https://doi.org/10.20965/jaciii.2010.p0055

Paper:

Omnidirectional Vision for Mobile Robot Navigation

Zahari Taha, Jouh Yeong Chew, and Hwa Jen Yap

Department of Engineering Design and Manufacture, University Malaya,
50603 Kuala Lumpur, Malaysia
E-mail: {zahari_taha, hjyap737}@um.edu.my, jouhyeong@perdana.um.edu.my

[Received April 25, 2009; accepted Nobember 7, 2009]

Machine vision has been widely studied, leading to the discovery of many image-processing and identification techniques. Together with this, rapid advances in computer processing speed have triggered a growing need for vision sensor data and faster robot response. In considering omnidirectional camera use in machine vision, we have studied omnidirectional image features in depth to determine correlation between parameters and ways to flatten 3-dimensional images into 2 dimensions. We also discuss ways to process omnidirectional images based on their individual features.

Keywords: omni-directional vision, mobile robot, localization, navigation, color detection

1. Introduction

How mobile robots may potentially play roles in hazardous, repetitive tele-operated or automated tasks has attracted much attention. We have focused on automated mobile-robot movement involving efficient, automated standalone movement through sensor assistance. With the wide variety of sensors now available, robots can obtain the information they need about their surroundings and react accordingly. Vision sensors have undergone wide study due to the humongous amounts of data they provide. Vision systems usually consist of "eyes" – one or more cameras – coupled to a "brain" – or processor – to handle information obtained. Effective hardware functioning requires (usually) embedded intelligent image recognition and processing.

Intensified research and success in this field has been largely limited to developed nations, among which examples include Honda's Asimo and Sony's Aibo which use vision-based navigation only as a part of other intelligence. Work in developing countries has just reached the autonomous level after great time and effort. The leap from manual to tele-operated robots is itself being made where researchers combine microprocessors and microcontrollers to overcome the computational limitation of microcontrollers alone although mobile robot space and resource restrictions make implementing both microcontrollers and microprocessors impractical especially for compact robots. Implementing microprocessors at a separate workstation would require wired control although an alternative is to use wireless data transmission from the workstation microprocessor to the robot microcontroller.

With the importance of mobile robot roles widely recognized, the fast pace of development begs for better effectiveness and efficiency, higher work and product quality, improved reliability, less human labor in dangerous and hazardous tasks and lower operating cost especially in semi- and fully automated factories. While automation is widely viewed as the solution to these problems, crucial factors hindering its implementation are limited space and energy resources and the cost of controllers. Mobile robots are widely used as Automated Guided Vehicles (AGV) in factories and as fire-fighters and security guards – all applications requiring robots to detect their own locations, headings and potential obstacles – information crucial to effective autonomous navigation and task completion.

The most suitable sensors available for acquiring this wide variety of information are vision sensors. In global navigation, ceiling cameras acquire views of entire environmental layouts. Cameras on robots informing their hosts of local navigation details are the focus here, which brings up the issue of the narrow field of view (FOV) [1]. We propose using omnidirectional cameras which have a 360° FOV to cover the navigation environment. This FOV has increased omnidirectional camera use among vision researchers. Kohsia used an omnidirectional camera to calculate the driver's view of the vehicular environment, presenting multistate statistical decision models with Kalman-filter based tracking to detect head positioning and calculate face orientation [2]. Kim presented Simultaneous Localization And Map building (SLAM) using omnidirectional stereovision to provide robust calculation and minimize the effects of motion drift [3].

Ulrich introduced appearance-based place recognition for topological localization, classifying real-time color imaging based on nearest-neighbor learning, image histogram matching and voting while obtaining 87-98% accuracy [4]. However, information explaining the basics of this wonderful invention remains insufficient. This paper's main objective is to address this issue and it is organized as follows: Section 2 details the mobile robot – The OmniBot and Section 3 describes omnidirectional camera features. Section 4 gives the experimental setup and results determining how different omnidirectional camera parameters correlate. Section 5 discusses robot localization and ways to process omnidirectional images and Section 6 presents conclusions and projected work.

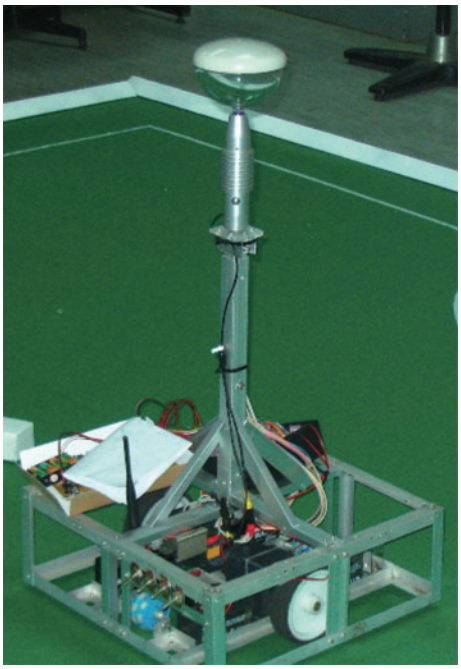


Fig. 1. The OmniBot using differentially driven wheels.

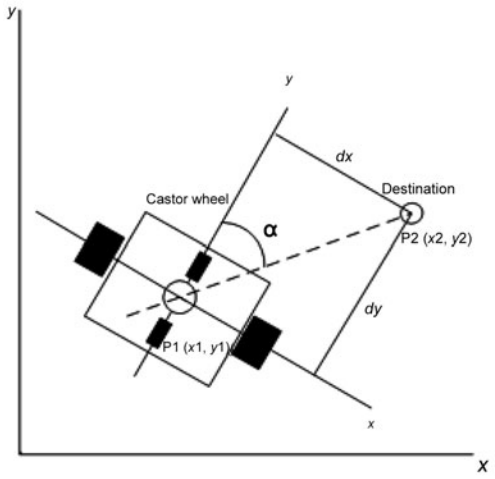


Fig. 2. Model of the OmniBot.

Table 1. Locomotion of the OmniBot.

Motion	Left Wheel		Right Wheel	
	Rotation	Speed	Rotation	Speed
Forward	CCW	Same	CW	Same
Reverse	CW	Same	CCW	Same
Left	Stop	Zero	CW	Same
Right	CCW	Same	Stop	Zero
Sharp Left	CW	Same	CW	Same
Sharp Right	CCW	Same	CCW	Same
Arc (Center point on left)	CCW	Slow	CW	Fast
Arc (Center point on right)	CCW	Fast	CW	Slow

CCW = Counter Clockwise
CW = Clockwise

2. The OmniBot

The OmniBot used in experiments and shown in **Fig. 1** is based on differential-driven wheeled mobile robots whose movement is controlled using independent left and right wheels supported front and back by 2 transwheel casters that swivel freely in the local x and y directions [5]. The transwheel's hard contact surface reduces friction [6]. The weight of the OmniBot is distributed equally on the platform to enhance robot stability. This design better controls the movement of the OmniBot in navigation even under space limitations. This can be further explained by the ability of the OmniBot to make a turning even without having to make any forward or reverse movement. It only needs to move both left and right wheels in the same direction to turn at a location. This movement is commonly called "hard" or "sharp" turns [6].

The properties of the OmniBot on a planar surface are shown in **Fig. 2**. Two motor-driven center wheels provide traction. Front and back casters freely rotate to support the OmniBot which is located in global coordinates at point P1. The x-axis of P1 is $x1$ and that of the y-axis is $y1$ which point lies at the center of the motor-driven wheel axle. The α-angle is the heading of the OmniBot as related to the y-axis in global coordinates – parameters represented by the following vector:

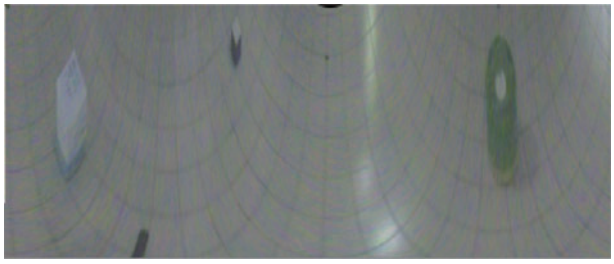
$$q = \begin{bmatrix} x1 \\ y1 \\ \alpha \end{bmatrix}. \quad \ldots\ldots\ldots\ldots\ldots\ldots (1)$$

This model has three degrees of freedom with nonholonomic constraints limiting local or instantaneous robot movement to two degrees of freedom.

This model assumes pure rolling, no friction, non-slippage and nonholonomically constrained to limit the OmniBot's speed but not its location [5]. Regarding the local coordinates in **Fig. 2**, the OmniBot only has velocity in the y direction and a rotation angle of α at any time. Velocity in the x direction is restricted with no displacement in that direction. The OmniBot can have velocity in the x and y directions in global coordinates as it can turn 90° with velocity in the new direction. Thus, the restriction on velocity does not limit the displacement in any direction. In short, local-coordinate movement and positioning are restricted but global movement and positioning are not.

Basically, the direction of individual wheel rotation and speed controls all the movements of the OmniBot. In order to move the OmniBot forward and backward, both wheels will be moving at the same speed but in the opposite directions. The OmniBot turns either by stopping one wheel or by turning both wheels in opposite directions. The OmniBot follows an arc by rotating both wheels in opposite directions at different speeds. An arc with the center of rotation on the left is produced by making the right wheel faster than the left wheel. **Table 1** shows basic locomotion of the OmniBot.

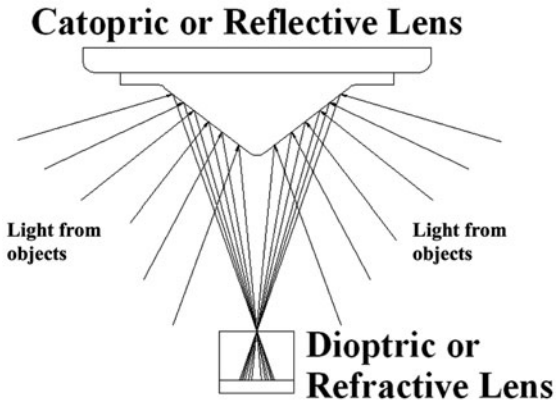
(a)



(b)

(c)

Fig. 3. (a) Original image from the omni-directional camera (b) panaroma image from the omni-directional camera and (c) addition of higher objects.



(a)

(b)

Fig. 4. (a) Side view of the camera-mirror system and (b) the mirror on the omni-directional camera.

3. Omnidirectional Cameras

This research uses an omnidirectional camera rather than a perspective camera installed on the OmniBot to obtain a 360° local view of the environment. Advantages of this include a greater FOV and enabling the OmniBot to learn the environment in the shortest time without excessive camera panning or tilt. The disadvantage is that omnidirectional camera images tend to be distorted by shape variation even in simple Cartesian conversion such as translation, scale or rotation. In an example of an omnidirectional camera image taken at the same height shown in **Fig. 3(a)**, the distorted surroundings are visibly doughnut-shaped with image distortion limited to object height. This means that an object's size increases with height, so objects on the same horizontal plane will have zero distortion as seen where grids formed by floor tiles remain undistorted except for slight barrel distortion which is commonly seen in perspective cameras [7].

Further observation shows individual objects in the image retain their top shape but at a different scale depending on height as Anderson found where physical object shapes remain intact on the ground plane [8]. The object size tends to grow with increasing distance from the image center as shown in **Fig. 3(c)** where additional objects are higher. The size of both the rectangular bracket and the stool increases with their respective height. The maximum object size is viewed if the object is shorter than the mirror height. **Fig. 3(b)** shows a panorama of the original image with images from the mirror's height at the center of the view. This explains distortion in images due to object height variation. **Fig. 4(a)** shows camera architecture and light ray transmission from the object into the camera.

4. Image Characteristic

Experiments were conducted to determine the relationship between pixel displacement and object height as shown in **Fig. 5**. The scale of real-world distance to pixel distance is measured, and then a wooden block is placed near the image center. The addition of identical wooden blocks provides a constant incremental height of 50 mm.

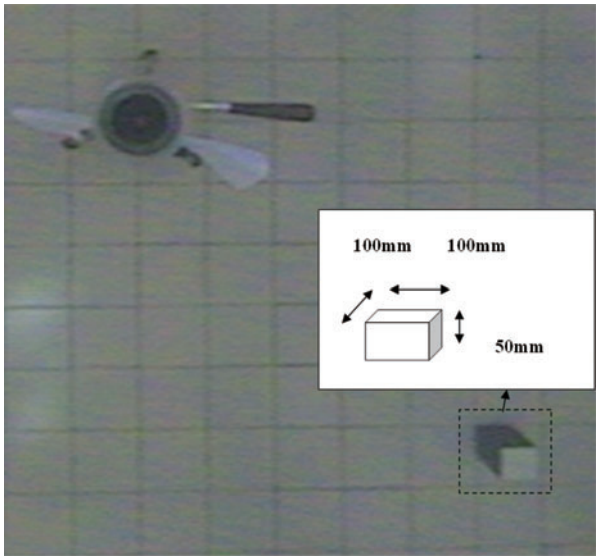


Fig. 5. The experimental setup and the blocks used in the experimental setup (inset).

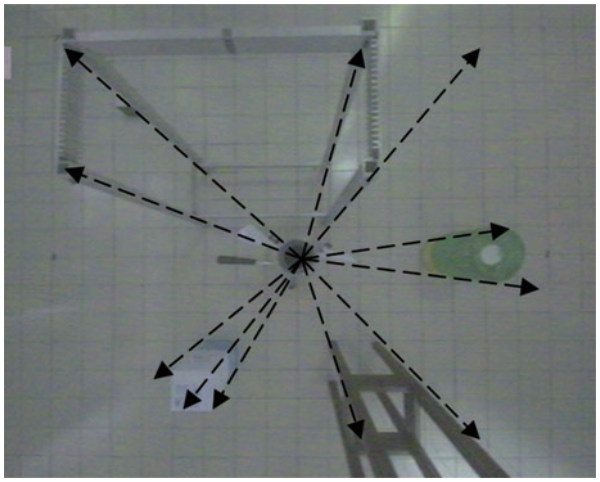


Fig. 6. Angle of pixel displacement relative to center of image.

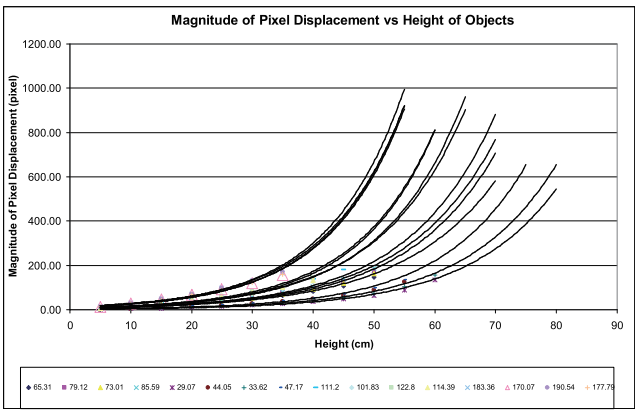


Fig. 7. The graph of pixel displacement versus height of objects.

Table 2. Coefficients a and b at different radius.

Pixel radius (pixel)	Real word distance (cm)	a	b
29.07	19.38	2.2888	0.0684
33.62	22.41	2.6189	0.069
47.17	31.45	2.8853	0.0723
65.31	43.54	6.5997	0.064
73.01	48.67	6.3398	0.0673
79.12	52.75	6.7933	0.0676
85.59	57.06	6.6439	0.0698
101.83	67.89	7.3444	0.075
111.2	74.13	8.4181	0.0719
114.39	76.26	6.7269	0.0798
122.8	81.87	7.9233	0.0772
170.07	113.38	11.391	0.0796
177.79	118.53	11.457	0.0798
183.36	122.24	12.94	0.0775
190.54	127.03	12.389	0.0798

The four base-corner coordinates are measured, and then the four coordinates of corners on the top of the first block are measured. This last measurement is being repeated when a new block is added until displaced pixels fall outside of the image. This process is repeated with wooden blocks shifted away from the image center to obtain a different image radius.

The data thus obtained is used to calculate the pixel displacement angle and the magnitude of pixel displacement. The first thing that this calculation shows is that the pixel displacement angle remains unchanged throughout incremental height as stated by Daneshpanah regarding angles preserved completely in a linear manner [9]. The pixel displacement angle thus depends solely on the object angle from the image center and any further displacement due to height is in the direction of the same angle obtained earlier as shown clearly in **Fig. 6**.

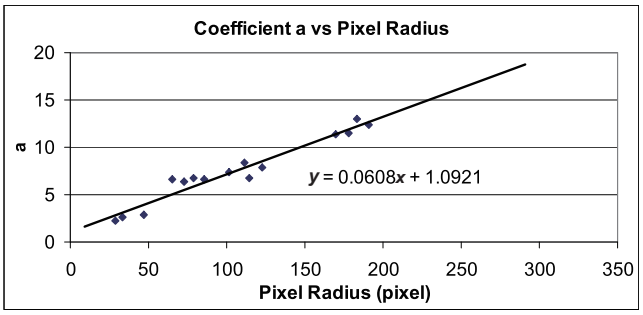
Regarding the magnitude of pixel displacement, the relationship between the object height and the magnitude of pixel displacement is an exponential function. The exponential function obtained at this point has different coefficients if the point is placed at different radius from the image center as shown in **Fig. 7** where each exponential function represents a different radius from the image center.

The relationship between the magnitude of pixel displacement and object height is generalized as follows:

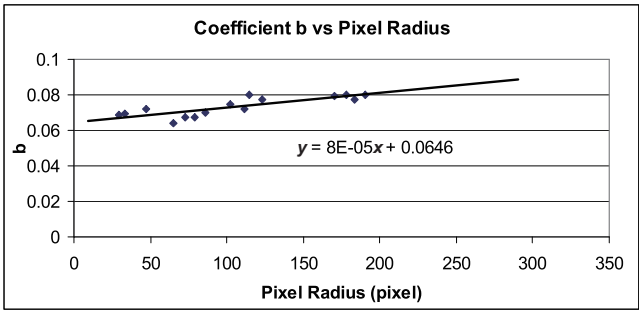
$$y = ae^{bx}. \quad \ldots\ldots\ldots\ldots\ldots\ldots \quad (2)$$

Where y is the magnitude of pixel displacement, x the object height, and a and b coefficients dependent on the object radius from the image center. Eq. (2) requires that we find the relationship between (i) pixel radius and coefficient a and (ii) pixel radius and coefficient b. Coefficients a and b values are calculated as follows for the object radius from the image center.

The result approximates the relationship between pixel radius and coefficients a and b as shown in **Fig. 8**.



(a)



(b)

Fig. 8. (a) Graph of coefficient a versus pixel radius and (b) graph of coefficient b versus pixel radius.

5. Experimental Setup

The transformation function for flattening omnidirectional camera images is approximated based on results obtained thus far. Markers are placed on environment landmarks and the combination of color and movement used to obtain their exact locations in local coordinates. This method is known as color blob tracking and robots need not recognize objects to complete tasks [10].

With a known height, pixel points from markers are converted to exact local coordinates enabling the OmniBot to know its global coordinate location by referencing the global coordinate of landmarks which are invariably within the view due to the omnidirectional camera's large FOV, enabling the robot's path to be planned as shown in **Fig. 9**. We confirmed the accuracy of this approach by comparing actual OmniBot axle and heading global coordinates to global coordinates calculated using the algorithm shown in **Tables 3** and **4**.

As shown above, a small deviation occurs between actual and calculated global locations of the OmniBot, possibly due to wireless transmission fluctuations and poor lighting. Such displacement deviation is invariably less than 0.4 at any location on the map, indicating that the OmniBot can locate itself within global coordinates using landmarks. Further experiments conducted to determine its heading in relation to a defined global location on the map showed that the OmniBot locates itself and finds its way to the defined global destination as shown in screen captures in **Fig. 10**.

This confirmed that the OmniBot approximates its global location in the global environment and navigates

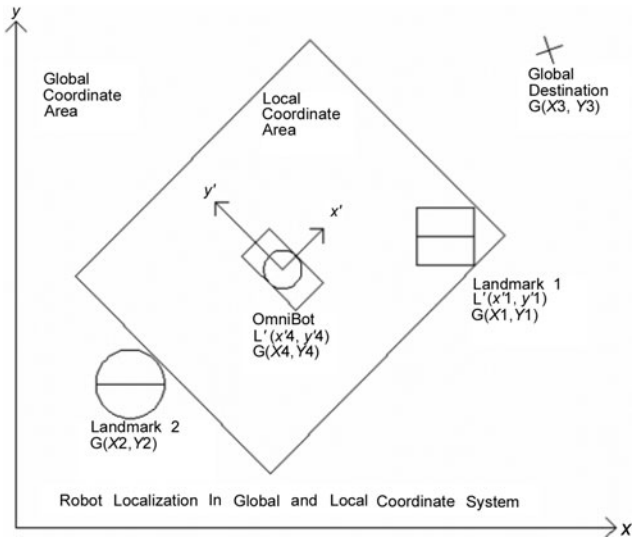


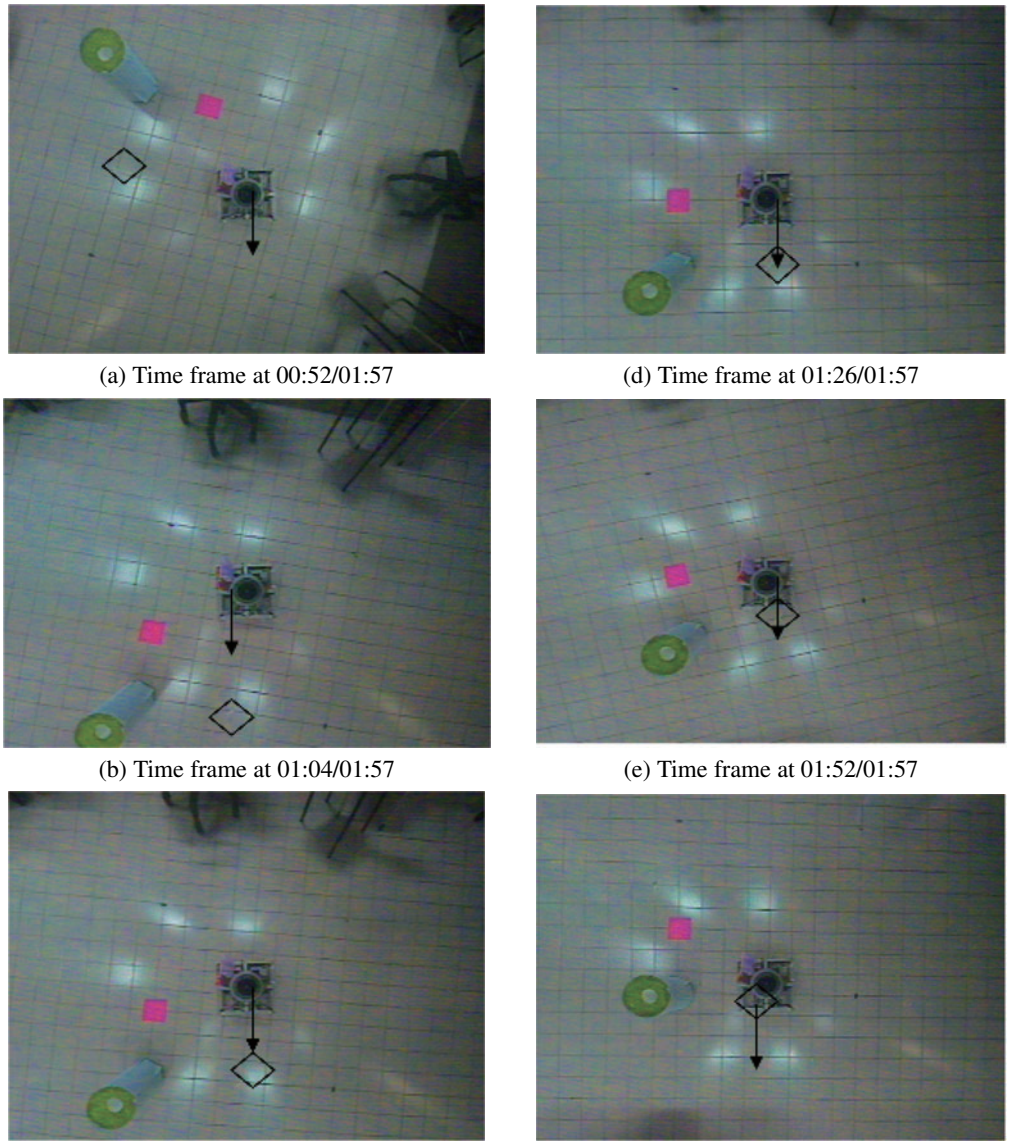
Fig. 9. Localization of the OmniBot by using landmarks.

Table 3. Global axle position.

Reading	Calculated Global Position		Actual Global Position		Error (Distance in Units)
	x	*y*	*x*	*y*	
1	7.5	3.0	7.4	3.0	0.100
2	12.5	3.5	12.3	3.2	0.361
3	12.0	4.8	11.7	5.0	0.361
1	11.5	7.9	11.5	7.6	0.300
2	12.0	9.2	11.9	8.9	0.316
3	11.8	11.0	11.8	11.3	0.300
1	6.0	12.5	6.0	12.8	0.300
2	4.0	12.0	3.9	12.3	0.316
3	3.5	10.5	3.6	10.2	0.316
1	4.0	7.8	3.9	7.8	0.100
2	2.9	6.0	3.0	6.3	0.316
3	4.3	3.2	4.6	3.2	0.300

Table 4. Global head position.

Reading	Calculated Global Position		Actual Global Position		Error (Distance in Units)
	x	*y*	*x*	*y*	
1	8.5	3.0	8.3	2.9	0.224
2	13.0	4.0	12.8	3.8	0.283
3	11.2	5.2	11.1	5.3	0.141
1	11.5	8.5	11.6	8.7	0.224
2	12.8	10.0	12.7	9.8	0.224
3	11.1	11.8	11.1	12.0	0.200
1	5.2	12.5	5.0	12.7	0.283
2	3.2	11.5	3.1	11.8	0.316
3	3.2	11.2	3.2	11.0	0.200
1	4.0	8.5	3.9	8.8	0.316
2	3.5	6.8	3.8	7.0	0.361
3	5.1	3.2	5.3	3.2	0.200



(a) Time frame at 00:52/01:57
(d) Time frame at 01:26/01:57
(b) Time frame at 01:04/01:57
(e) Time frame at 01:52/01:57
(c) Time frame at 01:13/01:57
(f) Time frame at 01:57/01:57

Fig. 10. Localization and navigation of the OmniBot (without obstacles).

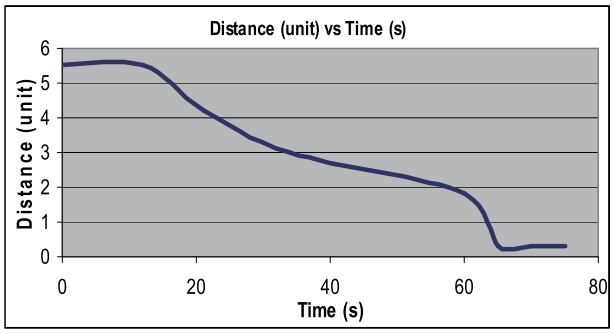


Fig. 11. Graph of distance from destination against time.

toward the defined global destination with minimal error. **Fig. 11** shows the robot's distance from the destination versus time. Final error is 0.3, the same as shown in **Tables 3** and **4**. **Fig. 12** shows robot heading angles versus time. Although the robot took much longer to reach a steady state, oscillation and steady state error are minimal and could be improved, for example by adding a Proportional-Integral-Derivative (PID) or fuzzy controller to the omnidirectional vision system. Error could also be minimized by replacing wireless vision system with an on board vision system to ensure more reliable vision transmission.

As yet, the OmniBot only navigates obstacle-free environments. For autonomous obstacle avoidance, the OmniBot must be able to differentiate between the floor and objects on the floor. Waypoints will be created in local navigation until the OmniBot reaches the global destination as shown in **Fig. 13** where the white blob is floor area for navigation and black blobs obstacles or areas that are not navigable. Red squares are possible navigation coordinates.

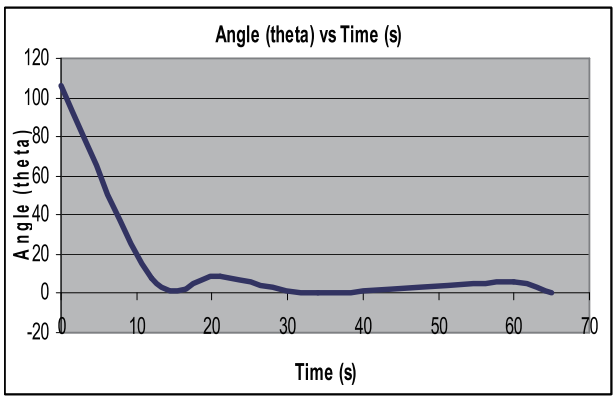


Fig. 12. Graph of heading angle against time.

Fig. 13. Floor finder.

6. Conclusions

Image distortion in omnidirectional camera occurs due to vertical object height and object radius from the image center. Thus, the transformation function required to flatten the image can be used to approximate the displaced coordinates useful in robot localization and navigation in a landmarked environment. However, the transformation function requires the height of the object to be known in the first place. Therefore, it is another effort to find out how to approximate the height of the object in the environment to provide greater autonomy to the system.

References:

[1] S. Benhimane and E. Malis, "A New Approach to Vision-based Robot Control with Omni-directional Cameras," Int. Conf. on Robotics and Automation, pp. 526-531, 2006.
[2] S. H. Kohsia, M. M. Trivedi, and T. Gandhi, "Driver's View and Vehicle Surround Estimation using Omnidirectional Video Stream," Proc. of Intelligent Vehicles Symposium, pp. 444-449, 2003.
[3] J. H. Kim and M. J. Chung, "SLAM with Omnidirectional Stereo Vision Sensor," Int. Conf. on Intelligent Robots and Systems, pp. 442-447, 2003.
[4] I. Ulrich and I. Nourbakhsh, "Appearance-Based Place Recognition for Topological Localization," Int. Conf. on Robotics and Automation, pp. 1023-1029, 2000.
[5] Appin Knowledge Solutions, "Robotics," Infinity Science Press, 2007.
[6] M. Gordon, "The Robot Builder's Bonanza," McGraw-Hill, 2001.
[7] Nanophotonics Co. Ltd., "Catadioptric Wide-angle Lens," http://www.nanophotonics.co.kr/, Access date: 10th July 2008 at 15.00.
[8] H. Andreasson and T. Duckett, "Object Recognition by a Mobile Robot Using Omni-directional Vision," Proc. of the Eighth Scandinavian Conf. on Artificial Intelligence, 2003.
[9] M. DaneshPanah, A. Abdollahi, H. Ostadi, and H. A. Samani, "Robotic Soccer," Itech Education and Publishing, 2007.
[10] M. J. Mataric, "The Robotics Primer," The MIT Press, 2007.

Name:
Zahari Taha

Affiliation:
Professor, Department of Engineering Design and Manufacture, University Malaya
Director, Centre for Product Design and Manufacure, University Malaya

Address:
50603 Kuala Lumpur, Malaysia

Brief Biographical History:
1995-1998 Research Fellow, Design Engineering Research Center, University of Wales Institute Cardiff
2000-2007 Professor, University Malaya
2008- Senior Profesor, University Malaya

Main Works:
• Prof. H. V. Brussel (ed), "Decoupled Control of Robots," Proc. 16th Int. Symposium on Industrial Robots, IFS (Publications) Ltd UK, Springer Verlag, pp. 1105-1117, 1986.

Membership in Academic Societies:
• The Institution of Engineering Designers, Corporate Member, 2001
• The Engineering Council UK, Corporate Member, 2001
• Academy of Sciences Malaysia, Fellow, 2008
• Asia Pacific Industrial Engineering and Management Society, Board Member, 2009

Name:
Jouh Yeong Chew

Affiliation:
Research Engineer, Robotics Lab, Centre for Product Design and Manufacture, University Malaya

Address:
50603 Kuala Lumpur, Malaysia

Brief Biographical History:
2005- Loans Operation Department, Deutsche Bank Malaysia
2006- Advanced Material and Manufacturing Processes Group

Name:
Hwa Jen Yap

Affiliation:
Lecturer, Department of Engineering Design and Manufacture, University Malaya

Address:
50603 Kuala Lumpur, Malaysia

Brief Biographical History:
1999 BP Malaysia (NPJ) Sdn. Bhd.
2000-2002 Matsushita Sdn. Bhd.

Main Works:
• "VR-based Robot Programming and Simulation System for an Industrial Robot," Int. J. of Industrial Engineering Theory and Application (IJIETAP), Vol.15 Issue 3, ISSN 1072-4761, pp. 314-322. (ISI net) (ISI-Cited Publication)