



Teknologi Robotika: Motion Planning

Teori, Algoritma dan Simulasi

Hendawan Soebhakti

TEKNOLOGI ROBOTIKA: MOTION PLANNING

Teori, Algoritma dan Simulasi

UU No. 28 Tahun 2014 tentang Hak Cipta

Fungsi dan Sifat Hak Cipta Pasal 4

Hak Cipta sebagaimana dimaksud dalam Pasal 3 huruf a merupakan hak eksklusif yang terdiri atas hak moral dan hak ekonomi.

Pembatasan Pelindungan Pasal 26

Ketentuan sebagaimana dimaksud dalam Pasal 23, Pasal 24, dan Pasal 25 tidak berlaku terhadap:

- i. Penggunaan kutipan singkat Ciptaan dan/atau produk Hak Terkait untuk pelaporan peristiwa aktual yang ditujukan hanya untuk keperluan penyediaan informasi aktual;
- ii. Penggandaan Ciptaan dan/atau produk Hak Terkait hanya untuk kepentingan penelitian ilmu pengetahuan;
- iii. Penggandaan Ciptaan dan/atau produk Hak Terkait hanya untuk keperluan pengajaran, kecuali pertunjukan dan Fonogram yang telah dilakukan Pengumuman sebagai bahan ajar; dan
- iv. Penggunaan untuk kepentingan pendidikan dan pengembangan ilmu pengetahuan yang memungkinkan suatu Ciptaan dan/atau produk Hak Terkait dapat digunakan tanpa izin Pelaku Pertunjukan, Produser Fonogram, atau Lembaga Penyiaran.

Sanksi Pelanggaran Pasal 113

1. Setiap Orang yang dengan tanpa hak melakukan pelanggaran hak ekonomi sebagaimana dimaksud dalam Pasal 9 ayat (1) huruf i untuk Penggunaan Secara Komersial dipidana dengan pidana penjara paling lama 1 (satu) tahun dan/atau pidana denda paling banyak Rp100.000.000 (seratus juta rupiah).
2. Setiap Orang yang dengan tanpa hak dan/atau tanpa izin Pencipta atau pemegang Hak Cipta melakukan pelanggaran hak ekonomi Pencipta sebagaimana dimaksud dalam Pasal 9 ayat (1) huruf c, huruf d, huruf f, dan/atau huruf h untuk Penggunaan Secara Komersial dipidana dengan pidana penjara paling lama 3 (tiga) tahun dan/atau pidana denda paling banyak Rp500.000.000,00 (lima ratus juta rupiah).

TEKNOLOGI ROBOTIKA: MOTION PLANNING

Teori, Algoritma dan Simulasi

Hendawan Soebhakti



**TEKNOLOGI ROBOTIKA: MOTION PLANNING
TEORI, ALGORITMA DAN SIMULASI**

Penulis : Hendawan Soebhakti
Desain Cover : Rulie Gunadi
Sumber : (Olesia Bech) www.shutterstock.com
Tata Letak : Zulita A.
Proofreader : Mira M.

Ukuran:
xii, 114 hlm., Uk.: 15.5x23 cm

ISBN:
978-623-02-9712-0

ISBN Elektronis :
978-634-200-199-8 (PDF)

Tahun Terbit Digital :
2024

Hak Cipta 2024 pada Penulis
Copyright © 2024 by Deepublish Digital
All Right Reserved

PENERBIT DEEPUBLISH DIGITAL
(Grup Penerbitan CV BUDI UTAMA)

Anggota IKAPI (076/DIY/2012)
Jl. Rajawali, Gg. Elang 6, No. 3, Drono, Sardonoharjo, Ngaglik, Sleman
Jl. Kaliurang Km. 9,3 – Yogyakarta 55581
Telp./Faks : (0274) 4533427
Website : www.penerbitdeepublish.com
www.deepublishstore.com
E-mail : cs@deepublish.co.id

Hak cipta dilindungi undang-undang.

*Dilarang keras menerjemahkan, memfotokopi, atau
memperbanyak sebagian atau seluruh isi buku ini
tanpa izin tertulis dari penerbit.*

Isi di luar tanggung jawab percetakan.

DAFTAR ISI

PRAKATA	v
KATA PENGANTAR PENERBIT	vi
DAFTAR ISI	vii
DAFTAR GAMBAR	x
1 PENGANTAR	1
1.1. Apa Itu Perencanaan Gerak?	1
1.2. <i>Continuous Space</i> dan <i>Discrete Space</i>	4
1.3. Strategi Perencanaan Gerak	6
1.4. Faktor Penting Memilih Strategi Perencanaan Gerak	9
1.5. Ringkasan	11
1.6. Pertanyaan Diskusi	11
1.7. Refleksi Pembelajaran	12
2 INSTALASI SOFTWARE	13
2.1. Instalasi Visual Studio Code	13
2.2. Instalasi Python	14
2.3. Instalasi NumPy dan SciPy	17
2.4. Instalasi Matplotlib Library	17
2.5. Program Simulasi Sederhana	17
2.6. Ringkasan	19
2.7. Pertanyaan Diskusi	19
3 ALGORITMA A*	21
3.1. Prinsip Kerja Algoritma A*	21
3.2. Contoh Pencarian Jalur Terpendek	24
3.3. Simulasi Algoritma A*	27
3.4. Kelebihan dan Kekurangan	35
3.5. Ringkasan	35
3.6. Pertanyaan Diskusi	35
3.7. Refleksi Pembelajaran	36

4	ALGORITMA DIJKSTRA'S	37
4.1.	Prinsip Kerja Algoritma Dijkstra's	37
4.2.	Simulasi Algoritma Dijkstra's	41
4.3.	Ringkasan	49
4.4.	Pertanyaan Diskusi	49
4.5.	Refleksi Pembelajaran	50
5	ALGORITMA D*	51
5.1.	Prinsip Kerja Algoritma D*	51
5.2.	Ekspansi	52
5.3.	Halangan atau <i>Obstacle</i>	52
5.4.	Simulasi Algoritma D*	53
5.5.	Ringkasan	62
5.6.	Pertanyaan Diskusi	62
5.7.	Refleksi Pembelajaran	63
6	<i>PROBABILISTIC ROAD-MAP (PRM)</i>	64
6.1.	Prinsip Kerja <i>Probabilistic Road-Map (PRM)</i>	64
6.2.	Simulasi <i>Probabilistic Road-Map (PRM)</i>	66
6.3.	Kelebihan dan Kelemahan Algoritma PRM	75
6.4.	Ringkasan	76
6.5.	Pertanyaan Diskusi	77
6.6.	Refleksi Pembelajaran	78
7	<i>RAPIDLY-EXPLORING RANDOM TREES (RRT)</i>	79
7.1.	Prinsip Kerja <i>Rapidly-Exploring Random Trees (RRT)</i>	79
7.2.	Simulasi <i>Rapidly-Exploring Random Trees (RRT*)</i>	80
7.3.	Kelebihan dan Kelemahan Algoritma RRT	88
7.4.	Ringkasan	88
7.5.	Pertanyaan Diskusi	88
7.6.	Refleksi Pembelajaran	89
8	<i>PARTICLE SWARM OPTIMATION (PSO)</i>	90
8.1.	Prinsip Kerja Particle Swarm Optimation (PSO)	90
8.2.	Simulasi <i>Particle Swarm Optimation (PSO)</i>	91

8.3.	Ringkasan	98
8.4.	Pertanyaan Diskusi.....	99
8.5.	Refleksi Pembelajaran.....	99
9	ALGORITMA GENETIKA.....	100
9.1.	Prinsip Kerja Algoritma Genetika	100
9.2.	Contoh Implementasi Algoritma Genetika	103
9.3.	Optimasi Perencanaan Gerak dengan Algoritma Genetika	107
9.4.	Pertanyaan Diskusi.....	109
9.5.	Refleksi Pembelajaran.....	109
10	PENUTUP.....	110
	DAFTAR PUSTAKA	111
	INDEKS	113
	TENTANG PENULIS.....	114

PENGANTAR

Tujuan

Setelah mempelajari materi ini maka pembaca diharapkan mampu:

- Menjelaskan apa itu perencanaan gerak.
- Menjelaskan arsitektur sistem robot otonom.
- Menjelaskan perbedaan *continuous space* dengan *discrete space*.
- Menjelaskan *node* dan *edge* dari sebuah grafik representasi perencanaan gerak.
- Memilih metode *path planning* sesuai kebutuhan.

1.1. Apa Itu Perencanaan Gerak?

Tantangan mendasar dalam membangun sistem navigasi robot atau kendaraan otonom adalah lokalisasi, membangun peta lingkungan, dan perencanaan jalur atau *path planning* (Lixing Liu 2023). Lokalisasi diperlukan untuk mengetahui posisi dan orientasi robot di sebuah lingkungan. Robot dapat menggunakan kombinasi sejumlah sensor, seperti sensor jarak, kamera, *Light Detection and Ranging* (LIDAR), *Global Positioning System* (GPS), *Inertial Measurement Unit* (IMU), *odometry*, dan lainnya. Data sensor diolah oleh kontroler untuk mendapatkan persepsi posisi dan orientasi sehingga dapat dibentuk peta atau pemodelan lingkungan sekitar. Salah satu metode yang dapat digunakan untuk membuat model lingkungan adalah *Simultaneous Localization and Mapping* (SLAM). Robot secara simultan akan membuat peta 2 atau 3 dimensi berdasarkan data sensor sekaligus memperbaharui posisinya di dalam peta tersebut.

INSTALASI *SOFTWARE*

Tujuan

Setelah mempelajari materi ini maka pembaca diharapkan mampu:

- Melakukan instalasi code editor, Python, NumPy, SciPy dan library Matplotlib yang diperlukan untuk membuat simulasi perencanaan gerak dan optimasi.
- Membuat simulasi program sederhana.

Pada buku ini, algoritma *path planning* akan diimplementasikan dalam bentuk kode program yang dapat disimulasikan hasilnya secara visual. Kode program dibuat dengan bahasa pemrograman Python yang akan ditulis pada *code editor*, yaitu Visual Studio Code. Agar hasil simulasi dapat divisualisasikan, maka perlu ditambahkan beberapa *library*, antara lain NumPy, SciPy dan Matplotlib. Berikut adalah *link download* untuk *software* yang digunakan:

- Visual Studio Code : <https://code.visualstudio.com/>
- Python : <https://www.python.org/>
- NumPy : <https://numpy.org/>
- SciPy : <https://scipy.org/>
- Matplotlib : <https://matplotlib.org/>

2.1. Instalasi Visual Studio Code

Installer Visual Studio dapat diunduh dengan menekan tombol **Download for Windows** seperti terlihat pada Gambar 2-1. Jika *file*

3

ALGORITMA A*

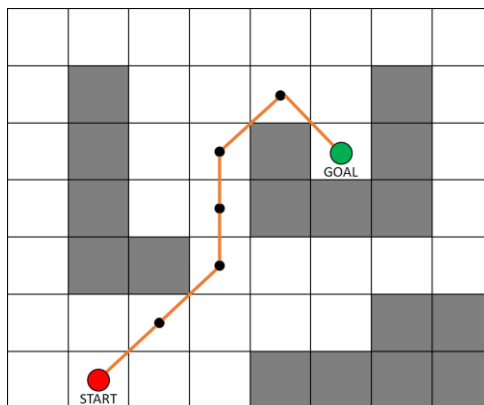
Tujuan

Setelah mempelajari materi ini maka pembaca diharapkan mampu:

- Menjelaskan algoritma A*.
- Membuat program simulasi perencanaan gerak menggunakan algoritma A*.

3.1. Prinsip Kerja Algoritma A*

Algoritma A Star atau A* dapat digunakan untuk mencari rute terpendek. Jika sebuah robot dari posisi START (digambarkan dengan bulatan merah), hendak menuju ke posisi GOAL (digambarkan dengan bulatan hijau), maka perlu direncanakan pergerakannya agar tidak menabrak halangan.



Gambar 3-1 Ilustrasi Pencarian Rute Terpendek

ALGORITMA DIJKSTRA'S

Tujuan

Setelah mempelajari materi ini maka pembaca diharapkan mampu:

- Menjelaskan algoritma Dijkstra's.
- Membuat program simulasi perencanaan gerak menggunakan algoritma Dijkstra's.
- Mengidentifikasi perbedaan algoritma Dijkstra's dengan algoritma A*.

4.1. Prinsip Kerja Algoritma Dijkstra's

Algoritma Dijkstra's digunakan untuk menentukan lintasan terpendek dari suatu simpul (*node*) ke semua simpul lainnya. Algoritma Dijkstra's diperkenalkan oleh Dr. Edsger W. Dijkstra pada tahun 1959. Untuk memahami algoritma Dijkstra's lebih detail, maka dapat dibuat ilustrasi seperti pada Gambar 4-1. Seseorang yang berasal dari Kota A hendak menuju Kota F. Maka perlu dipilih jalur terpendeknya. Beberapa hal yang perlu diperhatikan dari Gambar 4-1 adalah:

- Setiap kota digambarkan sebagai sebuah *node*.
- Setiap jalan digambarkan sebagai sebuah *edge*.
- Setiap jalan memiliki nilai. Nilai ini bisa berupa jarak antar kota, biaya tol, atau tingkat kemacetan lalu lintas. Pada contoh ini, nilai pada jalan ini adalah jarak antar kota.

Berdasarkan Gambar 4-1 maka terdapat beberapa rute yang bisa ditempuh dari Kota A menuju Kota F yaitu:

- A-C-E-F

ALGORITMA D*

Tujuan

Setelah mempelajari materi ini maka pembaca diharapkan mampu:

- Menjelaskan cara kerja algoritma D*.
- Membuat program simulasi perencanaan gerak menggunakan algoritma D* /.
- Mengidentifikasi perbedaan algoritma D* dengan Dijkstra dan A*.

5.1. Prinsip Kerja Algoritma D*

Algoritma D* diperkenalkan oleh Anthony Stentz pada tahun 1994. Nama D* diambil dari kata Dynamic A*, karena merupakan pengembangan algoritma A*. Pada algoritma D*, nilai *cost* dapat berubah saat algoritma dijalankan. Untuk implementasi yang lebih praktis, maka dapat digunakan varian lain dari D*, seperti Focused D* dan D* Lite. Struktur data D* adalah sebagai berikut:

- NEW, belum pernah masuk OPEN *list*.
- OPEN, saat ini ada di dalam OPEN *list*.
- CLOSED, sudah tidak ada lagi di OPEN *list*.
- RAISE, menunjukkan biayanya lebih tinggi daripada terakhir kali ada di OPEN *list*.
- LOWER, menunjukkan biayanya lebih rendah daripada terakhir di OPEN *list*.

PROBABILISTIC ROAD-MAP (PRM)

Tujuan

Setelah mempelajari materi ini maka pembaca diharapkan mampu:

- Menjelaskan cara kerja algoritma *Probabilistic Road-Map* (PRM).
- Membuat program simulasi perencanaan gerak menggunakan algoritma PRM.
- Mengidentifikasi perbedaan algoritma PRM dengan D*.

6.1. Prinsip Kerja *Probabilistic Road-Map* (PRM)

Probabilistic Road-Map (PRM) adalah salah satu algoritma perencanaan gerak yang digunakan untuk mencari jalur terpendek antara titik A dan titik B yang tidak berbenturan dengan halangan. Pada algoritma PRM, sebuah peta jalan akan dibuat secara acak, di mana terdiri dari sejumlah konfigurasi *node* (titik) dan *edge* (garis penghubung) yang akan menghubungkan antara titik awal dan titik tujuan. Setiap *node* akan dihubungkan oleh *edge* untuk dilihat apakah terdapat halangan atau tidak. *Node* dan *edge* yang tidak terhalang (*collision-free*) akan dipilih sebagai jalur terpendek menuju ke titik tujuan. Pada ruang 3 dimensi, sebuah titik atau *node* dapat menempati koordinat x, y, dan z atau disebut juga dengan *configuration space* atau *C-space*. Terdapat dua fase dalam algoritma PRM, yaitu:

1. Mempelajari peta atau map.
 - Langkah ke-1 adalah menghasilkan *node* secara acak dan menghubungkan antar *node* dengan *edge*. Langkah ini dilakukan sebanyak n kali iterasi.

RAPIDLY-EXPLORING RANDOM TREES (RRT)

Tujuan

Setelah mempelajari materi ini maka pembaca diharapkan mampu:

- Menjelaskan cara kerja algoritma Rapidly-Exploring Random Trees (RRT).
- Membuat program simulasi perencanaan gerak menggunakan algoritma RRT.
- Mengidentifikasi perbedaan algoritma RRT dengan PRM.

7.1. Prinsip Kerja *Rapidly-Exploring Random Trees* (RRT)

RRT adalah algoritma yang dapat menghasilkan perencanaan gerak dengan menggunakan teknik pengambilan *sample* secara acak. Seperti terlihat pada Gambar 7-1, cara kerjanya seperti layaknya ranting atau cabang pohon yang tumbuh dengan tahapan sebagai berikut:

- q_{init} merupakan titik awal dari lintasan yang ingin direncanakan. Pada awalnya, RRT memulai proses pembangunan pohon dari titik ini.
- Ambil satu titik atau *node* secara acak pada peta (q_{rand}).
- Coba hubungkan dengan *node* terdekat (q_{near}) yaitu tetangga terdekat dari pohon yang sudah ada.
- Jika tidak ada halangan dan semua parameter terpenuhi, maka sambungkan *node* tersebut menjadi bagian dari cabang pohon (q_{new})

PARTICLE SWARM OPTIMATION **(PSO)**

Tujuan

Setelah mempelajari materi ini maka pembaca diharapkan mampu:

- Menjelaskan cara optimasi perencanaan gerak dengan algoritma *Particle Swarm Optimization* (PSO).
- Membuat program simulasi optimasi perencanaan gerak menggunakan algoritma PSO.
- Menganalisis hasil kinerja optimasi algoritma PSO.

8.1. Prinsip Kerja Particle Swarm Optimization (PSO)

Particle Swarm Optimization (PSO) adalah algoritma optimasi berbasis populasi yang terinspirasi dari perilaku terbang kawanan burung atau kawanan ikan. Dalam PSO, sekelompok partikel bergerak mencari solusi paling optimal dengan memperbarui posisi dan *velocity*-nya berdasarkan pengalamannya sendiri dan pengalaman tetangganya. PSO menjadi pilihan populer untuk *path planning* karena kemampuannya menangani permasalahan optimasi nonlinier dan non-convex (Gad 2022).

Pada PSO terdapat fungsi penalti, yaitu memberikan nilai penalti kepada fungsi objektif pada saat melanggar batasan-batasan tertentu. Dalam konteks *path planning*, fungsi penalti digunakan untuk memastikan jalur yang dibentuk memenuhi kriteria atau batasan-batasan yang sudah ditentukan. Misalnya:

ALGORITMA GENETIKA

Tujuan

Setelah mempelajari materi ini maka pembaca diharapkan mampu:

- Menjelaskan prinsip kerja algoritma genetika.
- Menjelaskan optimasi perencanaan gerak dengan algoritma genetika.
- Membuat program simulasi optimasi perencanaan gerak menggunakan algoritma genetika.
- Menganalisis hasil kinerja optimasi algoritma genetika.

9.1. Prinsip Kerja Algoritma Genetika

Algoritma genetika atau *Genetic Algorithm* (GA) pertama kali diperkenalkan oleh John Holland pada tahun 1975 di University of Michigan, AS. Algoritma ini dibangun berdasarkan teori seleksi alam dan mutasi genetika yang menghasilkan perubahan fisiologis terhadap suatu makhluk hidup. Algoritma genetika dikembangkan untuk memecahkan suatu permasalahan yang memiliki banyak pilihan solusi dengan cara mengevaluasi semua alternatif solusi berdasarkan kriteria yang telah ditentukan atau *fitness function* hingga diperoleh solusi yang paling optimal (Katoch 2021).

Gambar 9-1 memperlihatkan *flowchart* algoritma genetika yang diawali dengan membentuk sebuah **populasi** yang terdiri dari individu-individu. Setiap individu merepresentasikan solusi-solusi dari permasalahan yang akan diselesaikan. Sebuah individu memiliki parameter atau variabel yang disebut **Gen**. Sejumlah gen akan

PENUTUP

Algoritma perencanaan gerak untuk robot memungkinkan robot dapat bergerak secara otonom melalui lingkungan operasinya. Ada berbagai algoritma yang tersedia, masing-masing dengan kelebihan dan kekurangannya. Saat memilih algoritma, penting untuk mempertimbangkan faktor-faktor seperti kompleksitas komputasional, optimalitas, pengetahuan tentang lingkungan, dimensi ruang, dan hambatan dinamis.

Python adalah bahasa pemrograman populer untuk mengimplementasikan algoritma perencanaan gerak karena kesederhanaan, kemudahan dibaca, dan dukungan pustaka yang luas. Buku ini membahas beberapa algoritma umum, seperti algoritma A*, algoritma Dijkstra, dan algoritma RRT, dan menyediakan potongan kode dalam bahasa pemrograman Python untuk mengimplementasikannya.

Dengan memahami dan mengimplementasikan algoritma-algoritma ini, kita dapat mengembangkan robot yang dapat bergerak dengan aman dan efisien di dalam lingkungan operasinya, mengoptimalkan gerakannya, meminimalkan waktu perjalanan, konsumsi energi, dan aus pada robot.

DAFTAR PUSTAKA

- al., J. Levinson et. 2011. "Towards fully autonomous driving: Systems and algorithms." *IEEE Intelligent Vehicles Symposium (IV)* 163–168.
- al., Y. Yang et. 2022. "An optimal goal point determination algorithm for automatic navigation of agricultural machinery: Improving the tracking accuracy of the Pure Pursuit algorithm." *Comput. Electron. Agric.*, vol. 194 106760.
- Amanda Baughan, Vinitha Ranganeni, Nathan Hatch, and Boling Yang. 2017. "Search-Based Test Generation for Robotic Motion Planning." *ACM Conference (Conference'17)*. New York: Association for Computing Machinery. 1-6.
- B. Paden, M. Čáp, S. Z. Yong, D. Yershov, and E. Frazzoli. 2016. "A survey of motion planning and control techniques for self-driving urban vehicles." *IEEE Trans. Intell. Veh.*, vol. 1, no. 1 33–55.
- Gad, A.G. 2022. "Particle Swarm Optimization Algorithm and Its Applications: A Systematic Review." *Arch Computat Methods Eng* 29 2531–2561.
- Guangrong Chen, Ningze Wei, Lei Yan, HuaFeng Lu, and Jin Li. 2023. "Time-optimal trajectory planning based on." *Plos One* 1-34.
- Johnson, J. G. Mooney and E. N. 2014. "A Comparison of Automatic Nap-of-the-earth Guidance Strategies for Helicopters." *J. F. Robot.*, vol. 23 1–17.
- Katoch, S., Chauhan, S.S. & Kumar, V. 2021. "A review on genetic algorithm: past, present, and future." *Multimed Tools Appl* 80 8091–8126.
- LaValle, Steven M. 2006. *Planning Algorithm*. Cambridge University Press.
- Lixing Liu, Xu Wang, Xin Yang, Hongjie Liu, Jianping Li, Pengfei Wang. 2023. "Path planning techniques for mobile robots: Review and prospect." *Expert Systems With Applications* 1-30.

- Mathworks. 2024. *Path Planning Algorithms for Navigation*. March 2023. <https://uk.mathworks.com/help/nav/ug/choose-path-planning-algorithms-for-navigation.html>.
- Sebastian Thrun, Mike Montemerlo, Hendrik Dahlkamp, David Stavens, Andrei Aron, James Diebel, Philip Fong, John Gale, Morgan Halpenny, Gabriel Hoffmann, Kenny Lau, Celia Oakley, Mark Palatucci, Vaughan Pratt, and Pascal Stang. 2006. "Stanley: The Robot that Won." *Journal of Field Robotics* 662-692.
- Tihanyi, Á. Domina and V. 2023. "Model Predictive Controller Approach for Automated Vehicle's Path Tracking." *Sensors*, vol. 23, no. 15.
- Yang Yang, Yankai Li, Xing Wen, Gang Zhang, Qianglong Ma, Shangkun Cheng, Jian Qi, Liangyuan Xu, and Liqing Chen. 2022. "An optimal goal point determination algorithm for automatic navigation of agricultural machinery: Improving the tracking accuracy of the Pure Pursuit algorithm." *Computers and Electronics in Agriculture* Vol. 194, No. C, March 01.
- Zhang, Lei, and Yuan Li. 2021. "Mobile robot path planning algorithm based on improved a star." *ournal of Physics: Conference Series*. Vol. 1848. No. 1. IOP Publishing.
- Zhuozhen Tang, Hongzhong Ma. 2021. "An overview of path planning algorithms." *6th International Conference on Energy Science and Applied Technology*. IOP Publishing. 1-10.

INDEKS

A*, v, 4, 6, 21, 22, 23, 27, 34, 35, 37, 50, 51, 62, 63, 66, 77, 108
 Algoritma, v, 2, 4, 6, 7, 8, 10, 11, 12, 13, 19, 20, 21, 22, 23, 27, 34, 35, 36, 37, 38, 39, 40, 41, 42, 49, 50, 51, 52, 53, 54, 55, 56, 61, 62, 63, 64, 65, 66, 75, 76, 77, 78, 79, 80, 87, 88, 89, 90, 92, 98, 99, 100, 101, 103, 107, 108
Artificial Intelligence-Based Algorithms, 7
 Cost, 22, 24, 95
 D*, v, 7, 51, 52, 53, 54, 55, 56, 61, 62, 63, 64, 77, 88, 99, 107
 Edge, 1, 5, 12, 37, 41, 42, 64, 65, 66, 67, 70, 76
 Estimasi jalur, 22
Graph-Based Algorithm, 7
Grid-Based Algorithm, 6
 Heuristic, 26
 Matplotlib, 13, 17
 Node, 1, 5, 12, 22, 24, 27, 28, 29, 30, 31, 32, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 49, 52, 53, 64, 65, 66, 68, 71, 72, 75, 76, 79, 80, 81, 82, 83, 84, 85, 86, 88
 Obstacle, 50, 52, 62, 77, 92, 93
Optimization-Based Algorithms, 7
 Particle Swarm Optimization (PSO), 90, 91
 Path planning, 7, 91
 Perencanaan Gerak, 1, 5, 6, 9
 Probabilistic Road-Map (PRM), v, 64, 66
 Pseudocode, 27, 42, 53, 54, 80
 Python, 13, 14, 15, 16, 108
 Rapidly-Exploring Random Trees (RRT), 7, 79
Sampling-Based Algorithm, 7

TENTANG PENULIS



Hendawan Soebhakti, dosen bidang keahlian Teknologi Rekayasa Robotika. Mendapatkan gelar S.T. (2003) bidang Teknik Elektronika dari Institut Teknologi Nasional Malang dan M.T. (2010) dari Institut Teknologi Sepuluh Nopember Surabaya. Bergabung di Politeknik Negeri Batam (Polibatam) sejak tahun 2004 dan aktif dalam mengembangkan teknologi robotika, termasuk dalam membina tim kontes robot Polibatam. Saat ini menjadi salah satu peneliti di *Centre of Excellence of Robotics and Artificial Intelligence* Polibatam dengan fokus pada pengembangan produk *Automated Guided Vehicle*, *Security Patrol Robot* dan *Didactic for Robotics Technology*.

Teknologi Robotika: Motion Planning

Teori, Algoritma dan Simulasi

Sebuah robot atau kendaraan otonom harus memiliki kemampuan dasar untuk dapat merencanakan pergerakan dari posisi saat ini menuju ke posisi tujuan dengan memperhitungkan kondisi lingkungan sekitarnya. Rute pergerakan harus direncanakan agar efisien dengan jarak yang terdekat dan biaya termurah. Terdapat sejumlah algoritma yang dapat digunakan untuk membuat perencanaan gerak atau *motion planning*, antara lain A*, Dijkstra, D*, *Probabilistic Roadmap* (PRM), *Rapidly-Exploring Random Trees* (RRT), *Particle Swarm Optimization* (PSO) dan *Genetic Algorithm* (GA). Buku ini membahas secara detail cara kerja algoritma-algoritma tersebut dan diimplementasikan ke dalam program simulasi menggunakan pemrograman Python. Dengan mempelajari berbagai algoritma *motion planning* yang ada di buku ini, maka diharapkan pembaca dapat mengetahui kelebihan dan kelemahan masing-masing algoritma sehingga dapat memilih algoritma yang tepat sesuai kebutuhan.

Penerbit Deepublish (CV BUDI UTAMA)

Jl. Kaliurang Km 9,3 Yogyakarta 55581

Telp/Fax : (0274) 4533427

Anggota IKAPI (076/DIY/2012)

✉ cs@deepublish.co.id

📠 Penerbit Deepublish

📱 @penerbitbuku_deepublish

🌐 www.penerbitdeepublish.com



Kategori : Teknik Kontrol Otomatis

ISBN 978-634-200-199-8 (PDF)



9 786342 001998