The 23rd International Technical Conference on Circuits/Systems,
Computers and Communications (ITC-CSCC 2008)

# Monocular Vision based Indoor Mobile Robot

Naoya Tada, Keisuke Murata, Takeshi Saitoh, Tomoyuki Osaki and Ryosuke Konishi
Department of Information and Electronics, Tottori University
4-101 Koyama-minami, Tottori 680-8552, Japan
E-mail: {saitoh, konishi}@ele.tottori-u.ac.jp

**Abstract**: This paper presents the indoor mobile robot that moves automatically without needing environment information beforehand while recognizing the frontal surrounding environment with only one general camera. Based on the frontal image, the robot detects two boundary lines, some obstacle regions, and a moving direction. When the obstacle is detected, the avoidance or stop movement is worked according to the size and the position of the obstacle, otherwise the robot moves at the center of the corridor. We developed two wheelchair based mobile robots, and carried out moving experiments. It was possible to pass each other without colliding by working the avoidance movement in face-to-face movement in the coexistence environment.

## 1. Introduction

Recently, there has been a number of works concerned with the indoor mobile robot [1], [2], [3], [4]. For instance, the robot which traces a course line drawn on the floor, detects some landmarks on the wall or floor, traces generated route based on the map information. However, these robots need the environmental information in advance. The robot using omni-directional camera is developed and its can analysis the moving environment in real-time. As this robot, though this robot has an advantage that all surrounding environments can be obtained, it has the placement problem of camera that is necessary to mount on the top of robot to take all round view, and it causes the limitation in appearance.

This research develops the indoor mobile robot that moves automatically without needing environment information beforehand while recognizing the frontal surrounding environment with only one general camera. Based on the frontal image, calculate the moving direction and detect the obstacle. When the obstacle is detected, the avoidance or stop movement is worked according to the size and the position of the obstacle, otherwise the robot moves at the center of the corridor, called center following movement.

## 2. Approach

The approach presented here is as follows roughly. The frontal view information is obtained by using the color imaging camera which was mounted in front of the robot, as shown in figure 1. Then two boundary lines of the wall and corridor are detected. As the space between two boundary lines, we apply the appearance based obstacle detection method. When obstacle is detected, the size and place of obstacle is calculated, and then, the robot works the avoidance or stop movement according to the information of obstacle. Otherwise, it moves toward the center of the corridor automatically.

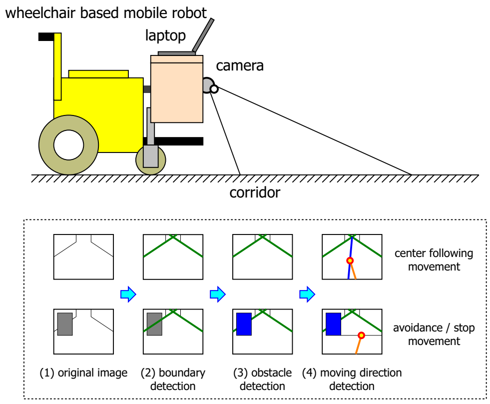



Figure 1. Overview of our mobile robot.

## 3. Moving direction detection

### 3.1 Boundary detection

To detect boundary line has the effect to the reduction of the analyzing area of obstacle detection, and the calculation of the moving direction such as the vanishing point. Figure 2 shows two frame images obtained by the camera. Generally, there is a baseboard at the bottom of the wall, and this color is darker than the wall and corridor. This characteristic is useful for boundary detection. Then, this paper focused on this baseboard.

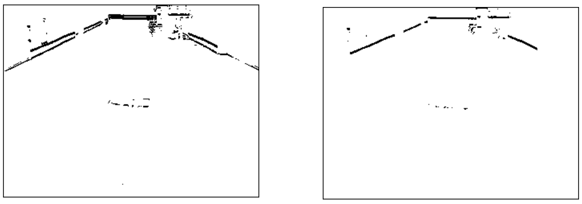
#### 3.1.1 Edge detection

Figure 3(a) shows the applied result of Sobel vertical edge detector to figure 2(a). Since the height of the baseboard is about 10cm, this result obtained two edges, one is the edge between the wall and baseboard, the other is the edge between the corridor and baseboard. As for a general corridor, because wax is painted, reflectivity is higher than walls. Then the image, such as the baseboard, the door, and the light source, reflects in corridor region, and the edge between the corridor and baseboard is unclear. Oppositely, the edge between the wall and baseboard is clear. Hence, we detect the edge between the wall and baseboard as the boundary.

First of all, the binary edge pixel (called the temporary edge pixel) is detected by using Sobel vertical edge detector. Only when the density value of the temporary edge pixel is lower than that of the upside pixel, it is assumed the desired edge pixel. This is based on the fact whose density value of the wall is higher than the baseboard. Figure 3(b) shows the applied result of our method to figure 2(a). Only the desired edge in the upper part of the baseboard has been detected. In addition, the detected edge pixel can be few, and then, the speed-up of processing time for Hough transform which describes in the next paragraph is brought.

(a) (b)

Figure 2. Original images.



(a) Applied result of Sobel vertical edge detector. (b) Detected result of the edge between the wall and baseboard.

Figure 3. Binary edge images.

(a) (b)

Figure 4. Detected boundary lines.

### 3.1.2 Boundary line detection

To detect two boundary lines and a point where a left-right boundary line crosses is detected as a vanishing point, we apply not a normal Hough transform but a piece wise linear Hough transform to reduce the processing time. This method was proposed by Koshimizu and Numada, and this method achieves the high speed processing by representing the Hough curve as piece wise linear approximation in Hough plane [5].

However, the actual boundary lines are sometimes hidden by the obstacle. In this case, the edge pixels of boundary lines reduced, and it causes to obtain wrong lines. To solve this problem, we use previous lines information. Here, $\theta_f$ and $\rho_f$ denote the obtained parameters using Hough transform at frame $f$. When either $|\theta_f - \theta_{f-1}| > 3$ deg. or $|\rho_f - \rho_{f-1}| > 10$ pixel is satisfied, we assume obtained boundary is wrong, and we set $\theta_f = \theta_{f-1}$ and $\rho_f = \rho_{f-1}$. Figure 4 shows the detected boundary lines overlaid on the original image.

## 3.2 Obstacle detection

As the space between two boundary lines, namely, the detected corridor region, we apply the obstacle detection method.

### 3.2.1 Appearance based obstacle detection method

To detect obstacle, Ulrich et al. proposed the appearance based method which is used the color histogram [6]. This method sets a reference area at the bottom of image, and assumes that the color of obstacle differs from this area. Then, any pixel that differs in appearance from this area is classified as obstacle region. This method is based on three assumptions, 1) obstacles differ in appearance from the corridor, 2) the corridor is relatively flat, and 3) there are no overhanging obstacles.

We first apply Ulrich's method to detect obstacle. Here, we set a trapezoidal area the bottom of image as shown in the left image of figure 5. For each image, we convert RGB (red, green, and blue) color space into HLS (hue, lightness, and saturation) color space. Then, we generate a histogram of L. All pixels in the target region are compared to this histogram. A pixel is classified as obstacle region if histogram bin value at the pixel's L is below the threshold $T_h$. The applied results of this method are shown in the second row of figure 5. As these results, we obtained wrong obstacle regions. The light source reflects on the corridor, and a lot of high luminance pixels detected, and these have been mis-detected as obstacle region.

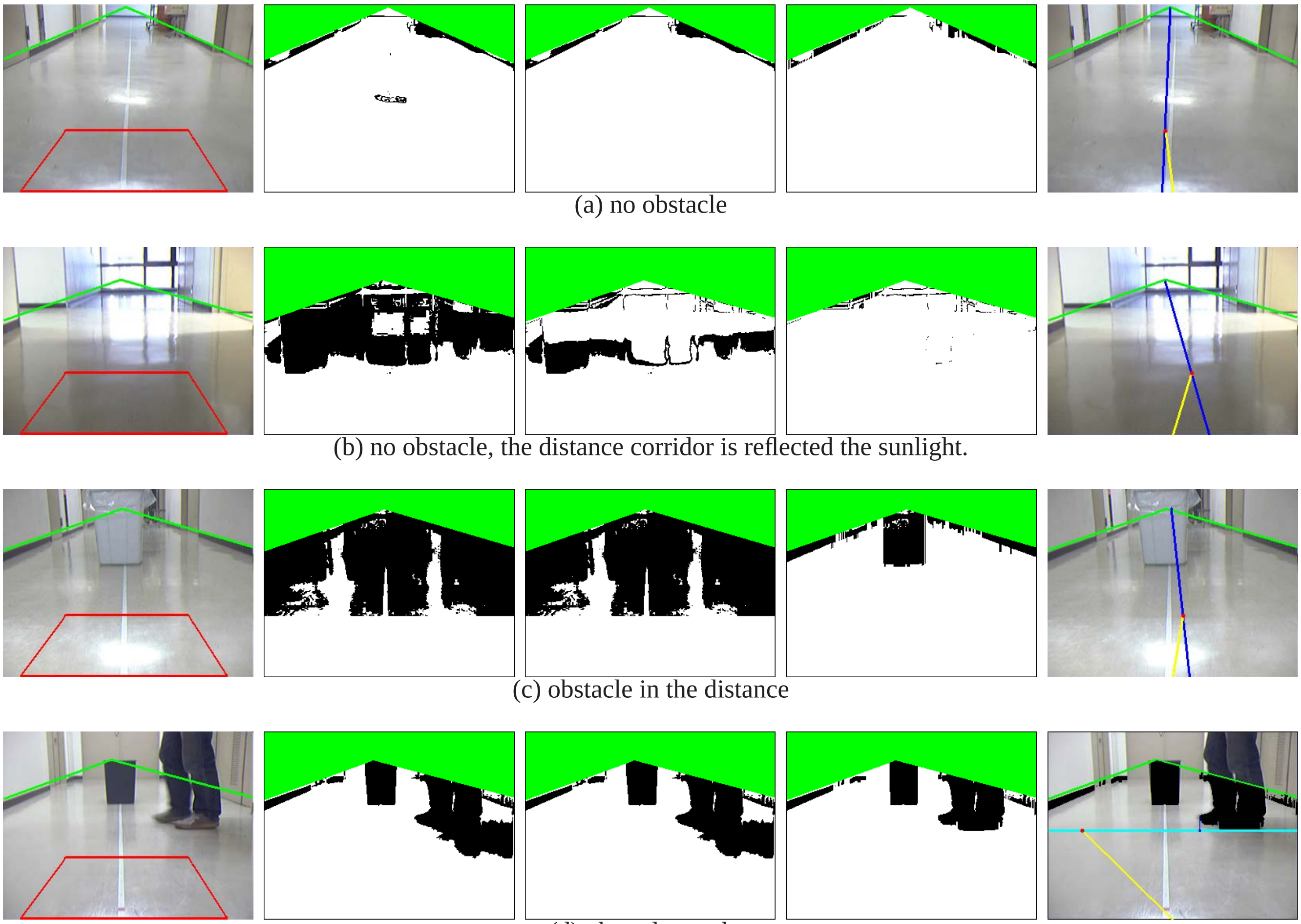
### 3.2.2 Proposed obstacle detection

Our system goes straight at the center of the corridor when the obstacle does not detected, and when the obstacle is detected, applies the avoidance movement.

To remove the reflected pixel of the light source, first of all, the high luminance pixel is removed. In particular, when the L value is grater than 0.7, that pixel belongs to the corridor region even if not included in the reference area. The result is shown in the third row of figure 5. By observing figure 5(a)(b), it can be confirmed that the lighting region has been removed compared with the Ulrich's method of the second row.

The pixel of the intermediate value as shown in figure 5(b), remains as obstacle region only by removing high luminance. Furthermore, the region that has darkened gradually has been mis-detected as obstacle region as shown in figure 5(c)(d) in case of no high luminance. To solve these problems, we use two characteristics, one is that the edge appears in the boundary of the obstacle and corridor, the other is the false detection is occurred easily to the dark color obstacle, which is not removed with edge information. As for the first characteristic, because there is roundness in a part of the edge, the object whose surface is almost flat, such as shoes and box, put on the corridor is detected as edge. On the other hand, as long as the corridor is not all-reflective material such as mirror, the reflected edge is blurred. Then, we use edge information by Sobel edge detector. As for the second characteristic, we pay attention not the edge but the color of object, and even if the edge is not detected, that pixel belongs to the obstacle.

The process flow of proposed method is as follows. At first, Ulrich's method is applied to detect temporary obstacle. Next, when the pixel detected as the temporary obstacle is not an edge or a dark pixel, it classifies into the corridor region. This classification process is scanned from the lower side of the image in the temporary obstacle to the upper part. The target line changes to the next line without scanning when the obstacle is detected. The processing time is shortened though it is a little because all temporary obstacle pixels are



(a) no obstacle

(b) no obstacle, the distance corridor is reflected the sunlight.

(c) obstacle in the distance

(d) obstacle nearby.

Figure 5. Obstacle detection and moving direction detection.

not scanned. The applied result of proposed method is shown in the fourth row of figure 5. It can be confirmed that the obstacle is detected accurately compared with the result of the second and the third rows of figure 5.

### 3.3 Moving direction detection

Our system goes straight at the center of the corridor when the obstacle does not detected, and when the obstacle is detected, applies the avoidance movement.

#### 3.3.1 Center following movement

When the vanishing point obtained as an intersection of two boundary lines is assumed to be a moving direction, a location about 15m or more away from the mobile robot should be assumed to be the destination. In this case, when the mobile robot is located at wall side, it does not immediately return to the central position. It returns gradually spending long time in the position of the center. Then, a central position of the corridor of 1m forward is assumed to be the destination.

The right of figure 5(a)-(c) shows the detected moving direction of the center following movement. The blue line is the center line of the corridor, the red point in the terminal of yellow line is the target position. In figure 5(c), though the obstacle is detected forward, the center following movement is worked because it is in the distance.

#### 3.3.2 Collision avoidance movement

In our system, when the large obstacle is detected, the stop or avoidance movement is worked. Which movement is worked is decided according to the position and size of the obstacle. If open spaces that are wider than the size of the robot are in either of the obstacle right or left, it is presumed that the robot can avoid this obstacle, and applied the avoidance movement. As this movement, the central location in open space is set to the destination. The robot stops when there is no space that the robot can pass.

Figure 5(d) shows the detected moving direction of the avoidance movement. The horizontal light blue line is the bottom side of the obstacle, and the avoidance direction $\theta$ is calculated using this line. The red point in the terminal of yellow line is the destination.

## 4. Experiment

We developed a mobile robot based on commercial powered wheelchair, and a camera is equipped in front of the robot. The main computer of our robot is a laptop computer. Images were acquired with a resolution of 320 $\times$ 240 pixels. The robot was moving with a constant forward velocity of 1.0 km/h. Though the reference area for the obstacle detection can be set up freely, a point of 1m ahead is set the top of

(a) person (side) (b) chair (c) book

Figure 6. Obstacles for stopping experiment.

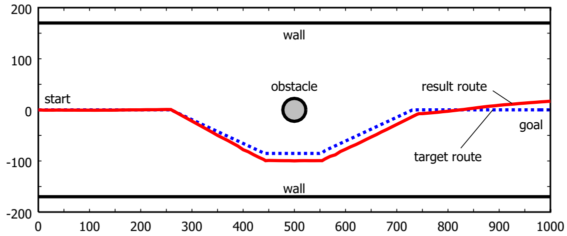



Figure 7. Avoidance movement.

the reference area considered the moving speed of the mobile robot. Moreover, we set the observation area from 1m to 2m to check obstacle exists or not.

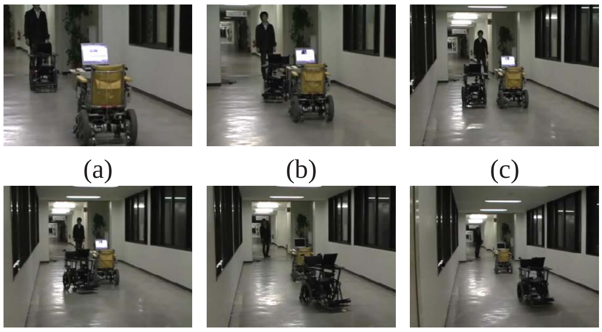
### 4.1 Stop and avoidance movement

The obstacle, such as a person, a trash can, and a chair, as shown in figure 6, was put forward of the mobile robot, and the stop movement was worked. This experiment was done in the corridor at our lab. The width of the corridor was approximately 2.0 m. Then, we measured the distance between the obstacle and the mobile robot when the robot stops. As the result, it was confirmed to stop within the range from 1m to 2m in distance from the robot to the obstacle though it differed depending on the size of the obstacle.

Moreover, the obstacle avoidance movement was successfully worked as shown in figure 7, when enough space existed. The blue line is the target route, and the red line is the result route. Here, the start point is (0, 0), and the goal point is (1000, 0). We put a potted plant which diameter is 0.45m at (500, 0). Oppositely, when enough space did not exist, the robot stopped within the range from 1m to 2m in distance successfully.

### 4.2 Face-to-face movement

Two mobile robots R1 and R2 were made to coexist in the same environment, and the face to face moving experiment was carried out. The width of the corridor was approximately 3.4 m. Here, the center following, stop and avoidance movement were implemented on R1. On the other hand, the avoidance movement was not implemented on R2, and other two movements, the center following and stop movement were implemented on R2. As the result, R2 worked the stop movement when R1 was detected as an obstacle, and R2 moved again at a central position of the corridor if R1 was not detected as an obstacle. Similarly, R1 worked the stop movement when R2 was detected as an obstacle. To avoid R2, R1 calculated the avoidance direction and distance. And then, R1 worked the avoidance movement. Moreover, R1 moved again at a central position of the corridor after having avoidance movement finished. The moving experiment was carried out as shown in figure 8.



(a) (b) (c) (d) (e) (f)

Figure 8. Face to face moving experiment.

## 5. Conclusion

This paper proposed the appearance based obstacle detection method, and developed the mobile robot that moves automatically without needing environmental information beforehand while recognizing the frontal surrounding environment with only one general camera. We carried out the stop experiment for various obstacles, and found the robot can be stop with the range from 1m to 2m though there was a difference in the distance that actually stopped according to the size of the obstacle. Furthermore, we developed two mobile robots and coexists these robots in same environment. As the result, these robots were able to pass each other without collision.

### References

[1] M. Ebner and A. Zell. Centering behavior with a mobile robot using monocular foveated vision. *Robotics and Autonomous Systems*, 32(4):207–218, 2000.

[2] T. Joochim and K. Chamnongthai. Mobile robot navigation by wall following using polar coordinate image from omnidirectional image sensor. *IEICE*, E85-D(1):264–274, 2002.

[3] T. Saitoh, T. Osaki, and R. Konishi. Monocular autonomy following vehicle. In *SICE-ICASE International Joint Conference 2006 (SICE-ICCAS2006)*, pages 5963–5966, 2006.

[4] H. Liu, Z. Zhang, W. Song, Y. Mae, M. Minami, and S. Aoyagi. Evolutionary recognition of corridor and turning using adaptive model with 3d structure. In *SICE-ICASE International Joint Conference 2006 (SICE-ICCAS2006)*, pages 2840–2845, 2006.

[5] H. Koshimizu and M. Numada. On a fast hough transform method PLHT based on piece-wise linear hough function. *IEICE*, J72-D-II(1):56–65, 1989. (in Japanese).

[6] I. Ulrich and I. Nourbakhsh. Appearance-based obstacle detection with monocular color vision. In *AAAI National Conference on Artifical Intelligence*, pages 403–408, 2000.