

# Aplikasi Metoda *Random Walks* untuk Kontrol Gerak Robot Berbasis Citra

R. Febriani, Suprijadi

Kelompok Keahlian Fisika Teoritik Energi Tinggi dan Instrumentasi

Fakultas Matematika dan Ilmu Pengetahuan Alam, Institut Teknologi Bandung

Jl. Ganesha 10, Bandung 40132, Indonesia

Telepon: (62) 22-2500834, Fax: (62) 22-2506452

*supri@fi.itb.ac.id*

Received date : 26 Februari 2010

Accepted date : 29 Mei 2010

## Abstrak

Algoritma merupakan bagian penting dalam sistem kontrol robot. Terdapat banyak metode yang biasa digunakan untuk keperluan ini, salah satunya adalah *random walks*. Metode ini menitikberatkan pada estimasi gerakan berdasarkan bilangan *random* (acak). Disisi lain aplikasi pengolahan citra masih belum banyak digunakan sebagai alat kontrol. Makalah ini akan membahas model dan simulasi metode *random walks* untuk kontrol gerak robot pada citra yang sederhana.

*Kata Kunci: random walks, citra*

## 1 Pendahuluan

Perkembangan teknologi robot dewasa ini sangat pesat, baik robot yang diam maupun bergerak ("mobile"). Hal ini disebabkan kemampuannya dalam mencapai tujuan dengan cepat dan tepat. Dalam pengaturan gerakannya, diperlukan suatu pengontrolan yang terdiri dari beberapa bagian yang terintegrasi. Terdapat tiga bagian utama yang membangun suatu sistem pengontrolan, yaitu input, algoritma, dan output [1]. Input merupakan sekumpulan informasi yang diterima oleh sensor pada robot, yang dalam hal ini akan diwakili oleh suatu kamera yang akan mengcapture suatu objek. Bagian kedua sebagai pembentuk sistem pengontrolan adalah suatu rule yang harus diikuti oleh robot dalam melakukan pergerakan. Rule ini biasanya disebut dengan algoritma [2]. Sedangkan bagian terakhir yaitu output, yang merupakan pencapaian akhir/pergerakan yang dihasilkan oleh robot tersebut.

Untuk membangun suatu sistem yang kompleks diperlukan penggabungan beberapa bidang ilmu. Hal ini juga diterapkan dalam kontrol gerak robot. Makalah ini memanfaatkan teknik-teknik yang terdapat pada pengolahan citra untuk aplikasi yang berbeda, yaitu kontrol gerak robot. Teknik yang akan digunakan adalah *edge detection* (deteksi tepi) yang pada pengolahan citra dapat dimanfaatkan untuk identifikasi objek. Pengolahan citra sendiri merupakan suatu teknik pembentukan citra baru yang berasal dari citra aslinya melalui berbagai operasi [3]. Pengolahan citra pada input akan dimanfaatkan untuk menempatkan objek (input) pada posisi tengah citra. Posisi tengah itu sendiri berkaitan dengan ukuran citra dan resolusi kamera yang digunakan. Ukuran citra disini dapat diartikan sebagai panjang dan lebar dari citra itu sendiri. Ukuran citra dapat ditentukan karena citra digital akan memiliki lebar dan tinggi tertentu. Ukuran ini biasanya dinyatakan dalam banyaknya titik pada sebuah citra [4].

Dalam pengontrolan yang berbasiskan citra ini, algoritma yang digunakan untuk lokalisasi posisi dari robot adalah suatu teknik estimasi random walks yang merupakan bagian dari metode Monte Carlo. Metode ini akan mengestimasi posisi berdasarkan pada probabilitas yang akan mungkin muncul [5][6][8].

Pada makalah ini akan dibahas tentang simulasi gerakan robot secara random walks yang berbasiskan citra untuk pengontrolan gerak dari suatu robot.

## 2 Edge Detection pada Citra

Edge atau tepi dari suatu objek dapat diartikan sebagai titik-titik yang memiliki perbedaan derajat keabuan yang cukup besar dengan titik tetangganya [8]. Pendeteksian tepi ini dapat mencirikan batas-batas objek untuk identifikasi suatu citra. Deteksi tepi merupakan salah satu contoh dari operasi bertetangga pada citra, yaitu operasi yang dipengaruhi oleh piksel tetangganya [3][8]. Tiap-tiap piksel akan memiliki nilai intensitas tertentu sehingga untuk memberikan hasil yang optimal citra awal harus di grayscale terlebih dahulu.

Prinsip dasar pada operasi bertetangga itu sendiri adalah konvolusi antara citra dengan suatu filter. Filter untuk proses pendeteksian tepi ini tergantung pada operator yang digunakan. Operator ini akan mengacu pada jenis metode deteksi yang akan dimanfaatkan. Secara umum terdapat dua metode yang dapat digunakan, yaitu metode gradien (turunan pertama) dan metode turunan kedua.

Pada makalah ini akan digunakan metode gradien, yaitu operator Sobel. Pada operator Sobel dilakukan pemberian nilai bobot lebih pada bagian tengah filter dengan asumsi bagian ini memberikan kontribusi lebih dibandingkan bagian lain. Filter yang digunakan pada operator Sobel ini terbagi dalam dua arah yaitu arah horizontal dan vertikal, sesuai dengan citra asal yang merupakan citra dua dimensi. Besarnya bobot filter pada operator Sobel ini dapat dilihat pada gambar di bawah ini :

Gx	Gy																		
<table border="1" style="border-collapse: collapse; width: 100%;"> <tr><td style="padding: 5px;">-1</td><td style="padding: 5px;">0</td><td style="padding: 5px;">1</td></tr> <tr><td style="padding: 5px;">-2</td><td style="padding: 5px;">0</td><td style="padding: 5px;">2</td></tr> <tr><td style="padding: 5px;">-1</td><td style="padding: 5px;">0</td><td style="padding: 5px;">1</td></tr> </table>	-1	0	1	-2	0	2	-1	0	1	<table border="1" style="border-collapse: collapse; width: 100%;"> <tr><td style="padding: 5px;">1</td><td style="padding: 5px;">2</td><td style="padding: 5px;">1</td></tr> <tr><td style="padding: 5px;">0</td><td style="padding: 5px;">0</td><td style="padding: 5px;">0</td></tr> <tr><td style="padding: 5px;">-1</td><td style="padding: 5px;">-2</td><td style="padding: 5px;">-1</td></tr> </table>	1	2	1	0	0	0	-1	-2	-1
-1	0	1																	
-2	0	2																	
-1	0	1																	
1	2	1																	
0	0	0																	
-1	-2	-1																	

**Gambar 1 Filter pada operator Sobel**

Dari filter ini penentuan tepi dari suatu objek dilakukan berdasarkan kekuatan tepi dari suatu citra. Penentuan kekuatan tepi diperoleh dari persamaan berikut [3][7] :

$$G[f(x, y)] = \sqrt{Gx^2 + Gy^2} \tag{1}$$

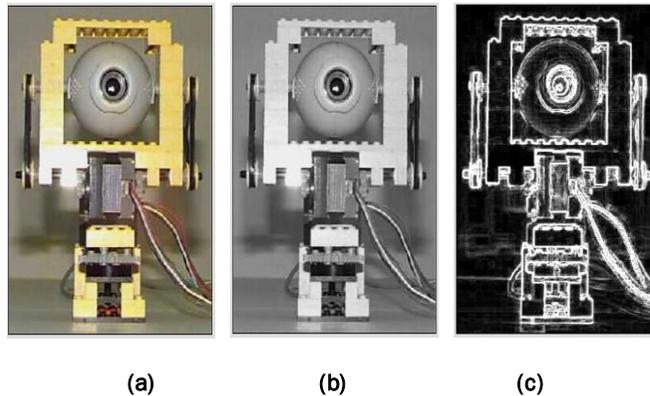
Dalam prakteknya persamaan ini biasanya disederhanakan menjadi :

$$G[f(x, y)] = |Gx| + |Gy| \tag{2}$$

Untuk penentuan suatu piksel merupakan tepi atau tidak dinyatakan dalam suatu operasi pengambangan (pemberian nilai treshold) seperti berikut :

$$g(x,y) = \begin{cases} 1, & \text{jika } G[f(x,y)] \geq T \\ 0, & \text{lainnya} \end{cases} \quad (3)$$

piksel tepi akan ditandai dengan warna putih dan bukan tepi ditandai dengan warna hitam. Hasil edge detection pada suatu objek dapat dilihat pada Gambar 2.



Gambar 2 (a) Citra asli, (b) Grayscale, (c) Deteksi tepi Sobel

### 3 Monte Carlo dan Random Walks

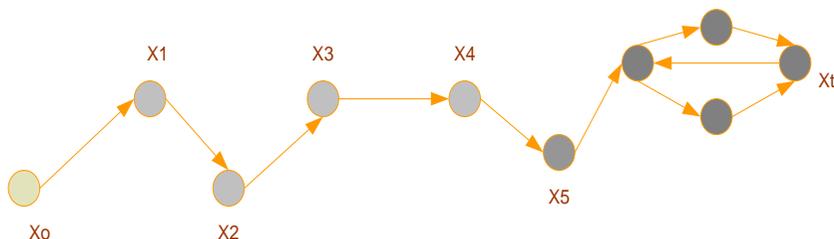
Metode Monte Carlo merupakan suatu metode yang sangat efektif untuk melakukan proses simulasi khususnya untuk sistem yang bersifat stokastik. Metode ini akan menirukan suatu proses secara acak atau random serta memperhitungkan probabilitas. Metode ini terutama digunakan untuk permasalahan yang rumit, yang tidak bisa lagi dipecahkan melalui perhitungan deterministik [9].

Dalam metode Monte Carlo ini dikenal suatu pemecahan yang disebut dengan random walks. Untuk metode Random Walks pada kasus satu dimensi (1D) dapat dituliskan seperti Markov Chain Monte Carlo, yaitu dengan suatu keadaan yang dapat digambarkan sebagai bilangan integer  $i = 0, \pm 1, \pm 2, \pm 3, \dots$  dengan  $0 < p < 1$ , maka

$$P_{i, i+1} = p = 1 - P_{i, i-1} \quad (4)$$

Metode ini disebut dengan Random Walk karena dapat dianalogikan dengan langkah manusia dalam satu garis / jalan lurus yang sewaktu-waktu dapat berjalan ke arah kanan dengan probabilitas  $p$  dan ke kiri dengan probabilitas  $1-p$  [8]. Random Walks ini merupakan salah satu contoh proses stokastik yang sederhana.

Ilustrasi gerakan secara random walks pada sistem yang bertujuan untuk mencapai titik tengah dapat dilihat pada Gambar 3.



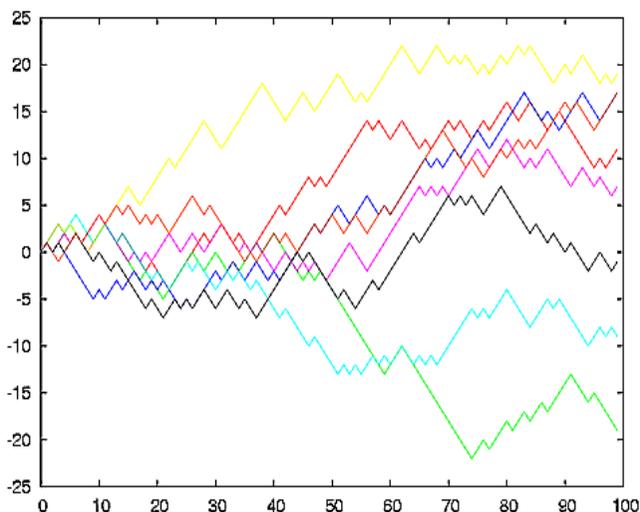
**Gambar 3 Gerakan yang dilakukan secara random walks**

Dari Gambar 3 terlihat adanya pergerakan secara random yang memungkinkan adanya pergerakan balik ke titik sebelumnya pada perpindahan ke  $X_t$ .

Dalam prosesnya Random Walks memiliki beberapa aturan yang harus ‘dipatuhi’ berkaitan dengan langkah yang akan ditmpuh selanjutnya yaitu :

1. Terdapat titik awal (starting point)
2. Penentuan titik berikutnya dilakukan dengan acak, dimana tidak ada satu peluang yang lebih dibandingkan dengan yang lain. Dalam artian setiap arah langkah memiliki peluang yang sama.
3. Besarnya tiap step adalah sama

Gambar 4 menunjukkan contoh gerak random walks dengan delapan random walks yang dimulai dari titik 0 [8] :

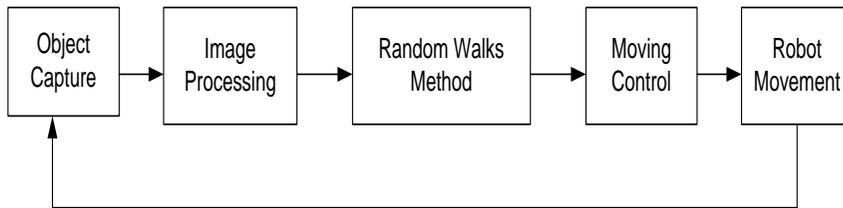


**Gambar 4 Contoh random walks**

#### 4 Model Random Walks untuk Kontrol Robot

Seperti yang telah disebutkan sebelumnya bahwa dalam robot haruslah terdapat suatu algoritma yang dimanfaatkan untuk mencapai tujuan yang diinginkan. Untuk pengontrolan gerak robot yang berbasiskan pada citra ini, robot akan diprogram berorientasikan objek sehingga setiap gerakannya akan memberikan feedback ke sistem.

Sistem secara keseluruhan untuk pengontrolan ini dapat dilihat dari diagram blok di Gambar 5 :



**Gambar 5 Diagram blok sistem keseluruhan**

Dari Gambar 5 dapat dijelaskan bahwa pada bagian awal akan dilakukan suatu pengambilan objek sebagai input dengan memanfaatkan kamera yang berfungsi sebagai sensor. Penentuan starting point untuk memulai random walks akan dilakukan dari titik tengah objek yang telah di capture. Titik awal ini diperoleh dengan suatu teknik edge detection pada image processing. Hasil ini akan digunakan untuk dipadukan dengan metode random walks yang melalui suatu controller akan menghasilkan pergerakan sampai didapatkan titik akhir yaitu posisi yang tepat berada di tengah citra.

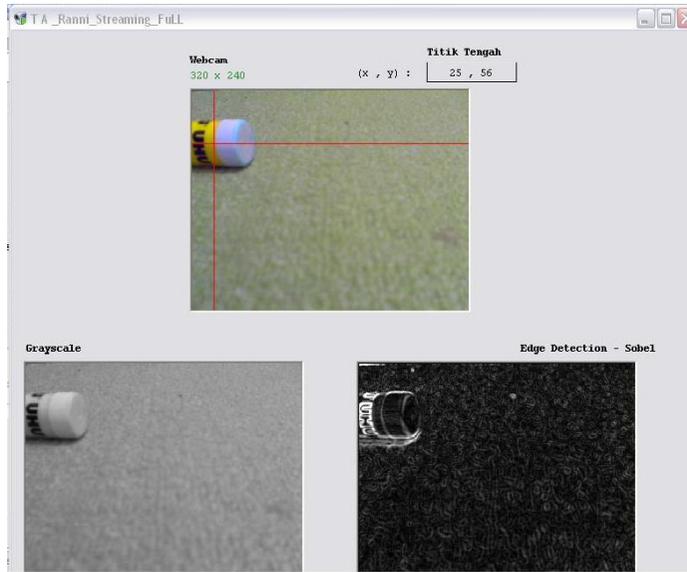
Untuk random walks pada sistem ini hanya menggunakan random walks tipe 1 dimensi, sehingga pergerakan yang dilihat nantinya adalah pada arah horizontal yaitu kiri dan kanan. Input yang berupa posisi awal dibandingkan dengan titik akhir yang akan dicapai.

Pengambilan keputusan gerakan ke arah kiri dan kanan didasarkan pada bilangan acak yang muncul. Dengan adanya integrasi dengan suatu controller nantinya akan terdapat delay untuk pengambilan posisi terbaru dan akan dibandingkan lagi dengan titik akhir yang ingin dicapai. Proses random walks akan terus berlangsung sampai pada suatu iterasi yang telah memberikan posisi yang sesuai dengan titik yang ditetapkan sebagai titik akhir

## 5 Hasil Pengujian Sistem

Simulasi Random Walks yang dilakukan adalah untuk tipe 1D yang bertujuan untuk menghentikan gerakan acak (random walks) pada posisi tengah citra. Pada simulasi ini diberikan suatu nilai input yang menyatakan lebar piksel dan masukan suatu nilai ambang tertentu. Nilai ambang disini berfungsi sebagai batasan dari posisi akhir yang menyatakan bahwa posisi telah berada di tengah piksel.

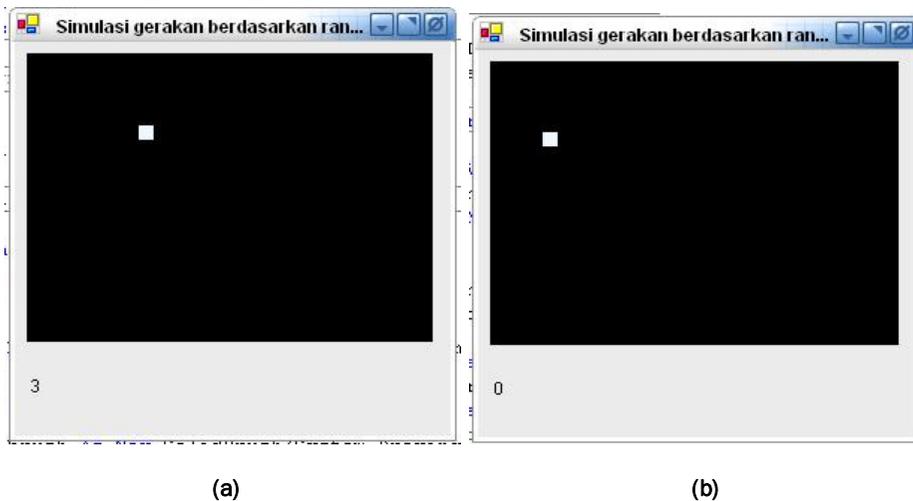
Penentuan titik awal didapatkan dari titik tengah yang diperoleh dari hasil pendeteksian tepi. Hasil perolehan titik ini dapat dilihat dari Gambar 6.



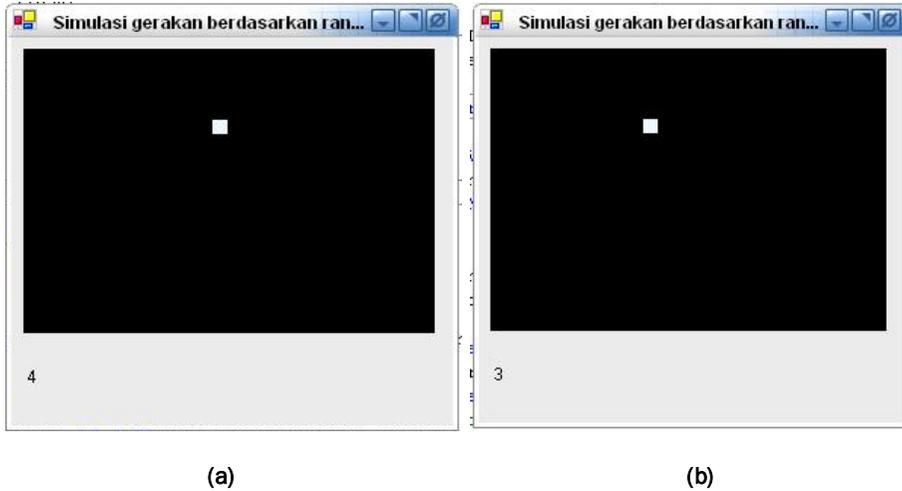
Gambar 6 Penentuan starting point melalui *edge detection*

Berdasarkan Gambar 6 di atas didapat starting point untuk melakukan random walks pada  $x = 25$ . Untuk random walks yang terjadi dilakukan suatu simulasi sesuai dengan bilangan random yang muncul.

Dari simulasi yang dilakukan, dapat dilihat adanya pengaruh bilangan random yang muncul dengan gerakan yang dilakukan. Gambar 7 memperlihatkan simulasi gerak robot (diwakili dengan kotak putih) sesuai dengan bilangan random yang muncul. Gerak yang terjadi dimulai dari  $x = 25$  (Gambar 7 (a)), dan berhenti pada  $x = 157$  (Gambar 8 (b)).



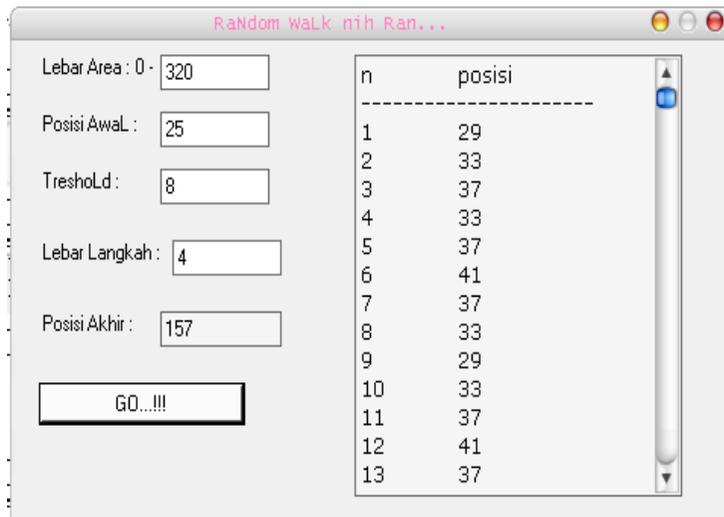
Gambar 7 (a) Titik awal ( $x = 25$ ) (b) Posisi  $x = 53$ , dengan bilangan random = 3



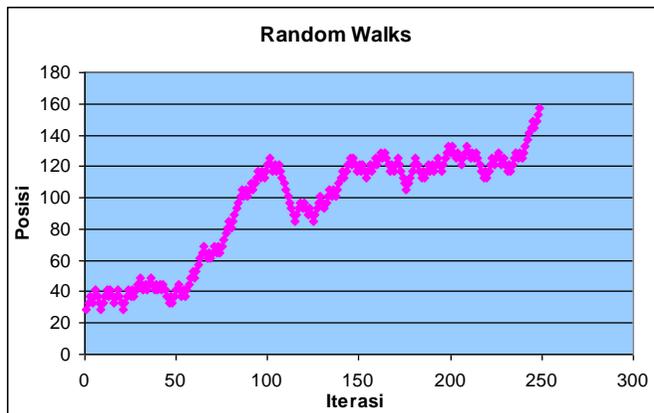
Gambar 8 (a) Posisi  $x=129$ , bilangan random = 3 (b) Posisi  $x= 157$ , bilangan random = 4

Untuk dapat mengamati setiap random walks yang terjadi dapat dilakukan simulasi yang akan memberikan plot dari setiap pergerakannya. Berikut ini simulasi yang dilakukan dengan nilai input pada titik  $x = 25$ , dan posisi akhir berada pada  $x=157$  dengan memberikan ukuran citra  $320 \times 240$ .

Berdasarkan perhitungan sesungguhnya titik tengah citra seharusnya terletak pada posisi  $x = 160$ , akan tetapi pada simulasi ini terdapat suatu nilai ambang atau treshold yang memberikan rentang posisi yang dianggap sebagai titik tengah dari citra. Proses simulasi dan plot gerakan random walks yang terjadi dapat dilihat pada gambar 9 dan 10.



Gambar 9 Simulasi random walks



**Gambar 10** Grafik simulasi random walks (251 iterasi)

Dari grafik di atas dapat dilihat adanya proses random walks yang terjadi. Simulasi ini memperlihatkan titik akhir yang dicapai adalah posisi  $x = 157$  yang dianggap sebagai titik tengah dari citra. Pada simulasi ini telah berhasil ditentukan posisi tengah dari citra yang ditandai dengan berhentinya proses random walks, yang juga menandakan berhentinya pergerakan yang dilakukan robot.

## 6 Kesimpulan

Untuk pengontrolan gerakan robot yang berbasis citra dapat memanfaatkan metode Monte Carlo dengan algoritma random walks. Metode ini akan memanfaatkan input dari titik tengah objek yang tercapture, dan estimasi terhadap pergerakan robot ke step selanjutnya dilakukan melalui bilangan random (acak).

Grafik yang diperoleh dari simulasi ini dapat dimanfaatkan untuk mengetahui besarnya error dari pergerakan robot dalam penentuan titik tengah dari citra. Dengan memanfaatkan metode ini dapat dilakukan suatu pengontrolan gerakan robot untuk mencapai titik tertentu dengan lebih efisien karena telah dilengkapi dengan suatu rule yang harus dipatuhi.

## 7 Daftar Pustaka

- [1] Lego, Education. Robotic System, "A Guide to Understanding the Robot Around Us". The Lego Group.1998.
- [2] J.J.Craig. 1989. *Introduction to Robotics*. Addyson-Wesley. New York
- [3] Jain, Anil K.1989.*Fundamental of Digital Image Processing*.Prentice-Hall International Inc.New Jersey.
- [4] Low,Adrian.1991.*Introductory Computer Vision and Image Processing*.McGraw-Hill.
- [5] G. Kantor and S. Singh. Preliminary results in range only localization and mapping. In*Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*,pages 1819.1825, 2002.
- [6] S. Singh, G. Kantor, and D. Strelow. Recent results in extensions to simultaneous localization and mapping. In *Proceedings of the International Symposium of Experimental Robotics (ISER)*, 2002.
- [7] Gonzales, Refael C.1977..*Digital Image Processing*.Addison-Wesley Publishing
- [8] <http://en.wikipedia.org/wiki/Physics>

[9] <http://suzy.unl.edu/bruno/che484/montecarlo.html>