# Recognition and Movement in Artificial Environment with Bipedal Robot

Naohiro OHTSU, Norihiro ABE, Kazuaki TANAKA, Yoshihiro TABUCHI
Kyusyu Institute of Technology
680-4 Kawazu, iizuka, Fukuoka 820-8502, Japan
Email:ohtsu@sein.mse.kyutech.ac.jp

***Abstract***: However well we control a bipedal robot in walking, images gotten by the cameras are tilted by left or right swing and small bumps. This complicates recognition of environment by using cameras in walking, and the robot can't move smoothly. Introducing a bipedal robot is to make it possible for both a human and the robot which have the same embodiment to make collaboration by making both take the same behavior as far as possible which is difficult to attain with other types of robots such as a wheel driven robot. Under the artificial environment mainly consisting of vertical and horizontal lines, images must be corrected by detecting the tilt angle of them using Hough Transformation, which detects nearly-vertical lines from artificial environment. As a result, a robot successfully recognizes the environment with a stereo vision using images obtained by correcting tilted ones.

***Keywords***: Bipedal Robot, Image Processing, Tilt, Correction.

## I. INTRODUCTION

In recent years, high-performance and excellent bipedal robots have been developed to support humans. Many of the robots recognize environment with cameras because the robots can acquire much information easily. Although the robots can get much information, even now the robots can't recognize around smoothly. The robots stop in front of obstacles, and then move again. One of the reasons is that recognition of the environment with cameras mounted on the robot is very difficult. However well we control the robots in walking, images gotten with the cameras are tilted by left or right swing and small bumps. This complicates recognition of the environment because it is too difficult to apply general image processing methods to such tilted images directly. Thus, the robots can't get any precise information from the images (distance between a robot and obstacles, and their shapes, etc) easily.

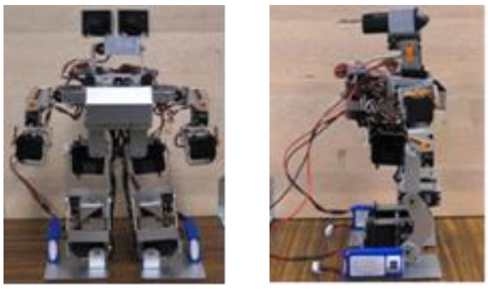
As a result, a robot can't move smoothly. This makes it difficult to make a human cooperate with it. Therefore, a method for correcting the tilted images is required.

One of ideas to correct the images is to use sensors which can detect a physical tilt angle. By using these sensors, a robot can correct the images by rotating them at the angle. However, the precision of the angle depends on mechanical characteristic of the sensors, and the acquisition of sensor values must be synchronized with that of images. Thus, usage of sensors is not suitable for the current purpose because the camera used is of not external but internal synchronization.

In our research, a bipedal robot is equipped with stereo cameras on its head, and images are corrected using a tilt angle detected from them directly with Hough Transformation based on the fact that it moves in artificial environment.

## II. BIPEDAL ROBOT

In our research, a bipedal robot as shown in Fig.1 is used to examine whether it can over stride or go up on and down an obstacle safely.



[a] Front View [b] Side View
Fig.1 Bipedal Robot

The specifications of the robot are shown in Table.1.

Table.1 Specifications of the Robot

| | |
|---|---|
| Size | W240×D140×H345mm(at stand up right) |
| Weight | 1720g (not contain the batteries) |
| Degrees of Freedom | 20 (Head:2 Arm:6 Leg:12) |

## III. SYSTEM CONFIGRATION

The system configuration constructed for this research is shown in Fig.2.

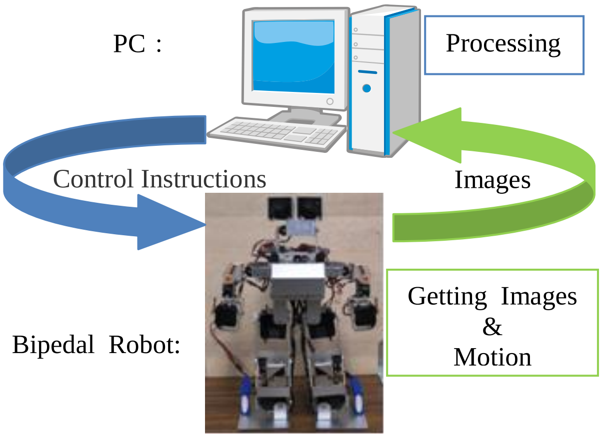



Fig.2 System Configuration

In this system, images captured with cameras are sent to PC and commands to a robot are sent through wireless communication.

## IV. LENS DISTORTION

A distorted image as shown in Fig.3[a] is captured with a camera mounted on a robot.

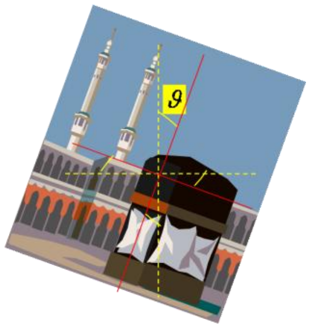
[a] Before Correction

[b] After Correction
Fig.3 Lens Distortion

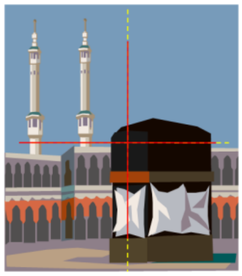
This distortion can be calibrated with an image processing method. As a result, the image as shown in Fig.3[b] is obtained.

## V. IMAGE PROCESSING

### 1. Methods for Correcting Tilted Images



[a] Before Correction



[b] After Correction
Fig.4 Image Blurring

Even when a robot walks on a flat plane, images acquired are tilted as shown in Fig.4[a]. Calculating the tile angle and rotating images as shown in Fig.4[b] by the angle makes it easy to measure the distance to obstacles with an ordinal stereo vision.

Artificial environment includes many static structures of which edges are vertical or horizontal lines. Of course, there are other lines constituting other objects, but the length of them is less than that of vertical lines of walls or doors.

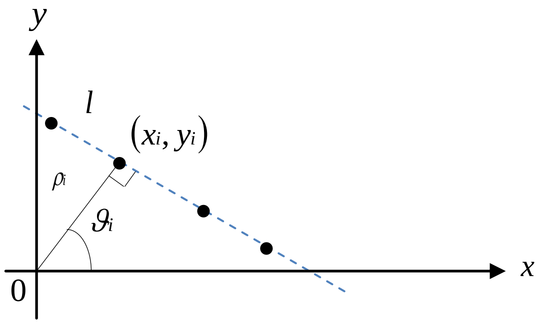
In Fig.4[a], if lines corresponding to the vertical or horizontal lines constituting the artificial environment can be detected, the tilt angle can be found.

In other words, as the longest nearly-vertical line in an image corresponds to a real vertical line of a static artificial structure, detecting it gives the tilt angel.

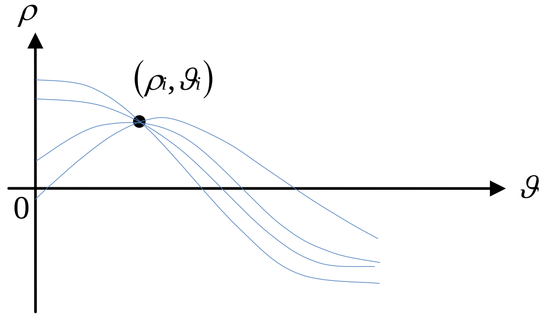
The number of points on the line, its gradient and the length of a perpendicular on the line from the origin are enough to find the line. They are easily determined with Hough Transformation.

### 2. Hough Transformation

Using a derivative operator, a sequence of points on an edge shown in Fig.5[a] are detected.




[a] $xy$ Image Space




[b] $\rho\vartheta$ Parametric Space

Fig.5 Mapping to $\rho\vartheta$ Parametric Space

Here, each point $(x_i, y_i)$ on a line as shown in Fig.5[a], can be mapped to one sinusoidal wave on a $\rho\vartheta$ parametric space shown in Fig.5[b] which is expressed by equation.(1) when $\vartheta_i$ is changed at a regular interval.

$$\begin{aligned} \rho_i &= x_i \cos\vartheta_i + y_i \sin\vartheta_i \\ &= A\sin(\vartheta_i + \alpha) \end{aligned} \qquad (1)$$

$\rho_i$ : the length of a perpendicular on the line

$\vartheta_i$ : a gradient angle of a perpendicular ($0 \leq \vartheta_i < \pi$)

$$A = \sqrt{x_i^2 + y_i^2}, \alpha = \cos^{-1}(y_i / A)$$

The $\rho\vartheta$ parametric space is composed of a set of small cells, and a counter value of each cell is added by one every time the point with the same $(\rho_i, \vartheta_i)$ is found. This processing is called voting.

As a result, the result as shown in Fig.5[b] is obtained after voting against the all points.

Based on the above theory, length of each detected lines is calculated as follow. Length of each line is expressed as a voting number because the number is relative to the length.

Each line detected has a unique $(\rho_i, \vartheta_i)$ as a parameter, $\vartheta_i$ corresponds to the gradient of the line.

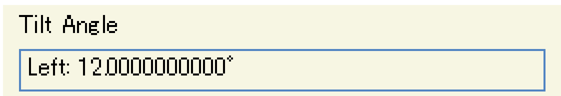
Based on experiment conducted using the robot, 30 degrees is set as the maximum angle of inclination. To detect only lines of which gradient is within the limit, as cells beyond the permissible rage are not taken into count, time needed for performing Hough Transformation is expected to be reduced than the conventional one, but at present it is not implemented.

## VI. DETECTION OF THE LONGEST LINE AND THE TILT ANGLE

Here the result obtained with Hough Transformation is shown.

[a] Detection of the Longest Line

[b] Detection of the Tilt Angle




[c] Values of Tilted Angle in the [b]

Fig.6 Detection of the Tilt Angle

Fig.6[a] shows the longest line in the given image, Fig.6[b] shows the image in which the longest line is successfully detected when a robot leans toward right about 12 degrees from a vertical line. The tilt angle is shown in Fig.6[c].

Finally, Fig.7 shows the image obtained by rotating the image shown in Fig.6[b].

Fig.7 Corrected Image

At present, there are cases where Hough Transformation detects not a real but false line as shown in Fig.8, because a long edge of a display is accidentally the longest near-vertical line.

Fig.8 Image of False Detection

The reason is that the system regards an edge of a display as vertical line because it is within 30 degrees. This happens because position of the camera is low and at quite a near location relative to the display. As the height of a robot is low, such an accidental case will often happen.

Therefore, the false detection must be avoided using following methods. Even if a real vertical edge of a wall or column is occluded by objects near a camera, the line will be detectable by searching the upper part of the image, unless the camera is too close to the objects. Usually a robot should not approach but avoid an object unless it must grasp the object. Therefore, it is thought possible to avoid false detection by only detecting vertical lines from the upper part of an image (In our research, we try to detect the lines from the upper quarter part as shown in Fig.9).

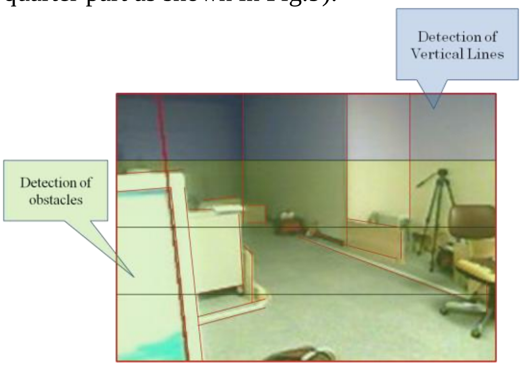



Fig.9 Avoiding False Detection and Detection of the Obstacles

In addition to this, it is thought that edges of obstacles tend to be in the lower part of the image and edges of polyhedral objects like furniture or equipments are orthogonal with vertical lines (see Fig.9). These facts allow Hough Transformation to find polyhedral obstacles from the lower region of the image. Of course, it is impossible to detect edges of other obstacles, but any artificial objects include straight edges except for a solid sphere, such edges will useful to obstacle recognition.

By subtracting all line segments from edges extracted with a derivative operator, other types of edges can be extracted. Consequently, straight edges and the rest edges are separately available to calculate distance information with a stereo vision method.

We try to avoid false detection and detect of the obstacles with these methods.

Exploiting multi core PC makes it possible to implement image processing using parallel computation.

Further, application of GPU should be also tried to make much faster computation possible than that of CPU.

## REFERENCES

[1] Okutomi M, Ozawa S, Shimizu M, Hori O, Ejiri M, Kitamoto A, Goto M, Saito H, Sato Y, Chiba N, Naoi S, Ninomiya T, Fujiyoshi H, Miyahara K, Morisaki K, Yamanouchi Y (2004), Digital image processing (in Japanese). pp.211-214

[2] Forsyth DA, Ponce J (2003), Computer Vision: a Modern Approach.

[3] Nara Institute of Science and Technology OpenCV Programming Book Creative Team (2007), OpenCV Programming Book (in Japanese).

[4] Yasuda K, Kobayashi S, Iiduka H, Abe T, Aoyagi S (2007), Parallel Programming for Multi-Core CPU (in Japanese).

[5] Minami Sato, Yoshihiro Tabuchi, Norihiro Abe, Hirokazu Taki, Shoujie He: "An Autonomously Moving Robot Using Network Cameras", International Symposium on Artificial Life and Robotics (AROB 2008).

[6] Akihiro Fujiwara, Norihiro Abe, Yoshihiro Tabuchi, Hirokazu Taki, Shoujie He: "Control of Autonomous Mobile Robot with Laser Sensor and Image Processing Approach", WMSCI2009.