Vision-based Obstacle Avoidance System for Autonomous Mobile Robot in Outdoor Environment.

Ji Eun Jung*, Kil Soo Lee*, Hyung Gyu Park*, Yung Ho Koh**, Man Hyung Lee*
**Graduate School of Mechanical Engineering, Pusan National University*
***Futronic Co,. Ltd*

Abstract: In this paper, we propose the obstacle avoidance system on based vision sensor for an autonomous mobile robot. For real-time turn angle correction, we obtain the state equation of a mobile robot from input-output continuous data. Each individual image pixel is classified as belonging either to an obstacle or non-obstacle based on its color property (HSI color model). HSI color model is less sensitive to illumination changes than RGB color model. Using some conditions and the voting system, we choose the path area, the navigation point, and the turn angle. This method uses a single color camera.

Keywords: Vision system, Autonomous mobile, obstacle avoidance.

I. INTRODUCTION

As to vision-based approaches to obstacle detection, they basically can be divided into three classes. In the first class, obstacles are extracted directly from 2D images. Only one camera and only the image in the current navigation cycle are used, with certain a priori knowledge and predefined assumptions being considered. In the second class of approaches, motion information obtained from a sequence of images are utilized to detect obstacles. The most popular approaches in this class are based on optical flow. In the third class of approaches, obstacles are detected using stereo-vision techniques.[1]

Although the first class in general takes less computing time and has better detection results than the second and the third classes, in fact, it does not really detect obstacles because obstacles are extracted directly from the 2D image. Shadows on roads may also be regarded as obstacles in this class of approaches. On the contrary, in the second and the third classes, 3D computer vision techniques are used to really judge whether object on roads are obstacles, although more computing time is required in these two classes than the first class

In this paper, we propose an intelligent approach to obstacle guidance in outdoor environments using a single color camera.

We use subspace system identification algorithms, calculate a state-space model form input-output measurements of system. As the state-space model, we predict the real wheel angle of robot. By the combination of wheel angle and velocity, prediction of location can be computed.

Various color information on roads is used in this paper to extract path and obstacle. For this, the HSI color model is chosen, which is less sensitive to intensity than RGB color model. To judge whether a pixel is an obstacle or not, the histogram of the front trapezoid is compared with one of the input image. The process to vote the possible area as path, the path area is extracted. We select the navigation point, turn angle from the path area.

The remainder of the paper is organized as follows. In section 2, subspace system identification algorithms are introduced. In section 3, the details of the proposed vision-based obstacle detection method is described. The descriptions of the obstacle avoidance method are included in section 4. Experimental studies from simulation in section 5 and conclusion are presented in section 6, respectively. The paper ends with reference.

II. Subspace System Identification

1. State-space model

Subspace identification algorithms calculate a state-space model from input-output measurements of a linear system of the form

$$x_{k+1} = Ax_k + Bu_k + w_k \quad (1)$$
$$y_k = Cx_k + Du_k + v_k \quad (2)$$

where $u_k \in R^m$ and $y_k \in R^l$ are the given measured input and output sequences of the multivariable system with *m* inputs and *l* outputs. The consecutive states $x_k \in R^n$ are unknown, as are the (real) system matrices *A*, *B*, *C* and *D* of appropriate dimensions. The sequences $w_k \in R^l$ and $v_k \in R^m$ represent so-called process and observation noises, respectively. The noise w_k and v_k are both assumed to be stationary white Gaussian processes with zero-mean and covariance matrices:

$$E\left[\begin{pmatrix} w_p \\ v_p \end{pmatrix}\begin{pmatrix} w_q^T & v_q^T \end{pmatrix}\right] = \begin{bmatrix} Q & S \\ S^T & R \end{bmatrix}\delta_{pq} \geq 0 \tag{3}$$

Where $\delta(\cdot)$ is the Kronecker delta function.

$$\begin{bmatrix} \hat{A} & \hat{B} \\ \hat{C} & \hat{D} \end{bmatrix} = \min_{A,B,C,D} \left\| \begin{bmatrix} \hat{x}_{i+1} & \cdots & \hat{x}_{i+j} \\ y_i & \cdots & y_{i+j-1} \end{bmatrix} - \begin{bmatrix} A & B \\ C & D \end{bmatrix}\begin{bmatrix} \hat{x}_i & \cdots & \hat{x}_{i+j-1} \\ u_i & \cdots & u_{i+j-1} \end{bmatrix} \right\|_F^2 \tag{4}$$

Where $\|\cdot\|_F$ denotes the Frobenius-norm of a matrix[2].

Formidable as it may seem, subspace algorithms manage to identify the order of the system n (the number of difference equations needed to model the data appropriately) and to calculate the matrices A, B, C, D, Q, R and S.[3]

A least squares problem to obtain the state space matrices solve Eq. (5).

$$\begin{pmatrix} X_{i+1} \\ Y_{i|i} \end{pmatrix} = \begin{pmatrix} A & B \\ C & D \end{pmatrix}\begin{pmatrix} X_i \\ U_{i|i} \end{pmatrix} \tag{5}$$

Where $U_{i|i}$, $Y_{i|i}$ are block Hankel matrices with only one block row of inputs respectively outputs, namely $U_{i|i} = \begin{pmatrix} u_i & u_{i+1} & \cdots & u_{i+j-1} \end{pmatrix}$ and similarly for $Y_{i|i}$. This set of equations can be solved. As there is no noise, it is consistent.

2. Modeling of robot

The system input is steering wheel angle and output is vehicle's yaw. Fig. 1 represents the input signal and Fig. 2 represents the output signal.

We use the Visual C++ for data acquisition and use MATLAB toolbox for data processing. By relation of input data and output data, we solve the discrete-time state equation.

Fig. 3 represents the estimation of system model using the discrete-time state equation.

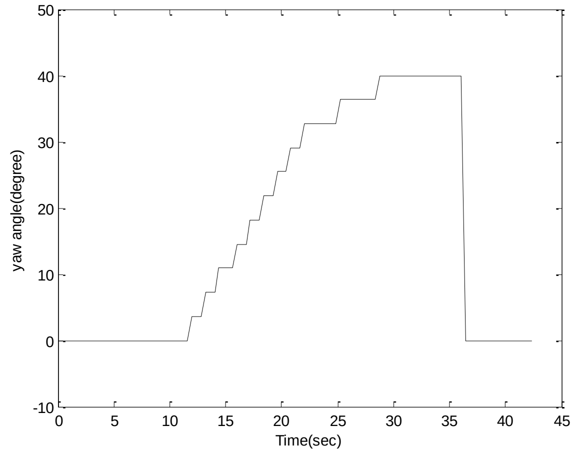

Fig.1 Input data(Steering wheeling angle)

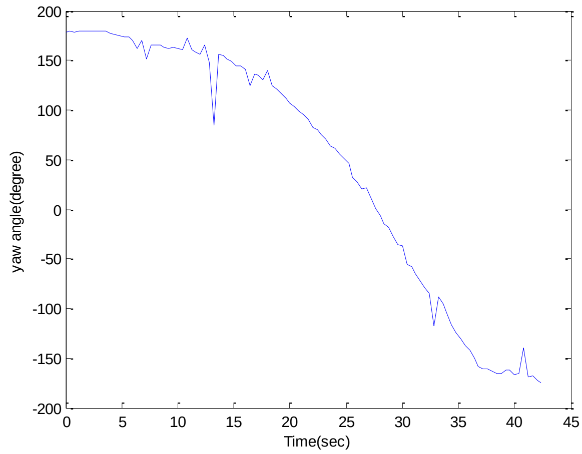

Fig.2 Output data(yaw angle)

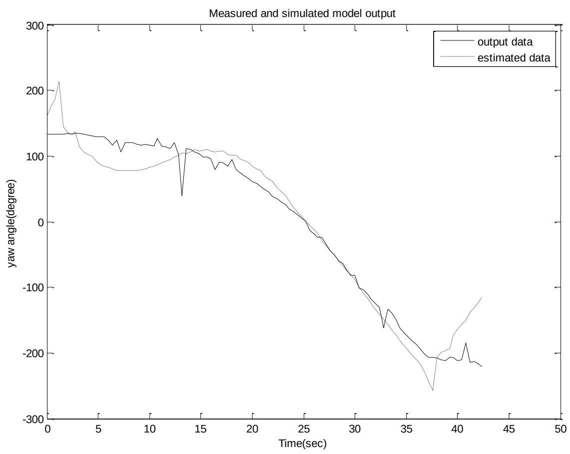

Fig.3 Performance of subspace identification

The discrete-time state equations in sampling time T=0.01sec is the following (Eq.6).

$$
\begin{aligned}
A &= \begin{bmatrix} 0.016214 & 0.17667 & -0.74701 \\ 0.12678 & 1.3821 & -4.8858 \\ 0.42002 & 5.1668 & -11.208 \end{bmatrix} \\
B &= \begin{bmatrix} 0.000086638 \\ 0.0085629 \\ 0.0027416 \end{bmatrix} \\
C &= \begin{bmatrix} -1062.4 & 2.8312 & -44.839 \end{bmatrix} \\
D &= 0
\end{aligned}
\qquad (6)
$$

III. Obstacle detection method

The simplified version of our appearance-based obstacle detection method consists of the following four steps:

i) Filter color input image.
ii) Transformation into HSI color space.
iii) Histogramming of reference area.
iv) Calculation the back project of the histogram.

In the first step, the 320×260 color input image is filtered with a 5×5 Gaussian filter to reduce the level of noise.

In the second step, the filtered RGB values are transformed into the HIS (hue, saturation, and intensity) color space. Because color information is very noisy at low intensity, we only assign valid values to hue and saturation of the corresponding intensity is above a minimum value. Similarly, because hue is meaningless at low saturation, hue is only assigned a valid value if the corresponding saturation is above another minimum value. An appealing attribute of the HSI model is that it separates the color information into an intensity and a color component. As a result, the hue and saturation band are less sensitive to illumination changes than the intensity band.

In third step, a trapezoidal area in front of the mobile robot is used for reference. The valid hue and intensity values of the pixels inside the trapezoidal reference area are histogrammed into two one-dimensional histograms, one for hue and one for intensity. Histograms are well suited for this application, as they naturally represent multi-model distributions. In addition, histograms require very little memory and can be computed in little time.[4]

In the fourth step, the backprojection of histogram puts the value of the histogram bin, corresponding to the tuple in the output image, for each tuple of pixels at the same position of all input single-channel images (hue channel, intensity channel). In terms of statistics, the value of each output image pixel is the probability of the observed tuple given the distribution (histogram). The backprojection of hue and intensity histograms are united by OR operation.

A pixel is classified as an obstacle if the value of pixel is above the threshold in backprojection image. In the current experiment, the threshold is set to 100.

a) b)

Fig.4 a) Input color image with trapezoidal reference area b) output image

IV. Obstacle avoidance method

The software of the obstacle avoidance algorithm uses two queues: a candidate queue and a reference queue. From the central line, we search a trapezoidal area. A trapezoid area is stored in the candidate queue if the two following conditions are satisfied:

i) The sum of the value at the inner pixel is below the intensity threshold.
ii) The angle δ of the nonparallel side and the longer parallel side is bigger than 45 degree.

If a trapezoid is satisfied condition of the candidate queue, all pixel of current trapezoid gain weight at dimension of trapezoid. After trapezoid retrieval is over, a pixel is classified a reference queue if the weight of pixel is above the voting threshold.

The reference queue is chosen to path area. The center point of the top horizon line is selected as the navigation point. After the navigation point is chosen, the turn angle θ is calculated to be

$$\theta = \tan^{-1}\left(\frac{u_0 - \frac{u}{2}}{v_0}\right) \quad (7)$$

Where NP: (u_0, v_0) is the navigation point, u, v are width and height of image, respectively.

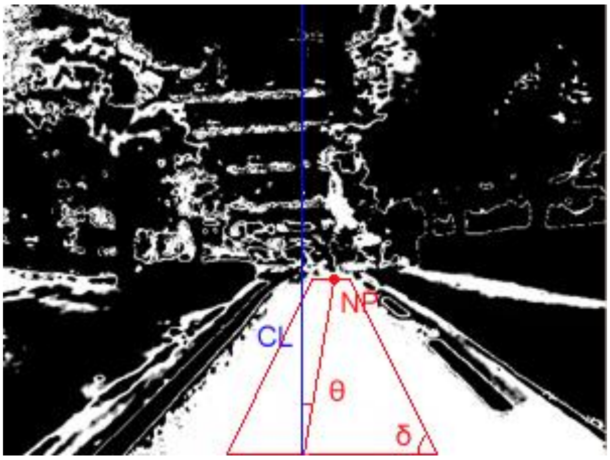

Fig.5 Illustration of how the candidate queue is chosen.

The turn angle θ inputs Eq. (6) and gets the predicted output data (the yaw angle of mobile robot) by system state equation. Until the error of current angle and destination angle is less than the past error, the turn angle input four times. In the current experiment, the intensity threshold is set to 5000.

V. EXPERIMENT

Fig. 6 shows some images and their results in several complex road environments. Fig 6 (a) shows a road image that include one red color area, which is candidate queue. NP_1 (the naviga-tion point) is set to be central and avoid lane that exist left side of image.

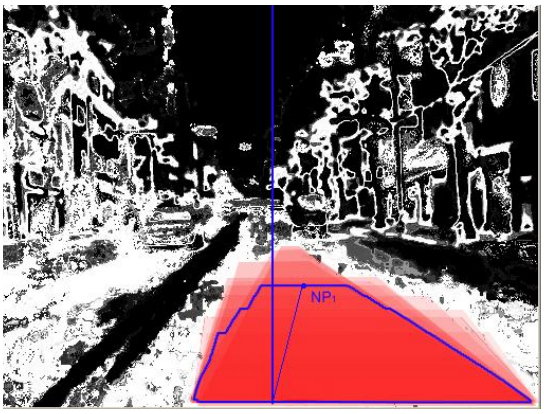

Fig.6 Experimental result

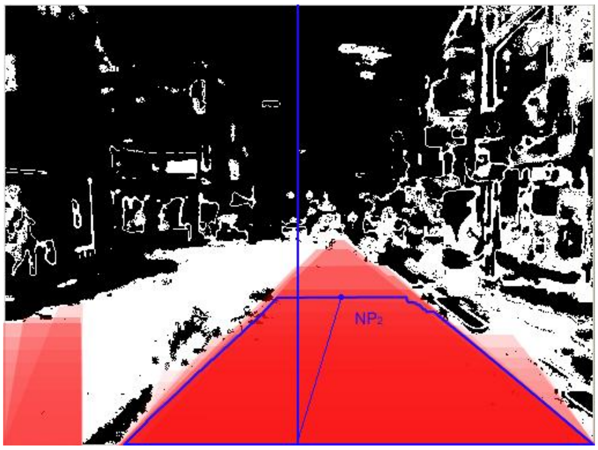

Fig.7 Experimental result

The pixel of blue line zone has over weight than weight threshold. As shown Fig.7, candidate queues of the left side are being dropped, because pixel's weight is below the threshold. As a result, the right navigation point can be calculated successfully.

VI. CONCLUSION

This paper presented a method for obstacle detection and avoidance with a single color camera. The method performs in real-time and provides the path area, the navigation point, and the turn angle. Using the system state equation, the turn angle is closed to the destination yaw angle gradually. The system can be trained and has performed well in outdoor environments.

REFERENCES

[1] Kuang-Hsiung C, Wen-Hsiang T, "Vision-based obstacle detection and avoidance for autonomous land vehicle navigation in outdoor roads," *Automation in Construction*, Vol. 10, pp. 1-25(2000)

[2] Seong Man Yoon, Seong Taek Hwang, and Jae Heon Ryu and Man Hyung Lee, "System Identification and Controller Design for Lateral Control of an Unmanned Vehicle", *The Thirteenth International Symposium on Artificial Life and Robotics* 2008(AROB 13th '08), 2008

[3] J. Poshtan and H. Mojallali, "Subspace system identification," *Iranian Journal of Electrical & Electronic Engineering*, Vol. 1, No. 1, pp. 11-17, January 2005.

[4] I. Ulrich and I. Nourbakhsh, "Appearance-Based Obstacle Detection with Monocular Color Vision," *Proceedings of the AAAI National Conference on Artificial Intelligence*, July/August 2000