PSD sensors in Figs. 4 and 5; Fig. 4 shows the output of the micro-controller when challenged with an 18-mm-high upward step, and Fig. 5 shows that for a 20-degree downward slope. In both cases, the robot had a constant velocity on the floor. In Fig. 4 and Fig. 5, the thin lines are the unprocessed digital range data transferred from the micro-controller, while the bold lines are the range data after being passed through a median filter (window number $N1$= 5) (See Appendix I).

Fig. 4 shows that the median filter is effective in removing the impulse noise. It also shows that the filter yields a time delay, resulting in a 5-mm local position difference in the case of $N1$ = 5. The noise can be further reduced by increasing $N1$, but at the cost of simultaneously increasing the time delay. Because of this trade-off, it's preferable to adjust $N1$ to suit the application.

In Fig. 5, the impulse noise is sufficiently removed at short distances (as well as in Fig. 4),

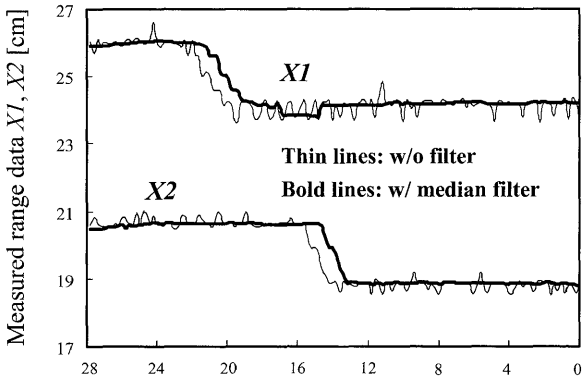

Fig. 4 Range data evolution when the robot is approaching an upward step.

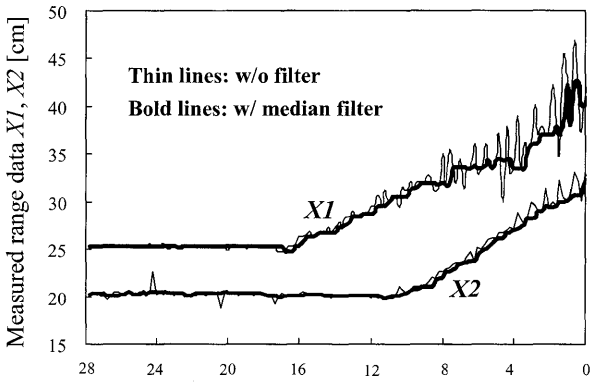

Fig. 5 Range data evolution when the robot is approaching a downward slope.

but not at distances beyond 50 cm. This is due to the sensor's performance limitation [9] ; when *X1* (> 50 cm), even a small voltage shift of output signal of sensor results in a large variation in range data. When the angle between the IR-light beam from the sensor and the object surface increases, the IR signal returned attenuates, and the influence of room light becomes significant. This means that a downward slope yields a large variation in detected signal.

4. Method of Extracting Spatial Values

In Section 4.1, we describe how the robot identifies steps, slopes, and trenches using the upper and lower sensors (PSD1 and PSD2). Section 4.2 describes the mathematical model that the robot applies to calculate step height, slope angle or trench width. Section 4.3 details the results of experiments on the robot's determination of step height, slope angle and trench width.

In this chapter, we assume that the robot directly faces the obstacle (the width of which is taken to be effectively infinite). Note that all the range data (*X1L, X1R, X2L,* and *X2R*) displayed in the figures are the result of median filtering. Results obtained assuming more practical situations are shown in Section 5.

4.1 How to classify slopes, steps and trenches

First, we should explain the notations used in this section. *X1* and *X2* stand for the distances given by PSD1 and PSD2, respectively. When the robot runs on a flat floor, it is assumed that PSD1 and PSD2 yield distance data *X1o* and *X2o*, respectively. In a practical situation, we have to take account of noise in the data yielded by the sensors. Accordingly, we introduce positive threshold values of *X1T* and *X2T* to improve the detection reproducibility of distance data when determining whether the event (i. e., slope, step, or trench) has occurred. When PSD1 outputs data satisfying the condition of $|X1-X1o|<X1T$, the robot 'thinks' that he is on a flat floor. In this case, we say that S (PSD1) = "Flat". When PSD1 outputs data satisfying the condition of $|X1-X1o|> X1T$, the robot 'ethinks' that he may be facing a slope, a step, or a trench. In this case, we say that S (PSD1) = "NON-F". In the present experiment, we empirically set $X1T$=0.8 [cm] and $X1T$=0.5 [cm] by taking account of the noise level shown in Figs. 4 and 5, respectively. For example, when the robot is running on a flat floor, the "states" output from the 4 sensors are "flat", and we use the following descriptions:

S(PSD1L) = "Flat", (1)
S(PSD1R) = "Flat", (2)
S(PSD2L) = "Flat", (3)
S(PSD2R) = "Flat". (4)

Next, we describe how the robot uses the geometric method shown in Fig. 6 to differentiate slopes, steps, and trenches.

(a) Flat floor

When the robot compares the range data to the threshold value given in the previous Section, and PSD1 and PSD2 output data satisfying the condition of S (PSD1) ="FLAT" & S (PSD2) ="FLAT", the robot 'thinks' that he is on a flat floor (see Fig. 6(a)). The equivalent mathematical relationship can be expressed as

$$|X1 - X1o| \leq X1T \text{ and } |X2 - X2o| \leq X2T. \quad (5)$$

(b) Downward step

When PSD1 and PSD2 output data satisfying the following condition, the robot 'thinks' that he is facing a downward step (see Fig. 6(b)).

$$X1 - X1o > X1T \text{ and } X2 - X2o > X2T \text{ and } |X1 - X1o| - |X2 - X2o| < X1T + X2T. \quad (6)$$

(c) Upward step

When PSD1 and PSD2 output data satisfying the following condition, the robot 'thinks' that he is facing an upward step (see Fig. 6(c)).

$$X1 - X1o \leq -X1T \text{ and } X2 - X2o \leq -X2T \text{ and } |X1 - X1o| - |X2 - X2o| X1T+X2T. \quad (7a)$$

or

$$|X1 - X1o| \leq X1T \text{ and } X2 - X2o \leq -X2T. \quad (7b)$$

(d) Trench

When PSD1 and PSD2 output data satisfying the following condition, the robot 'thinks' that he is facing a trench (see Fig. 6 (d)). In this case, the PSD sensor receives the IR signal reflected from the side wall of the trench or the IR signal reflected from the bottom of the trench as shown in Fig. 7 (c). The algorithm in its simple present form cannot distinguish these two cases.

$$|X1 - X1o| \leq X1T \text{ and } X2 - X2o > X2T. \quad (8)$$

(e) Downward slope

When PSD1 and PSD2 output data satisfying the following condition, the robot 'thinks' that he is facing a downward slope (see Fig. 6 (e)).

$$|X1 - X1o| > X1T \text{ and } |X2 - X2o| > X2T \text{ and } (X1 - X1o) - (X2 - X2o) > X1T + X2T. \quad (9)$$

(f) Upward slope

When PSD1 and PSD2 out data satisfy the following condition, the robot 'thinks' that it is facing an upward slope (see Fig. 6 (f)).

$$|X1 - X1o| > X1T \text{ and } |X2 - X2o| > X2T \text{ and } (X1 - X1o) - (X2 - X2o) < X1T + X2T. \quad (10)$$

4.2 Equations to calculate slope angle, step height, trench width

Slope angle, step height and trench width can be calculated from range data $(X1, X2)$. Fig. 7 illustrates the geometric techniques used.

(a) Step height

Step height $h1$ is calculated using Eq. (11). A diagram is shown in Fig. 7 (a).

$$h1 = (X2o - X2) \sin\alpha, \quad (11)$$

where α is the angle of sensor signal against a flat floor (here, $\alpha = 45$ deg.). When $h1 < 0$, a downward step is suggested, and when $h1 > 0$, an upward step is suggested.

(b) Slope angle

Slope angle θ is calculated using Eq.(12). A diagram is shown in Fig. 7(b).

$$\theta = p/2 - \alpha - \tan^{-1}((X1 - X2)/L). \quad (12)$$

When $\theta < 0$, a downward slope is suggested, and when $\theta > 0$, an upward slope is suggested.

(c) Trench width

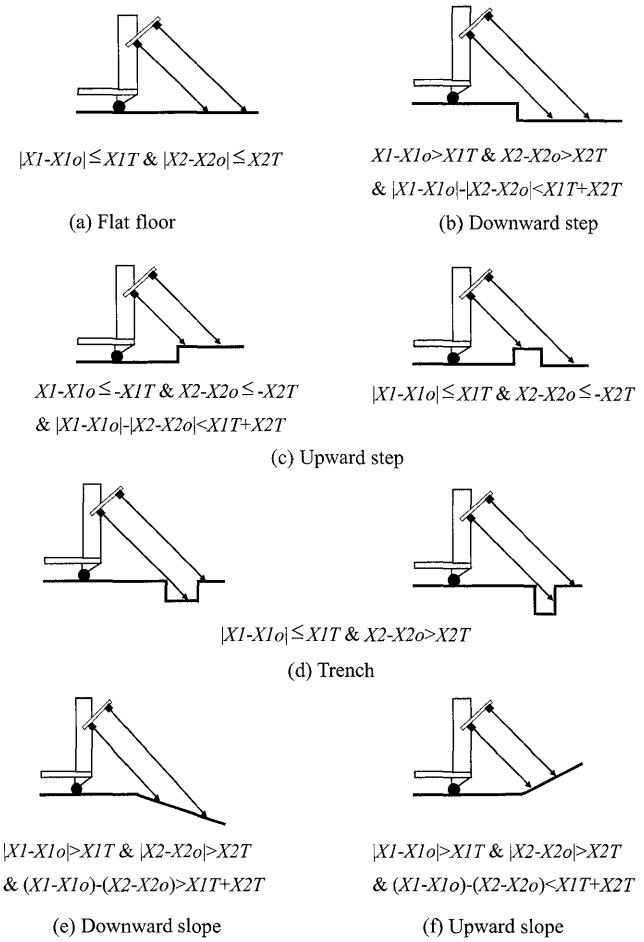

Fig. 6 How to classify slopes, steps and trenches. Mathematical algorithms are shown.

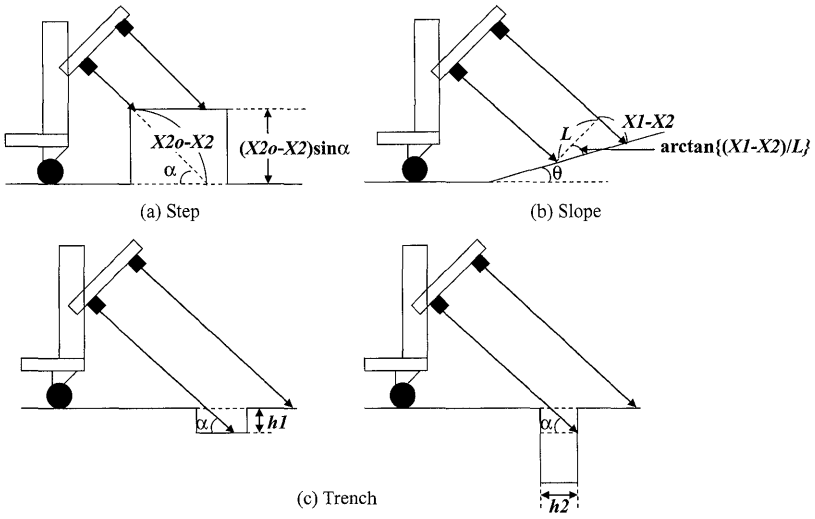

Fig. 7 How to classify slopes, steps and trenches. The characteristic parameter extraction method is shown.

When the robot faces a trench, sensor output may be one of two cases; (i) the sensor signal is reflected from the bottom of the trench, (ii) the sensor signal is reflected from a side wall of the trench. Initially, the robot cannot judge which is correct. Our solution is to force the robot to calculate two dimensions, trench depth and trench width. Trench depth *h1* is calculated using the following equation:

$$h1 = (X2o - X2) \sin\alpha. \tag{13}$$

Trench width *h2* is calculated with the next equation:

$$h2 = (X2o - X2) \cos\alpha. \tag{14}$$

Since all sensors are positioned so that their surfaces are angled at 45° against a flat floor, the calculated values of *h1* and *h2* are identical. When the robot approaches the trench, it's judgment of whether or not it can cross the trench depends on the diameter (D_w) of the robot's wheels. Consider case (i). When D_W is much larger than the trench depth, the robot may be able to cross the trench. Consider case (ii). When D_W is much larger than the trench width, the robot can go over the trench. Therefore, the robot can pass through the trench in both cases, (i) and (ii), when *h1* or *h2* is much smaller than D_W. In other words, it is not necessary for us to distinguish cases (i) and (ii); we can apply Eq.(13) to decide whether the robot can go forward or not when the robot detects a trench.

In practical applications, it is not always possible to derive precise characteristic values in order to use the above equations because of various noises (including external disturbance) and/or spatial dispersion of the emitted IR signal. This suggests the need for an additional method to guarantee the accuracy of the characteristic values and judgment reliability; details will be given in Section 5 below.

4.3 Measurement results: step height, slope angle, and trench width

Measurement results of step height for which the robot should stop in front of the step are summarized in Table I. 1000 sensing trials were averaged in each event of obstacle discovery and the medial filter number (*N1*) was 5. As is evident in Table I, the variation of evaluated step height *h1* is very small; the difference between the maximal value and the minimal value is about 3 mm for the upward step and about 6 mm for the downward step. We can see that the present evaluation technique does not always yield accurate data.

Measurement results of upward slope angle (θ) are shown in Table II. 1000 sensing trials were averaged in each event of obstacle discovery and the medial filter number was 5 or 21. In the experiment, we assumed 3 cases of horizontal distance (*d1*) between the front edge of the robot and the boundary of the flat floor and the slope 0 cm, 4 cm, and 7 cm. The slope angles were 20 deg., 15 deg. and 10 deg. It can be seen from Table II that the averaged value of θ increases with *d1*. In this study, the slope angle evaluation algorithm does not estimate distance *d1*, and so the robot estimates characteristic values without stopping as it approaches the obstacle, resulting in a slight drop in accuracy. It is also evident from Table II that a large N1 value reduces the variation in estimated values, although a large N1 value results in a longer time before judgment. In addition, we note that for N1=21, the difference between

Table I. Step height evaluation results in units of cm. 1000 sensing trials are averaged. The medial filter number (*N1*) is 5. The real step height (*h1*) is 1.30 cm.

	Upward step height [cm]	Downward step height [cm]
Mean value	1.30	-1.33
Max. value	1.45	-1.00
Min. value	1.18	-1.56
Variance	0.0057	0.0019

Table II. Upward slope angle evaluation results in units of cm. 1000 sensing trials are averaged. The medial filter number (*N1*) is 5 or 21.

Median filter number (*N1*)	Real slope angle [deg.]	*d1* [cm]	Mean value [cm]	Max. Value [cm]	Min. Value [cm]	Variance [cm]
5	20.0	0.00	17.00	19.43	15.87	0.4648
		4.00	18.90	21.00	17.49	0.5608
		7.00	20.47	22.15	17.39	1.0129
	15.0	0.00	14.70	16.50	13.01	0.2176
		4.00	15.29	16.18	14.24	0.1210
		7.00	16.09	18.73	12.80	1.8943
	10.0	0.00	9.68	12.05	7.74	0.2426
		4.00	10.29	13.39	7.84	1.3552
		7.00	10.82	13.18	9.26	0.1617
21	20.0	0.00	16.90	18.11	16.46	0.0951
		4.00	19.69	20.48	18.85	0.1100
		7.00	19.25	22.15	18.44	0.2841
	15.0	0.00	14.57	15.32	12.96	0.1041
		4.00	14.76	17.58	13.54	0.8732
		7.00	15.46	16.32	14.83	0.1287
	10.0	0.00	9.80	10.48	9.11	0.0630
		4.00	9.89	10.66	8.21	0.1137
		7.00	9.40	11.48	8.59	0.2549

Table III. Downward slope angle evaluation results in units of cm. 1000 sensing trials are averaged. The medial filter number (*N1*) is 21.

Real slope angle [deg.]	*d1* [cm]	Mean value [cm]	Max. value [cm]	Min. value [cm]	Variance [cm]
-20.0	10.00	-19.91	-13.72	-23.23	2.0619
	5.00	-18.90	-10.42	-25.41	7.6875
	2.00	-18.37	-10.35	-27.16	10.2642
-15.0	10.00	-14.56	-10.12	-18.02	1.4096
	5.00	-14.23	-10.02	-21.70	3.7594
	2.00	-14.46	-10.04	-19.76	4.3537
-10.0	10.00	-10.58	-7.55	-13.66	1.4499
	5.00	NA	NA	NA	NA
	2.00	NA	NA	NA	NA

the maximal value and the minimal value is not always reduced.

Table III shows measurement results of downward slope angle. As is evident in Table III, the variation of measurement results is very large in contrast to the upward slope values. This suggests a need to improve judgment reliability for practical applications.

5. Dynamic Detection of Obstacles in a Test Road

In sections 5.1 to 5.4, we describe an algorithm to be used in practical situations. Section 5.5 details the accuracy of several evaluations.

5.1 Logical Flow

A schematic flow showing how the robot avoids obstacles is shown in Fig. 8. First, when the left or the right sensor state "NON-F", the robot changes it's position so that the front edge of the robot remains parallel to the border line of the obstacle and the flat floor. Next, the robot approaches the border line, again detects signals from the obstacle, and subsequently concludes whether the obstacle facing it is a slope, step or trench. Finally, when the robot recognizes that the obstacle is a step, it calculates the tentative step height, compares the calculated value to the threshold value, and then concludes whether it has to avoid the obstacle or not. When the robot detects a slope or a trench, the robot traces the same logical flow. As just described, in order to classify the obstacle successfully and to get reliable characteristic values, the causes of errors in detecting the signals from the obstacle must be analyzed.

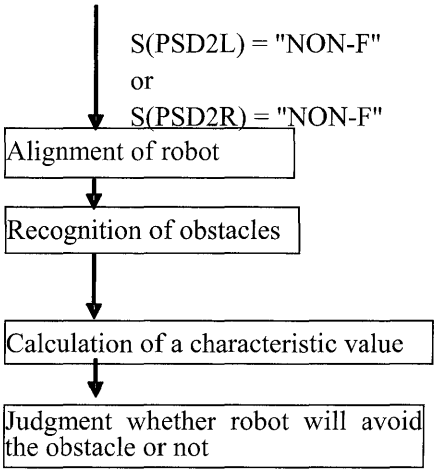

Fig. 8 Algorithm for the robot in order to detect an obstacle and avoid it.

5.2 Aligning the robot to the obstacle

In this section, we describe how the robot positions himself in the vicinity of the obstacle. For all obstacles, to maximize detection accuracy, the robot should directly face the obstacle. This process is detailed below (see Fig. 9).

(1) First, the robot approaches the obstacle, receives range data, and examines whether the

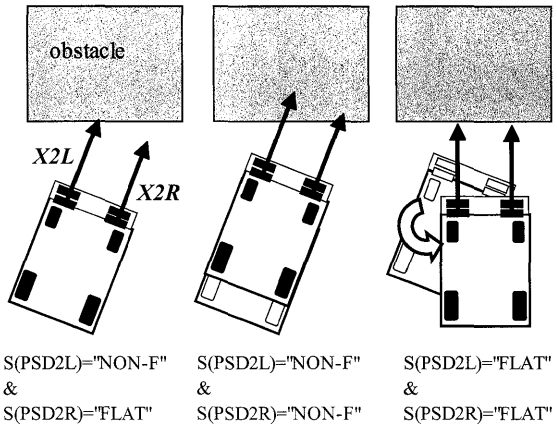

Fig. 9 Front edge alignment of the robot facing the obstacle.

data satisfies the condition S(PSD2L) = "NON-F" Λ S(PSD2R) = "FLAT" (see Fig. 9(a)).

(2) The robot moves forward slightly, again receives range data, and examines whether the data satisfies the condition S(PSD2L) = "NON-F" Λ S(PSD2R) = "NON-F" (see Fig. 9(b)).

(3) When both PSD2L and PSD2R detect "NON-F" signals, the robot moves as follows (see Fig. 9(c)).

(i) When S(PSD2L) = "NON-F" for example, the left motor reverses.

(ii) When S(PSD2L) = "FLAT", the left motor idles.

(iii) When S(PSD2R) = "NON-F", the right motor reverses.

(iv) When S(PSD2R) = "FLAT", the right motor idles.

The above algorithm ensures that the robot directly faces the obstacle.

(4) When both motors stop, the algorithm has successfully terminated and the robot approaches the obstacle again.

In the detecting process from (1) to (2), when the time interval of "NON-F" events of two sensors is longer than a certain value, the robot turns around before reaching the expected obstacle. In other words, the robot doesn't estimate the vertical offset when the incident angle is very small.

The present algorithm yields a small degree of uncertainty as to the robot's alignment, due to the use of the threshold values *X1T* and *X2T*. In this experiment, we found alignment error of up to 10°. Later, we evaluate the influence of this alignment error on the determination of characteristic parameters.

5.3. How to classify obstacles using sensor pairs

First, we explain how to classify slopes, steps and trenches (see Fig. 8). Using the simple method described in Section 4, the robot may, for example, incorrectly classify a real slope as a step or a flat floor. This erroneous judgment comes from sensor noise and relatively large threshold values (*X2T* and *X1T*). When these threshold values are large, erroneous judgment becomes more common. To avoid this difficulty, we make the robot calculate characteristic values repeatedly, to get the mean or median value. This flow is described below.

(1) When the robot detects an object, it calculates the characteristic values 20 times, and

stores this data in memory.

(2) The robot classifies the obstacle according to the highest frequency of classification after the 20 trials.

(3) When the frequency of "trench" exceeds 5 out of 20 trials, the robot classifies the obstacle as a trench.

We have confirmed that this majority-decision process reduces the frequency of erroneous judgment.

5.4 How to calculate the characteristic values of a specific obstacle

Here we describe a method for calculating the characteristic values.

(1) When upward (or downward) step height *h1* is calculated, the robot calculates the mean of 20 trials.

(2) When upward (or downward) slope angle θ is calculated, the robot calculates the mean of 20 trials.

Next, when the robot stops, the robot gets the data set *X1L, X1R, X2L,* and *X2R* 200 times. After determining the mean values of *X1L, X1R, X2L,* and *X2R,* they are labeled *X1L', X1R', X2L', and X2R',* respectively. Finally, using values of *X1L', X1R', X2L',* and *X2R'* and Eq. (9), the robot calculates the downward-slope angle θ. This technique is very effective for suppressing noise (as described in sections 3 and 4). This benefit incurs the cost of a 5-sec. delay in determining the downward-slope angle.

5.5 Evaluation results of characteristic values

Tables IV to VI show the characteristic values yielded by the logical process described in the previous section.

Table IV shows slope angle values extracted from signals given by sensors mounted on the robot; the offset value of sensor signals is considered in calculating the characteristic values, and the robot logically determines which obstacle has been encountered. As a result, the robot showed very few errors in classification of obstacles. Erroneous judgment, however, sometimes takes place when dealing with a gentle slope, which depends on threshold values of *X1T* and *X2T*. A gentle slope sometimes gives the sensor a noisy signal that cannot be easily detected as meaningful data; in this case, the robot fails to correctly determine the slope angle. Raising the values of *X1T* and *X2T* yields more conclusive data at the cost of limiting the detectable range of slope angle.

Next, we discuss the accuracy of extracted slope angle θ. In the present experiment, the detectable range of slope angle θ is 20° to -10°, and the deviation of extracted slope angle is at most 2.5°. At θ = -20°, however, the uncertainty in detected angle rises to 4°. When the angle of the sensor-light incident on the object's surface becomes small, the intensity of the reflected-light signal becomes very weak. This results in a lower dynamic range in the sensor's output signal; basically the same phenomenon described in section 3.

Since the sensor emits an infra-red light signal, the reflection rate of the light depends on the color of the object's surface. In addition, the sensor's output attenuates as the distance of the sensor from the object increases. When the color of the object is dark, the sensor's output falls, as does its dynamic range. As a result, it is usually difficult to detect accurately the

Table IV. Successful trials in detecting slopes. In detection of the slope angle, detected data are averaged from 100 trials.

Real slope angle [deg.]	Successful detection rate	Mean value [cm]	Max. value [cm]	Min. value [cm]	Variance [cm]
-20.0	100/100	-20.00	-18.26	-24.06	0.9255
-10.0	100/100	-12.71	-11.44	-13.98	0.2614
10.0	100/100	10.27	10.91	9.52	0.0748
20.0	100/100	21.35	22.26	20.09	0.1678

Table V. Successful trials in detecting steps. In detection of the step height, detected data are averaged from 100 trials.

Real step height [cm]	Successful detection rate	Mean value [cm]	Max. value [cm]	Min. value [cm]	Variance [cm]
1.30	100/100	1.37	1.47	1.17	0.0017
-1.30	100/100	-1.21	-1.11	-1.33	0.0022

Table VI. Successful trials in detecting trenches. In detection of the trench width, detected data are averaged from 100 trials.

Real trench dimension	Successful detection rate	Mean value [cm]	Max. value [cm]	Min. value [cm]	Variance [cm]
1.30[cm] x 1.30[cm]	55/100	-0.94	-0.85	-1.26	0.0098
2.00[cm] x 2.00[cm]	90/100	-1.67	-1.24	-2.32	0.0310

angle of a steep slope. Two possible ways to remove this difficulty are: to use a PSD sensor, whose distance-output-voltage characteristic is almost linear, and to widen the window of the median filter, although this would increase the detection time.

Table V shows step height values that are recalculated by the sensor module with some offset angle when the robot approaches the step. Erroneous detection did not occur. The detected step height was more accurate than that described in Section 4, where the signal-filtering technique was applied to the step-height detection. When the maximal step height that forces the robot to back away is 13 mm, the maximal value of calculated step height should be 11 mm, because the maximal variation in the output voltage signal of the sensor is equivalent to the step height of 2 mm.

Table VI shows trench-width values determined by the robot; the offset value of sensor signals was considered in calculating the characteristic values.

The basic algorithm used to detect a trench was described in section 3. When the robot approaches the trench at an oblique angle (see Fig. 10), the correction of trench depth, used in the algorithm described in section 5.1, cannot be employed because the algorithm assumes a direct approach to the trench. In this experiment, the robot was limited to approaching the trench at nearly 90 deg. This experiment was made on two trenches of different sizes.

As is seen in Table VI, when trench width is reduced, the frequency of erroneous judgment rises. One cause is incompleteness of the detection algorithm; the robot incorrectly judges the trench as a flat floor. A way of overcoming this difficulty is to reduce the values of *X1T* and *X2T*. Another approach is to increase the diameter of the wheels.

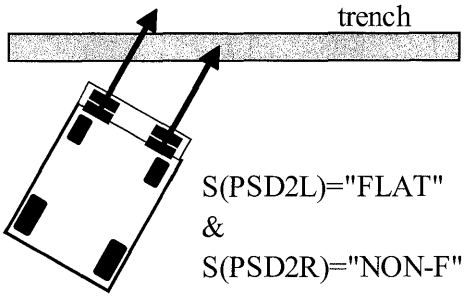

Fig. 10 How to detect a trench.

6. Concluding remarks

We have proposed a simple method for detecting step height, slope angle and trench width using four PSD range sensors (GP2D12), and have examined the reproducibility and accuracy of characteristic parameter detection. Detection error of upward slope angle is about 2.5°, while the detection error for downward slope angles exceeding 20° is very large. To reduce these errors, we will have to use a range sensor that offers better range-voltage performance, or we could increase the trial frequency so as not to increase the detection delay. Step height is extracted with an error of ±1.5 mm. The current algorithm for trench width is quite accurate. An additional method must be introduced to advance the obstacle detection technique.

Appendix I: Median filter algorithm

In this paper, we used the following algorithm in order to reduce the electrical noise in the original signal. First, we get N datum points from the micro-controller. After sorting the N data ($D[1]$ to $D[N]$), we extract the maximal value, Dmax, and the minimal value, Dmin, from all data, and order the data set (n= 1 to N) ; i. e., $D[1]= Dmax$ and $D[N]= Dmin$. Finally, we get $D[N/2]$ as the median value. Sets of $D[N/2]$ are plotted in Figs. 4 and 5.

References

1) J. Velagic, B. Lacevic, and B. Perunicic, "A 3-level autonomous mobile robot navigation system designed by using reasoning/search approaches," *Robotics and Autonomous Systems*, Vol. 54, pp. 989-1004, 2006.

2) T. Miyake, H. Ishihara, R. Shoji, and S. Yoshida, "Development of small-size window cleaning robot with a traveling direction control on vertical surface using accelerometer," *IEEE International Conference on Mechatronics and Automation.* (Luoyang, 2006), pp. 1302-1307.

3) H. Kawata, A. Ohya, S. Yuta, W. Santosh, and T. Mori, "Development of ultra-small lightweight optical range sensor system," *IEEE/RSJ International Conference on Intelligent Robots and Systems.* (Luoyang, 2005), pp. 1078-1083.

4) P. Gemeiner and M. Vincze, "Motion and structure estimation from vision and inertial sensor

data with high-speed CMOS camera," *IEEE International Conference on Robotics and Automation.* (Barcelona, 2005), pp. 1853-1858.

5) H. Zhang, S. Liu, and S. X. Yang, "A hybrid robot navigation approach based on partial planning and emotion-based behavior coordination," *IEEE/RSJ International Conference on Intelligent Robots and Systems.* (Beijing, 2006), pp. 1183-1188.

6) D. F. Wolf, G. S. Sukhatme, D. Fox, and W. Burgard, "Autonomous terrain mapping and classification using Hidden Markov Models," *IEEE International Conference on Robotics and Automation.* (Barcelona, 2005), pp. 2026-2031.

7) D. H. Kim and S. Shin, "Local path planning using a new artificial potential function configuration and its analytical design guidelines," *Advanced Robotics*, Vol. 20, No. 1, pp. 115-135, 2006.

8) http://www.analog.com/UploadedFiles/Data_Sheets/ADUC7019_7020_7021_7022_7024_7025_7026_7027.pdf

9) http://www.acroname.com/robotics/parts/SharpGP2D12-15.pdf