# Point Number Algorithm for Position Identification of Mobile Robots

## 로봇의위치계산을위한포인트개수알고리즘

유 강*, 손 영 익**, 김 갑 일***
Liu Jiang, Young-Ik Son, Kab-Il Kim

**Abstract** - This paper presents the use of Point Number Algorithm (PNA) for real-time image processing for position identification of mobile robot. PNA can get how many points in the image gotten from the robot vision and can calculate the distance between the robot and the wall by the number of the points. The algorithm can be applied to a robot vision system enable to identify where it is in the workspace. In the workspace, the walls are made up by white background with many black points on them evenly. The angle of the vision is set invariable. So the more black points in the vision, the longer the distance is from the robot to the wall. But when the robot does not face the wall directly, the number of the black points is different. When the robot faces the wall, the least number of the black points can be gotten. The simulation results are presented at the end of this paper.

**Key Words** : Point Number Algorithm, real-time image processing, robot vision system

### 1. Introduction

Vision system is the primary sensor input to advanced robotic applications such as mobile robots, agricultural robots and inspection systems [1]. However, vision causes a significant processing burden in intelligent robotics since the majority of image processing algorithm are computationally extremely expensive. For this reason any improvement in the vision algorithm leads to significant benefits in terms of achieving real-time performance [2-4].

In general, image processing algorithms are applied to uncompressed images. The reason for this is that most of the popular image compression techniques distort the color, size and position of the objects in the image that can only be recovered by decompressing the image. Processing of compressed images in real time also is problematic when the compression algorithm is computationally expensive. Messom etc. [5] examined the processing of compressed images in real-time using the Run Length Encoding algorithm. This algorithm is not computationally expensive. It allows an image to be encoded with just a single access to each pixel. But it also need a lot of time to compress and decompress the image.

Messom etc. [6] reported a real-time vision system using incremental tracking, whereby a window around each object is created. Once the location of an object has been determined, its movements are tracked only within the window. The position of this window is adjusted each frame after identifying the new position of the object. This technique is used mainly due to the saving it offers in processing time, as the whole image of the playing field does not need to be analyzed. Instead, only the tracking windows for each object need to be analyzed. The technique analyzes a very small proportion of the field, leading to the intermittent problem of losing objects that stray outside their tracking windows. This problem is particularly hard to solve for fast moving unpredictable objects. Nevertheless, this method was used in order to achieve real-time image processing.

In this paper, a new method called Point Number Algorithm for self localization of mobile robot by real-time image processing is proposed in order to save the processing time. This paper is organized as follows. In Section 2, the proposed algorithm is presented. the computer simulation results of the proposed approach are presented in Section 3. Finally, some conclusions are summarized in Section 4.

### 2. The Proposed Algorithm

PNA is short for Point Number Algorithm. This algorithm is applied in 3-dimensional workspace. The reason we call the proposed algorithm as PNA is that the main work of this algorithm is to calculate the number of black points in the image which we got.

The whole process of PNA is as follows:

* 劉 江 : 明知大學校 電氣工學科 博士課程
** 孫 瑛 翼 : 明知大學校 電氣工學科 助教授・工博
*** 金 甲 一 : 明知大學校 電氣工學科 教授・工博

First, we can get an RGB image of the background from a camera, as shown in Fig.1. In order to calculate how many black points in it, we should transform the RGB image to gray scale image, and then transform to binary image at last as shown in Fig.2 and Fig.3. Because the image is made up by black and white, we can't distinguish the difference between Fig.1 and Fig.2 by eyes. For Fig.3 we used the reverse transformation. From Fig.3 we can directly get the number of black points in the image by Matlab Toolbox. After that, by the ratio of the number of black points and the distance from the robot to the wall, we can calculate out the position of the robot in the work space. The details are as follows:

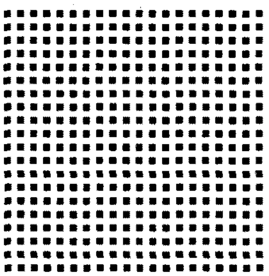


Fig.1. RGB image gotten from the camera

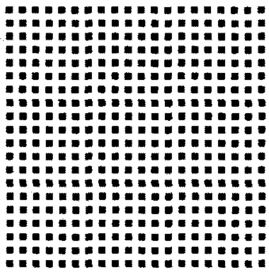


Fig.2. Gray scale image

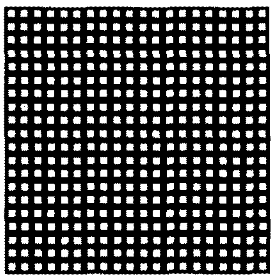


Fig.3 Binary Image

Before we calculate the distance, we set the origin of the coordinate to the lower-left quarter on the ground. And the walls are made up by white background with many black points on them evenly- N black points in $1m^2$ area. Assume the current black point we get is n. The sight of the robot is a rectangular pyramid whose angle between the vertical line and slant is alpha ($\alpha$). The bottom of rectangular pyramid is square.

$$\frac{N}{1} = \frac{n}{S} \qquad - (1)$$

S is the area of current image.

$$Dis = \frac{\sqrt{S}}{2tan(\alpha)} = \frac{\sqrt{n}}{2\sqrt{N}tan(\alpha)} \qquad - (2)$$

$Dis$ is the distance from the robot to the walls.

But here is a problem: when the robot does not face the wall directly, we can get more black points than that when the robot face to the wall directly. This will lead to a wrong result. In order to solve this problem, we take several picture for the wall as the robot rotates. Then we can get the image who contains the least black point. So this is the image which is taken when the robot face the wall.

## 3. The Simulation Results

Computer simulations have been performed to test the proposed algorithm in a normal environment.

Here are some enactments;

1. The work space is 3-D. The size of the ground is $5.5m*5.5m$; the size of wall is $5.5m*3m$.
2. There are 1628 points in $1m^2$. N = 1628.
3. The angle between the vertical line and slant of the rectangular pyramid of the robot sight is $\alpha = \pi/12$.
4. The height of vision system of mobile robot is $1.5m$ which is near to human eyes.
5. The nearest distance to the wall is 0.2m because of the safety.

3.1 When the robot faces to the wall directly.

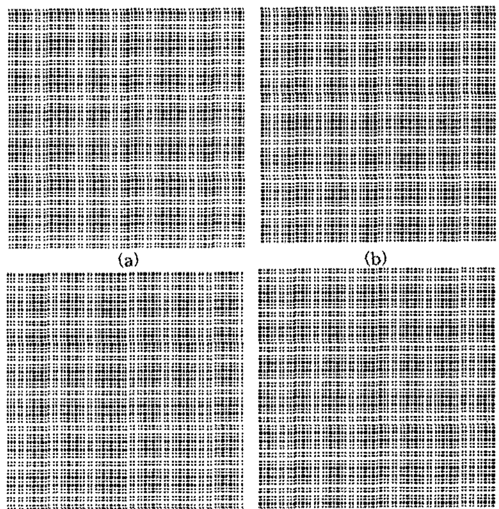


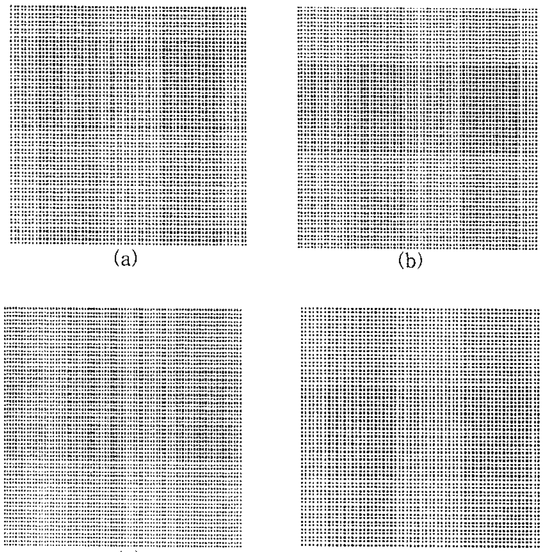

Fig.4. Simulation results

In Fig.4(a), n = 3782. Dis = 2.8441.
In Fig.4(b), n = 3721, Dis = 2.8211.
In Fig.4(c), n = 3844, Dis = 2.8674.
In Fig.4(d), n = 3782. Dis = 2.8441.

After hundreds of test, we proved that the error is $\pm 0.025$.

3.2 When the robot does not face the wall directly.

When the robot does not face the wall, we will take several pictures and calculate all of them to get the image which has the least number of black points.



(a) (b) (c) (d)

Fig.5. Simulation results

In Fig.5(a), n = 5146. when the angle between the vertical line of the robot sight and the wall is 75 degree.

In Fig.5(b), n = 4599. when the angle between the vertical line of the robot sight and the wall is 80 degree.

In Fig.5(c), n = 4154. when the angle between the vertical line of the robot sight and the wall is 85 degree.

In Fig.5(d), n = 3844. when the angle between the vertical line of the robot sight and the wall is 90 (vertical to the wall) degree. From (d) we can get the least number of black points.

## 4. Conclusion and Future Work

In this paper, a new algorithm called Point Number Algorithm has been proposed for self-localization of mobile robot in real-time image processing. This algorithm takes less time than others, but it need more work to make the background.

In the future, we will try to utilize the nature information of the work space to realize self-localization.


## Acknowledgment

This work was supported by the ERC program of MOST/KOSEF (Next-generation Power Technology Center)


## Reference


[1] R.Templer, H.Nicholls and T.Nicolle, "Robotics for meat Processing from research to commercialisation", Industrial Robot, Vol. 26, No. 4, 1999.

[2] J.Hrabec and B.Honzik, "Mobile robots playing soccer", Advanced Motion Control, 2002. pp. 510-513, July 2002.

[3] A.Treptow and A.Zell,"Real-Time object tracking for soccor-robots without color information", Robotics and Autonomous Systems 48, pp. 41-48, 2004.

[4] G.S.Gupta, C.H.Messom and S.Demidenko, "Real-Time Identification and Predictive Control of Fast Mobile Robots Using Global Vision Sensing", IEEE Transactions on Instrumentation and Measurement, Vol. 54, No. 1, 2005.

[5] C.H.Messom, S.Demidenko, K.Subramaniam and G.S.Gupta, "Size/Position Identification in Real-Time Image Processing using Run Length Encoding", IEEE Instrumentation and Measurement Technology Conference, Vol. 2, pp. 1055-1060, May 2002.

[6] C.H.Messom, G.S.Gupta and H.L.Sng, "Distributed Real-Time Image Processing for a Dual Camera System", CIRAS 2001, Singapore, pp. 53-59, 2001.