



Real-time obstacle detection in a darkroom using a monocular camera and a line laser

Sota Akamine¹ · Shingo Totoki¹ · Taku Itami¹ · Jun Yoneyama¹

Received: 14 April 2022 / Accepted: 3 August 2022 / Published online: 23 August 2022
© The Author(s) 2022

Abstract

When a disaster strikes, one of the highest priorities is to save the lives of the victims. The proportion of victims that rescuers can save is strongly related to how quickly rescue efforts can begin. Therefore, early detection of disaster victims is very important. However, significant risks are involved in rescue operations immediately after a disaster. In fact, in the Great East Japan Earthquake, approximately 250 firefighters died while rescuing victims. Under such circumstances, rapid and safe rescue operations are needed at disaster sites. For this purpose, it is important to improve the technology of disaster relief robots. In this paper, we propose an algorithm to measure the linear distance to an obstacle in real time using only a line laser and a monocular camera. This approach allows the use of a camera to obtain more information than the one-dimensional information such as that obtained by ultrasonic sensors. Moreover, this method of obstacle detection for disaster rescue robots is smaller and more durable than large measurement systems such as LiDAR that have been used in the past. In addition, since only one camera is used, the processing cost is low and the processing equipment is expected to be small. The proposed method's effectiveness is indicated by comparing the distance measured from the image processing results in a dynamic environment with the actual distance between the obstacle and a crawler robot by having the robot move straight toward to an obstacle.

Keywords Robot vision and image processing · Mobile robots · Motion planning and navigation · Obstacle detection

1 Introduction

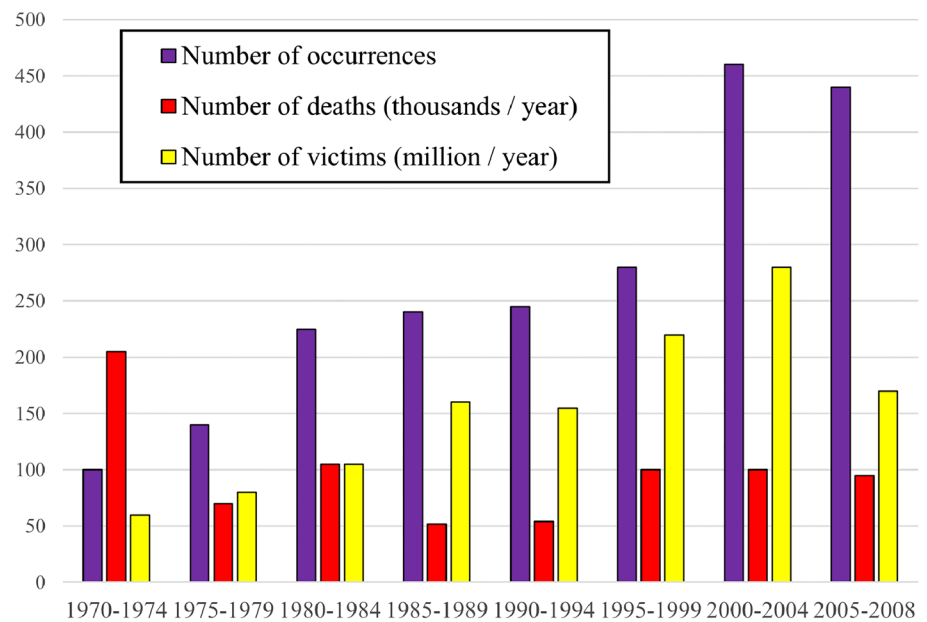
According to the Centre for Research on the Epidemiology of Disasters (CRED) database, about 8,000 large-scale disasters worldwide occurred in the 50 years between 1967 and 2016, in which about 2.8 million people were killed [1]. In addition, a 2010 Cabinet Office White Paper on Disaster Prevention observed that, given the many natural disasters in Japan (such as earthquakes with a magnitude of 6.0 or more, about 20 % of the world total [2]), rescue activities are very important here. In fact, about 250 members of the fire brigade carrying out rescue activities in the Great East Japan Earthquake perished [3]. Figure 1 summarizes the numbers of natural disasters, deaths, and victims globally between 1920 and 2008. Given these circumstances, rescue activities at disaster sites must be

both prompt and safe. Therefore, it is important to improve the technology of disaster robots. At a disaster site, a self-sustaining mobile robot needs to quickly and safely select a route to a destination in order to carry out rescue activities, sometimes in the dark where camera function is impaired or roads are damaged [4]. In addition, detecting both dynamic and static obstacles with high accuracy and low delay is an important factor that enables autonomous driving [5]. Focusing specifically on the obstacle-detection function, conventional technologies includes ultrasonic sensors, position-sensitive detector (PSD) sensors, 2-D LiDAR, and 3-D laser scanners. Each of these, however, has disadvantages. Ultrasonic sensors have a slow response speed, are affected by dust and water, and have low angular resolution [6, 9, 11]. PSD sensors are affected by colors and materials and have short measurement distances; 2-D LIDAR is costly and has poor durability; finally, 3-D laser scanners are vulnerable to black objects and mirrors [4, 6]. In contrast, cameras are inexpensive and small, can acquire environmental factors such as terrain appearance, and can operate in various climatic environments [7]. Current mobile robots obtain position information about obstacles via multiple

✉ Taku Itami
itami@ee.aoyama.ac.jp

¹ Graduate School of Science and Engineering, Aoyama Gakuin University, 5-10-1 Fuchinobe, Chuo-ku, Sagami-hara-shi, Kanagawa 252-5258, Japan

Fig. 1 Numbers of disasters, deaths and victims (1970 → 2008)



installed cameras. However, if technology can be developed to acquire location information using only a monocular camera, more data can be acquired and the amount of information and accuracy can be improved; if multiple devices are used, the accuracy can be further improved [7, 10].

In this paper we propose an algorithm that can measure the straight-line distance to an obstacle in real time with only a line laser and a monocular camera. Suetsugu et al. conducted obstacle-detection research using a line laser. However, in addition to manual photography, the conventional method requires a long processing time because calculations are performed after image processing, and the obstacle and camera must both be static. Moreover, conventional methods of detecting obstacles in real time do not consider the influence of surrounding light, and the method of detecting the position of the line laser is also not considered. Thus, some challenges remain to be resolved for practical use. In addition, the formula for calculating distance in conventional research had a large error when measured dynamically. There are also studies that use machine learning to estimate depth from information obtained using a monocular camera. However, a maximum error of about 10% occurs in distance measurement, and the processing cost is high, so the machines that do the processing calculations are too large, which makes it difficult to install in a disaster rescue robot [8].

2 Purpose

In this paper, camera information is read from a robot equipped with a line laser and a monocular camera, and OpenCV and Python are integrated to perform image

processing tasks such as noise removal and setting the laser color recognition threshold. On the basis of our findings, we propose an algorithm that can acquire position information and measure the straight-line distance to an obstacle in real time even in a dynamic environment. This method can detect obstacles using only a line laser and a monocular camera. Therefore, this method is not affected external noise., and it is easy to improve downsizing and durability. We demonstrate the effectiveness of the proposed method by actually measuring the distance to an obstacle after moving the robot for 1 s and comparing the result with the distance measured by an image processing result.

3 Methods

To enable real-time measurement of the straight-ahead distance to an obstacle even in a dynamic environment, we first irradiate a red-line laser diagonally from above the robot in the straight-ahead direction. The position information of the obtained line laser is then extracted from the image information. The straight-line distance is measured by detecting the position information of the line laser light projected onto the obstacle with OpenCV and calculating the distance using the similarity relationship of triangles. When the red-line laser is radiated diagonally from above the robot in the straight-line direction, the laser light will be projected onto an obstacle in that direction if one exists. The straight line of the horizontal light projected onto the obstacle will then rise vertically. The actual situation is shown in Fig. 2.

This line of light is processed when the camera captures an image, which is then imported by OpenCV and binarized, which binarizes the image by the color threshold set for the

laser color. After the noise is removed from the binarized image, the distance to the obstacle can be calculated in real time by extracting the coordinates of the laser and calculating the distance using the similarity relation of triangles. The flowchart of this algorithm is shown in Fig. 3.

The image information from the camera is first sent to OpenCV, which converts it into HSV values. The threshold value is the red light extracted from the image in advance of the line laser using the HSV value. Binarization is then performed between red and other colors.

The binarized output screen is then morphologically transformed using OpenCV. Shrinking, a basic morphological process, is used to remove small noise in the image, followed by opening, a noise processing method that increases the area by performing expansion. Then, the part of the line laser that does not hit an obstacle (the position of the line laser when there is no obstacle) is deleted by specifying the area. Finally, the distance to the obstacle is calculated by labeling and extracting the coordinates of the centroid of the line laser. Figure 4 shows the image after the video from the camera is binarized by OpenCV using a threshold with a predetermined HSV value. Figure 5 shows the output screen after noise removal and labeling processing, and Fig. 6 shows the screen displaying the original image and the coordinates being extracted. Figure 5 shows the image of Fig. 4, in which OpenCV shrinks the pixel values, noise is removed, the image is expanded to avoid holes in the pixels, and labeling is applied to detect the centroid of the line laser hitting the obstacle. The coordinates of the detected centroid are the blue dots in Fig. 6, which is an image from a camera in which the program displays the position coordinates of a line laser hitting an obstacle as blue dots. As can be seen from this image, this method can obtain the position information of the line laser in real time using OpenCV.

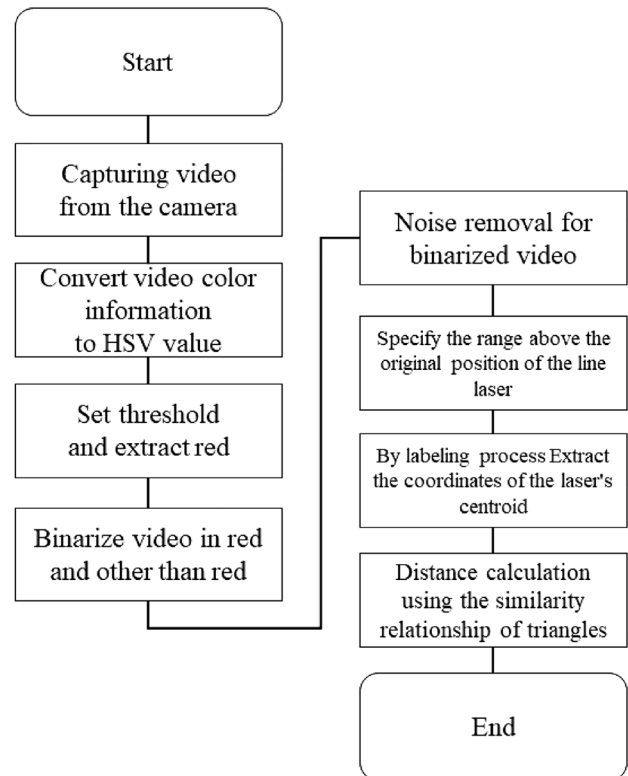


Fig. 3 Flowchart of distance measurement algorithm

Specific calculation methods, the red horizontal line laser is attached to the robot at height H [mm], and the maximum irradiation distance is L [mm]. Let T_{max} [mm] be the maximum shooting range at the maximum irradiation distance of the line laser in the camera attached to the robot, and let t_m [pixel] be the position of the line laser in the image when no obstacle is present. Then, an obstacle is placed in the robot's straight-ahead direction, and the height of the irradiation



Fig. 2 Actual experiment

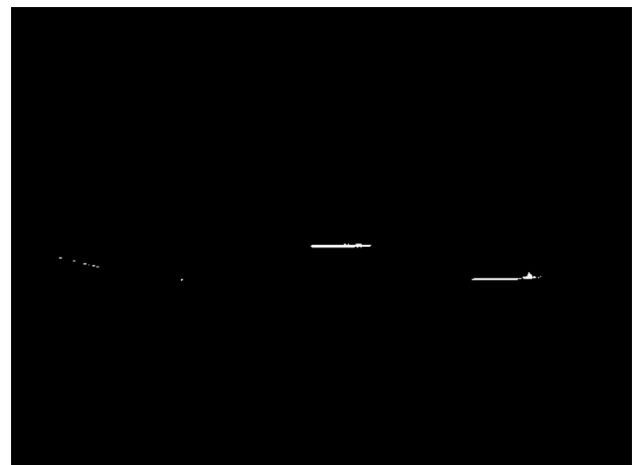


Fig. 4 Binarization using HSV values

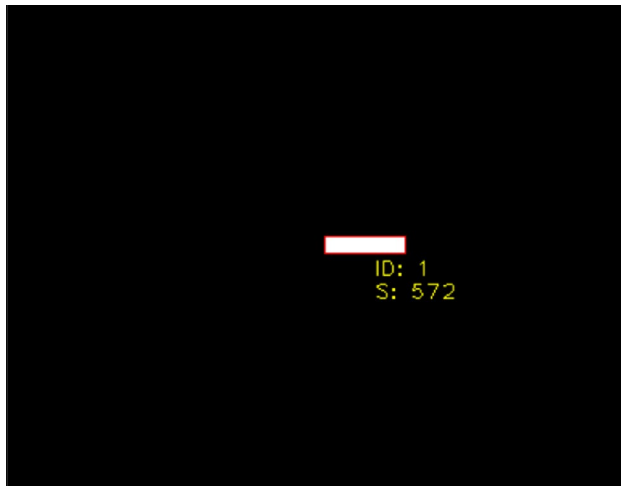


Fig. 5 Noise removal and labeling for binarized image



Fig. 6 Original image and coordinate extraction position

position where the line laser is projected onto the obstacle is T_R [mm], while the position of the line laser in the image at that time is t [pixel]. The height T_R [mm] of the laser irradiation position can be calculated by Eq. (1).

$$T_R = \frac{T_{max} * t}{t_m} \quad (1)$$

From the result, the distance D [mm] to the obstacle can be obtained by Eq. (2) using the similarity relation of triangles.

$$D = L - \frac{T_R * L}{H} \quad (2)$$

Since the position information in the image of the red-line laser is used to measure the linear distance to the obstacle, it is necessary to read the image from the camera and detect the position of the line laser in real time. Therefore,

in this study, the position information of the red line laser is detected using OpenCV. The detection method was as follows. First, the image captured from the camera is imported into OpenCV. To detect only the red line laser in the image, the HSV value is set to the color of the laser, and the image is binarized according to the pre-set threshold value. The place judged to be red in the image is converted to white (1), and the other colors are converted to black (0) to create the image. Noise is removed from a binarized image by opening processing, and the system detects the center coordinates of the line of the line laser irradiated to the obstacle in the image. This makes it possible to obtain the value of t [pixel] used to calculate distance. Then, by substituting this value into Eq. (1), the value of T_R [mm] can be obtained. Figures 7 and 8 present explanations of these values.

4 Experiment

To verify whether the proposed method actually measures the distance to obstacles in the dark, we built a simple darkroom consisting of wooden boards and blackout curtains. The effectiveness of the proposed method is verified by measuring the distance to obstacles in real time in the darkroom, in which only the light of the line laser could be observed from the read image. The camera was directly connected to a PC, and it was possible to read the video in real time and start the developed program. Also measured in advance are the height H of the line laser from the ground, the maximum irradiation distance L when there are no obstacles, and the maximum shooting distance T_{max} at the maximum irradiation distance. Table 1 lists the initial values measured in advance, and Fig. 8 shows the robot equipped with the camera and line laser. In this research, the camera is eMeet C960 made by EMeet and the line laser is Qlaers Red Laser Module 650nm 1mW.

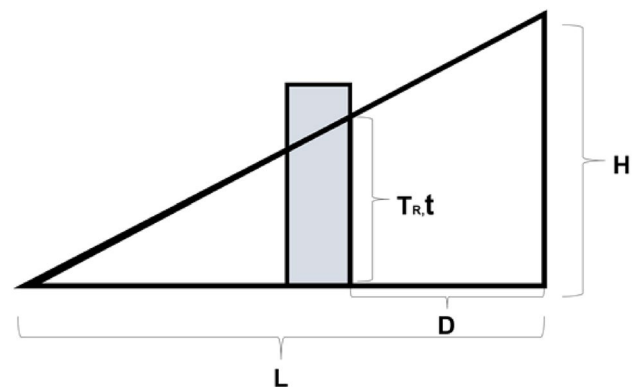


Fig. 7 Trigonometric relationship of the values used to calculate the distance to obstacles

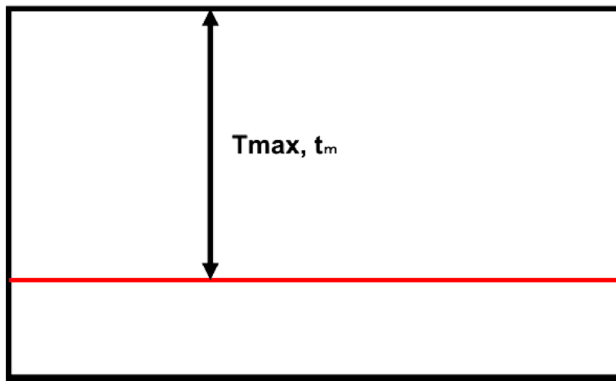


Fig. 8 Relationship between the image obtained from the camera and the constant when no obstacles are present

Table 1 Initial value measurement result

Measurement points	Actual measurement value
Line laser height H [mm]	173
Maximum irradiation distance L [mm]	853
Maximum shooting range T_{max} [mm]	563
Pixel value of T_{max} t_m [pixel]	298

Image processing was performed on these initial values in real time, and the measured T_R and t were substituted into Eq. (1) and (2), respectively, to calculate the distance between the robot and the obstacle. The experimental method is to calculate the distance in real time while moving the robot for 1 s. The actual distance between the robot and the obstacle after 1 s was used as the measured value. The same experiment was repeated 5 times (Fig. 9).

5 Experimental Results

Table 2 presents the process-measured and manually measured values after the obstacle and the robot were placed 700 mm apart and moved for 1 s. In addition, Fig. 10 shows the third graph in which the error was 0, and Fig. 11 shows the displacement of the reading position of the line laser during movement for 2 s.

6 Discussion

The experimental results demonstrate that there was almost no error between the process-measured and manually measured values. It is considered that the reason why error was not completely eliminated is that the initial value



Fig. 9 Robots actually used in the experiment

Table 2 Experimental result

Times	Process measured[mm]	Manually measured values[mm]	Error [mm]
1	564	560	4
2	545	542	3
3	536	536	0
4	554	551	3
5	536	539	3
Avg.	547	545.6	2.6

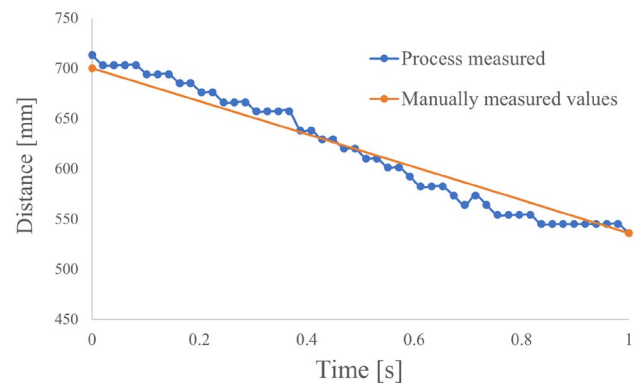


Fig. 10 Graph of experimental results

was measured manually with a ruler, and if the initial value deviates by even 1 cm, the manually measured value also has an error of several millimeters. In this experiment, it was confirmed that there were no problems and that recognition was performed with minimal error. In view of its use in real-world conditions, it is also considered necessary to use it outside the darkroom.

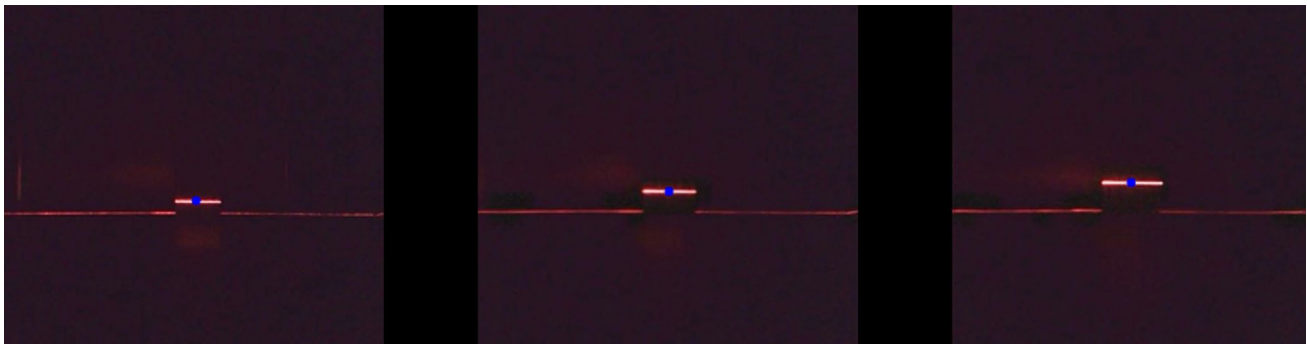


Fig. 11 Displacement of reading position

7 Conclusion

In this paper, we proposed an algorithm to calculate the straight-line distance to an obstacle in real time and conducted an experiment to verify its effectiveness. Our results confirmed that it is possible to extract the position information of a laser in the dark where the camera can recognize the light of the line laser and the position information of the moving laser. We demonstrated that the position information could be used to detect the distance to the obstacle in real time. In the future, we will develop a self-sustaining mobile robot equipped with a line laser and a camera, and will detect errors during movement by setting the speed. In addition, the position information of the laser will be recognized in response to the change in the color of the line laser due to brightness.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

1. CRED International disaster database (EM-DAT), available from <https://www.emdat.be/>, (accessed on 20 May, 2022)
2. Cabinet Office, Government of Japan, Disaster Management in Japan (2021)
3. Fire and Disaster Management Agency Great East Japan Earthquake Damage Report No. 145
4. Suetsugu M, Yang S, Kyushu SS (2019) Institute of Technology-Mobile robot using line laser and camera Obstacle detection and shape recognition method, Proceedings of IIAE Annual Conference 2019
5. Francisco AX, da Mota MXR, Joel JPC, Rodrigues VHC, de Albuquerque A, de Alexandria AR (2018) Localization and navigation for autonomous mobile robots using petri nets in indoor environments. *IEEE Access* 6:31665–31676
6. Siegwart R, Nourbakhsh IR (2004) Introduction to autonomous mobile robots. MIT Press, Cambridge
7. Garnett N, Silberstein S, Oron S, Fetaya E, Verner U, Ayash A, Goldner V, Cohen R, Horn K, Levi D (2017) Advanced Technical Center Israel, General Motors R & D, Hamada 7, Herzlyia, Israel, Real-time category-based and general obstacle detection for autonomous driving, Proceedings of the IEEE International Conference on Computer Vision (ICCV), pp.198-205
8. Chen H-C (2019) Monocular vision-based obstacle detection and avoidance for a multicopter. *IEEE Access* 7:167869–167883
9. Gibbs G, Jia H (2017) Obstacle detection with ultrasonic sensors and signal analysis metrics. *Transp Res Proc* 28:173–182
10. Christian H, Lionel H, Lee GH, Fraundorfer F, Furgale P, Sattler T, Pollefeys M (2017) 3D visual perception for self-driving cars using a multi-camera system: Calibration, mapping. *Image Vis Comput* 68:14–27
11. Suaif A, Gani FQ (2021) Implementation of ultrasonic sensor as a radar for obstacle detection. *Andal J Electr Electron Eng Technol* 01(01):12–15