

MODEL-GUIDED MONITORING OF A BUILDING ENVIRONMENT BY A MOBILE ROBOT

M.Yachida, T.Ichinose and S.Tsuji

Dept. Of Control Engineering,

Osaka University

Machikaneyama, Toyonaka, Osaka 560, Japan

ABSTRACT

This paper presents an initial attempt to build a mobile robot which navigates in and monitors buildings such as offices, schools, factories, hospitals, etc. and reports changes there to a human operator. The robot is initially parked at a standard position and is then driven around passages of the building via a given route. While moving around, it stops every a few meters, takes a picture and compares the picture with a stored model of the building. It first extracts vertical edges from the image which often appear in the building scene as stably extracted features, and then determines camera parameters (such as camera position) by matching the extracted features and the model. Using these camera parameters, it generates an image model from the building model and detects changes in the scene by comparing the input image and the generated image model.

1. INTRODUCTION

There has been much work on mobile robots which navigate in unknown environments, using vision [1,2,3] or ultrasonic range finder [4]. Because purpose of these robots is to find a path in an unknown environment, few robots use a priori knowledge (model) of the environment or find changes there.

In this paper, we describe a mobile robot which finds not only obstacles for navigation but also changes for monitoring an environment. In order to simplify the problem, we restrict the robot's environment to buildings such as offices, factories, schools, etc., most of which consist of straight edges. The robot is initially given a model of the building where it works, and uses it to guide the image processing process and to find changes in the scene.

The robot is initially parked at a standard position and is then driven around passages of the building via a given route. While moving around, it stops every a few meters, takes a picture and analyzes the image to find obstacles and changes in the scene as follows. It first extracts vertical edges from the image which often appear in the building scene by the Hough-like method as "stably extracted features" and then determines camera parameters by matching the extracted features and the building model. Using these camera parameters, it generates an image model from the building model and detects changes in the scene by comparing the input image and the generated image model. If there are important changes in the scene, it reports them to a human operator.

2. SYSTEM DESIGN

Fig.1 shows hardware configuration of the mobile robot, which consists of a control unit, a micro-computer and a floppy disc. The image from the camera is stored in the floppy disc and is sent to the mini-computer via a communication line.

The robot is initially parked at a standard position in front of a known object. It views the object and calibrates the camera parameters such as the camera position (x,y,z) , camera's viewing angles (α,β,γ) and its focal length (f) . Then it is driven around passages of the building via a given route; however, as an initial step, it is currently driven manually. While moving around, it stops every a few meters (currently 1 meter) and takes pictures to find obstacles and changes in the scene. Here, we assume that variables in the camera parameters are x, y and α , that is x, y , positions of the robot and the panning angle of the camera. Changes of these parameters $(\Delta x, \Delta y, \Delta \alpha)$ from the last stop are measured by the position sensors of the robot. However, the position sensors usually have measurement errors; and therefore it utilizes the viewed image to find the precise values of these camera parameters. Then, the image is analyzed to detect obstacles and changes in the scene. Fig.2 shows image analysis modules and its processing flow used at this stage. If important changes are found in the scene, they are reported to an operator. If not, the robot is driven a few meters and takes another picture. This is iterated until all the scene along the given route has been examined.

3. SYSTEM DESCRIPTION AND EXPERIMENTAL RESULTS

3.1. Moving object detection

When the robot stops to view a scene, it first examines if there are non-stationary objects in the scene. It takes several pictures and detects non-stationary object (usually man) by a systematic difference method proposed by Yachida

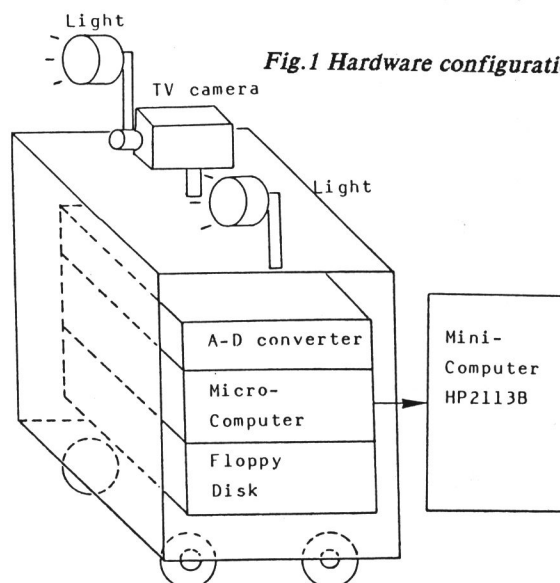


Fig.1 Hardware configuration

et.al [5]. Then the size of each non-stationary region is calculated in order to know whether the moving object is coming closer to or going away from the robot. If the size becomes larger, the object is considered as coming closer to the robot, and the robot stays there until the moving object passes by the robot so that it may not collide against the moving object. If not, it analyzes the image to find changes in the scene. Since the moving objects complicates the image analysis task, it deals the non-stationary regions as "don't care" regions in the rest of the image analysis.

3.2 Vertical edge detection

In order to eliminate noise in the input image, we first apply edge preserving smoothing operation [6] to the image, and then apply Sobel's operator to find edge points in the image.

Since vertical lines often appear in the building scene and can be extracted stably, they are first sought in the edge image. To find vertical lines, we make a histogram whose horizontal axis is the y axis of image plane and the vertical axis is the number of edge points which have approximately vertical edge direction. Since each peak in the histogram corresponds to a vertical line in the image, positions of vertical lines in the image can be known from the histogram. Then, we trace vertical edge points in the image. Fig.3 shows results of this process.

When the vertical lines have been found, we try to determine precise camera parameters. Since we know the approximate changes of camera parameters by the position sensors, we can generate an image model from the building model and the camera parameters. By matching the vertical lines obtained from the image and the generated image model, we can estimate the precise camera parameters. However, a problem here is that a slight measurement error of the panning angle causes large difference in the image model, which makes difficult the comparison between the detected vertical lines and the image model. Therefore, it is necessary to estimate rather precise panning angle before the above process.

3.3 Panning angle estimation

We can determine the pan angle of the camera from the position of the vanishing point of the floor. To know the vanishing point in the input image, we use the fact that the vanishing point and the floor boundary make a triangle. We

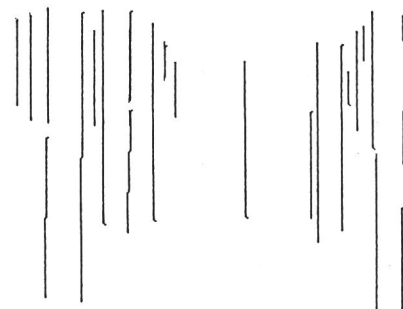
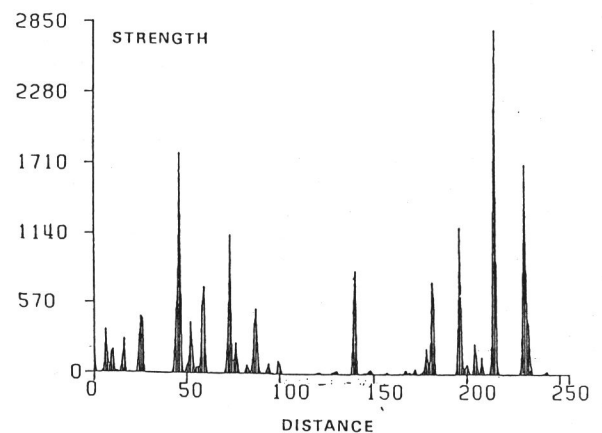
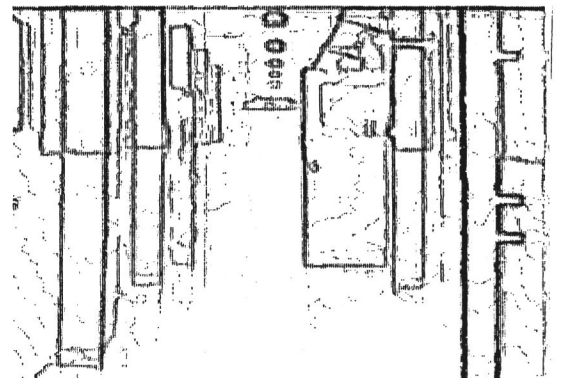
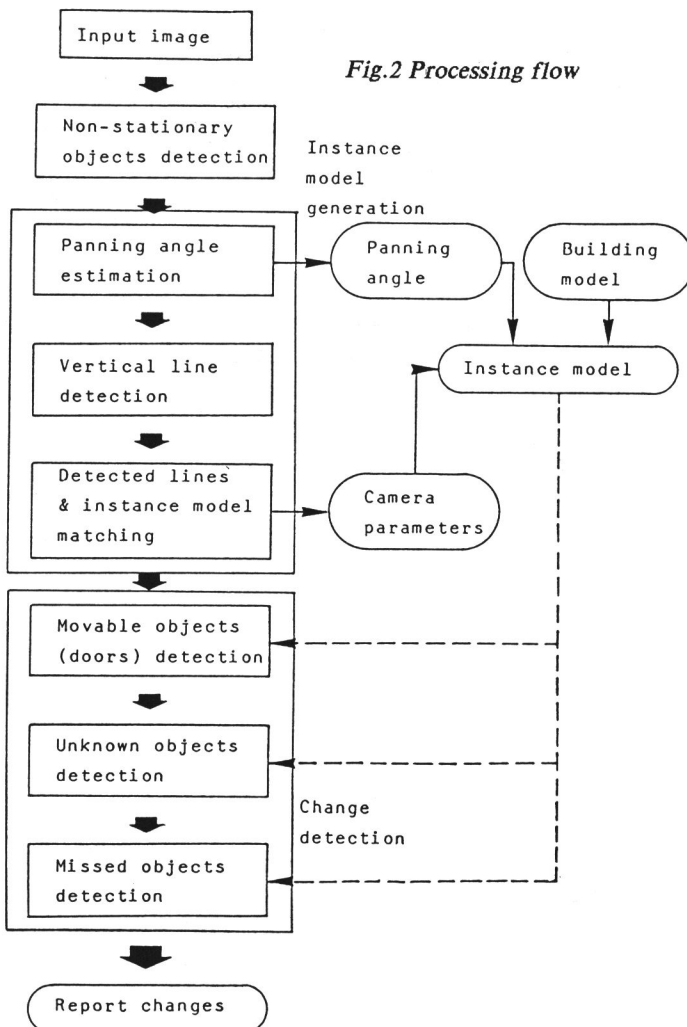


Fig.3 Vertical edge detection
 (a) result of Sobel's operator (b) histogram
 (c) result of vertical edge detection

first generate this triangle from the building model and the camera parameters, and search a region in the input image which matches to this triangle.

We first set the small window around the center of the triangle and calculates the average gray level (G) in the window. Since the measurement error of pan angle causes the horizontal translation of the vanishing point in the image, we move the generated triangle horizontally in the image. Then, at each position of translation, the difference between the G and the average gray level in the triangle is calculated. The position of triangle where the difference is minimum gives the position of vanishing point in the input image. It is now easy to calculate the pan angle of the camera from the position of the vanishing point.

3.4 Model

Fig.4 shows a part of the building model. As seen in the figure, the model consists of a map which is viewed from the vertical direction and attributes such as height and so on which are attached with each object. Therefore, when the camera parameters are given, we can generate the image model. Movable objects such as a door also has an attribute of maximum movement if they have constraints on the movement.

3.5 Matching vertical lines to image model

Since rather precise pan angle has been obtained, we generate an image model from the building model and the camera parameters. Then this image model is compared with the vertical lines obtained from the input image as follows.

We first set a search region around each vertical line in the image model. The size of search region is determined from the estimated measurement errors of position sensors. Then we examine if a vertical line of similar length exists in each search region. From the positions of matched lines, we can determine the precise camera parameters.

3.6 Change detection

We have obtained the precise camera parameters. Then we generate the more precise image model using them, and detect changes in the scene by comparing the edge image of input picture and the image model.

Each line in the generated image model is examined whether it actually exists in the input images. Here, in order to check the lines of movable objects (doors), we simulate their movement and compares the lines generated at each movement and the input image. After comparing all the lines in the generated image model, we classify those lines which could not be found in the input image into following three cases.

- (1) hidden by other objects,
- (2) too small to distinguish,
- (3) the object has been missed.

(1) is the case when the object is hidden by the moving objects detected earlier or the movable objects. This case is easily distinguished by checking whether the sought line is inside the region of these objects. (2) is the case when an object is far from the robot. These object can be distinguished by the building model and we do not examine lines of these objects in the input image, because the robot can get better view of them when it moves closer to them later. For the rest of lines, we classify them into the situation (3) and reports that the object is missed.

After all the lines in the generated image model have been examined, we check whether all the edge points in the input image have been matched to some line of the image model. If there are edge points which have not been matched to the image model, the robot considers that they are caused by some unknown objects and reports it. Experimental results will be shown at the conference.

4. CONCLUSION

This paper described a monitoring robot which navigates in and monitors an building. The robot has a model of the building which is used to update its camera parameters, to navigate and to detect changes in the environment. The robot can be used for a different building by changing its model. The authors would like to thank T. Hayase of our university in preparation of the manuscript.

REFERENCES

- [1] D.B.Gennery; A stereo vision system for an autonomous vehicle, pp. 576-582, IJCAI-77 1977
- [2] A.M.Tompson; The navigation system of the JPL robot, pp. 749-757, IJCAI-77, 1977
- [3] H.P.Moravec; Rover visual obstacle avoidance, pp.785-790, IJCAI-81, 1981
- [4] J.Iijima, Y.Kanayama, S.Yuta; A location control system for mobile robots, pp. 779-784, IJCAI-81, 1981
- [5] M.Yachida, M.Asada, S.Tsuji; Automatic analysis of moving images, pp. 12-20, IEEE Trans. On PAMI, vol. PAMI-3, no.1, 1981
- [6] M.Nagao and T.Matsuyama; Edge preserving smoothing, Computer Graphics and Image Processing, vol.9, no.4, pp.394-407, 1979

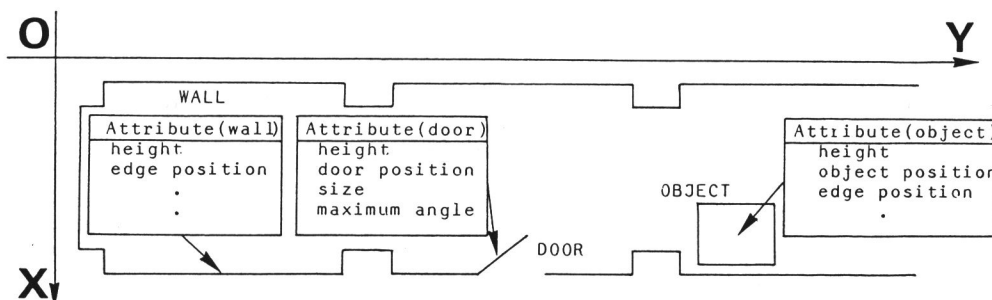


Fig.4 Model