

Implementasi Algoritma A-Star untuk Perencanaan Jalur Robot Menggunakan Deteksi Warna Berbasis Kamera

Akhmad Jayadi^{1*}, Panji Andika Pratomo¹, Ernando Rizki Dalimunthe²

¹ Jurusan Teknologi Informasi, Program Studi Manajemen Informatika, Politeknik Negeri Lampung, Bandar Lampung, Indonesia

² Fakultas Teknik dan Ilmu Komputer, Program Studi Teknik Elektro, Universitas Teknokrat Indonesia, Bandar Lampung, Indonesia

INFORMASI ARTIKEL

Diterima Redaksi: 23 Juni 2025
Revisi Akhir: 30 Juni 2025
Diterbitkan Online: 23 Juli 2025

KATA KUNCI

V-Rep
A*
Citra
Mobile Robot
Algoritma

KORESPONDENSI (*)

E-mail: akhmad.jayadi@polinela.ac.id

A B S T R A K

Penelitian ini bertujuan untuk mengembangkan sistem perencanaan jalur (path planning) robot mobile dengan menggunakan algoritma A* berbasis pengolahan citra dari kamera top-view di lingkungan simulasi V-REP (CoppeliaSim). Metode yang digunakan melibatkan deteksi posisi robot, tujuan, dan rintangan melalui segmentasi warna pada citra RGB, kemudian mengubah hasil deteksi menjadi representasi grid navigasi biner. Jalur optimal dari posisi awal ke tujuan dihitung menggunakan algoritma A*, dan hasil lintasan divisualisasikan langsung di atas peta citra asli. Pengujian dilakukan dengan variasi bobot heuristik untuk mengevaluasi pengaruhnya terhadap jumlah node yang dikunjungi dan panjang lintasan. Hasil menunjukkan bahwa semakin besar bobot heuristik, semakin sedikit node yang dieksplorasi oleh algoritma, meskipun panjang lintasan cenderung sedikit meningkat. Visualisasi berupa heatmap node yang dikunjungi dan grafik perbandingan performa mendukung kesimpulan tersebut. Sistem ini menunjukkan potensi pendekatan navigasi visual yang sederhana namun efektif, dan dapat menjadi dasar untuk implementasi lebih lanjut pada sistem robot fisik berbasis kamera. Pendekatan ini juga cocok untuk pembelajaran dan penelitian karena fleksibel, bebas sensor fisik, dan mudah dimodifikasi di lingkungan simulasi.

PENDAHULUAN

Perkembangan teknologi robotika dan sistem cerdas telah mengalami kemajuan pesat dalam beberapa tahun terakhir. Salah satu aspek penting dalam pengembangan *mobile* robot adalah kemampuan untuk melakukan navigasi secara otonom dalam suatu lingkungan yang tidak sepenuhnya terstruktur. Navigasi otonom mencakup proses penentuan jalur (path planning), penghindaran rintangan, serta kontrol gerak robot untuk mencapai tujuan dengan efisien. Di antara berbagai algoritma pencarian jalur yang telah dikembangkan, algoritma A* (A-star) merupakan salah satu algoritma yang paling banyak digunakan karena efisiensinya dalam menemukan jalur terpendek pada ruang pencarian diskret. A* menggabungkan kelebihan dari algoritma Dijkstra dan pencarian heuristik [1][2], dengan mempertimbangkan biaya total perjalanan dan prediksi jarak ke tujuan, sehingga menjadikannya pilihan utama untuk perencanaan jalur pada *mobile* robot.

Algoritma A* dipilih dalam penelitian ini karena menggabungkan kelebihan dari algoritma Dijkstra yang menjamin pencarian jalur terpendek secara menyeluruh, serta efisiensi dari pendekatan greedy best-first search yang mengutamakan estimasi jarak ke tujuan [3][4]. Berbeda dengan Dijkstra yang mengeksplorasi semua node secara merata tanpa mempertimbangkan arah menuju goal, dan greedy yang cenderung cepat namun tidak selalu menghasilkan jalur optimal, A* mampu menyeimbangkan antara biaya aktual dari titik awal (*g-cost*) dan prediksi ke tujuan (*h-cost*) melalui fungsi heuristik. Kombinasi ini membuat A* lebih efisien dalam menemukan solusi optimal dibanding kedua algoritma tersebut, terutama dalam lingkungan grid yang kompleks dan penuh rintangan.

Simulasi merupakan tahap krusial dalam pengembangan sistem navigasi robot, karena memungkinkan pengujian dan penyempurnaan algoritma dalam lingkungan yang aman dan terkendali sebelum diterapkan pada robot fisik. Salah satu simulator yang populer digunakan dalam riset robotika adalah CoppeliaSim [5][6], yang menyediakan antarmuka pemrograman API dan kemampuan pemodelan lingkungan 3D yang realistis. Dalam konteks penelitian ini, digunakan robot Pioneer 3-DX yang dilengkapi dengan penanda warna untuk membantu deteksi posisi robot dari citra kamera *top-view*. Citra ini kemudian diolah menggunakan metode pengolahan citra digital untuk mengekstraksi informasi termasuk posisi awal robot (*start*), tujuan (*goal*), dan rintangan (*obstacle*).

Penelitian ini mengadopsi pendekatan berbasis visi (*vision-based*) untuk menghasilkan representasi lingkungan dalam bentuk grid biner, di mana sel bernilai 0 menunjukkan area bebas, dan 1 menunjukkan area terhalang. Representasi ini diperoleh melalui segmentasi warna dari citra kamera atas menggunakan *thresholding* berbasis ruang warna BGR. Selanjutnya, algoritma A* diterapkan pada grid tersebut untuk mencari jalur terpendek dari posisi awal menuju titik tujuan. Salah satu keunikan dari pendekatan ini adalah penggunaan langsung dari citra nyata lingkungan dalam simulator, tanpa melakukan pemetaan manual terhadap lokasi rintangan maupun titik awal dan tujuan, yang menjadikannya lebih fleksibel dan realistis untuk skenario dinamis.

Dalam implementasinya, sistem tidak hanya menentukan jalur optimal tetapi juga menyediakan visualisasi proses pencarian berupa heatmap dari node yang dikunjungi serta evaluasi terhadap efisiensi algoritma berdasarkan jumlah node yang di-expand dan panjang lintasan yang dihasilkan. Dengan mengubah parameter heuristik dalam A*, pengaruhnya terhadap efisiensi dan jalur yang dihasilkan juga dianalisis. Hal ini penting karena dalam lingkungan yang nyata, relevansi antara kecepatan pencarian dan kualitas jalur menjadi pertimbangan utama dalam implementasi sistem navigasi cerdas.

Penggunaan algoritma A* dalam perencanaan jalur pada simulasi robot Pioneer 3-DX di lingkungan V-REP berbasis input citra *top-view* memberikan kontribusi pada pengembangan metode navigasi otonom berbasis pengolahan citra. Penelitian ini juga membuka peluang integrasi lebih lanjut dengan sistem cerdas lain seperti pemrosesan berbasis *machine learning*, pengenalan objek, serta penggunaan *semantic map* untuk meningkatkan pemahaman lingkungan oleh robot. Dalam konteks pendidikan dan riset vokasi, pendekatan ini juga dapat menjadi wahana pembelajaran integratif antara pemrograman, pengolahan citra, dan sistem cerdas, sehingga memperkuat keterampilan lintas bidang yang dibutuhkan dalam era revolusi industri 4.0.

Dengan demikian, penelitian ini bertujuan untuk merancang dan mengimplementasikan sistem navigasi robot berbasis A* dengan input citra *real-time* dalam simulator V-REP, serta mengevaluasi efektivitasnya dalam menyelesaikan masalah navigasi dengan mempertimbangkan faktor akurasi jalur, jumlah node yang dikunjungi, dan waktu komputasi. Melalui pendekatan ini, diharapkan diperoleh pemahaman yang lebih mendalam tentang potensi dan keterbatasan metode perencanaan jalur berbasis citra dalam konteks robotika mobile.

TINJAUAN PUSTAKA

Path Planning dalam Robotika

Perencanaan jalur (*path planning*) adalah proses penting dalam sistem robotika *mobile*, yang bertujuan untuk menentukan jalur terbaik bagi robot [7][8][9] agar dapat bergerak dari posisi awal menuju tujuan akhir tanpa menabrak rintangan. Proses ini tidak hanya mempertimbangkan aspek jarak tempuh terpendek, tetapi juga harus memperhatikan keamanan, efisiensi energi, keterbatasan medan, dan dinamika lingkungan sekitar. Dalam konteks robot otonom, kemampuan untuk melakukan *path planning* yang baik menjadi tolok ukur utama dalam penilaian kecerdasan dan kemandirian sistem robot.

Secara umum, *path planning* terbagi menjadi dua kategori utama, yaitu *global path planning* dan *local path planning* [10]. *Global path planning* dilakukan berdasarkan informasi peta lingkungan yang lengkap dan statis, di mana rute optimal dihitung sebelum robot bergerak. Sementara itu, *local path planning* bersifat reaktif dan dilakukan secara *real-time* berdasarkan data dari sensor robot untuk menghadapi lingkungan yang dinamis atau tidak sepenuhnya diketahui. Beberapa pendekatan *path planning* yang banyak digunakan antara lain adalah algoritma graf (seperti Dijkstra dan A*) [11], *sampling-based planning* (PRM, RRT), dan metode berbasis potensi (*potential field*).

Dalam pengembangan sistem robotik yang berbasis simulasi visual seperti pengolahan citra dari kamera atas (top-view), peta lingkungan biasanya dibentuk dalam bentuk grid atau matriks yang menunjukkan area bebas dan area terhalang. Representasi grid ini sangat kompatibel dengan pendekatan algoritmik klasik seperti A*, karena memungkinkan konversi langsung antara posisi piksel dengan koordinat grid.

A* (A-Star)

Algoritma A* merupakan salah satu algoritma pencarian jalur yang paling banyak digunakan dalam bidang robotika dan kecerdasan buatan. A* pertama kali diperkenalkan oleh Peter Hart, Nils Nilsson, dan Bertram Raphael pada tahun 1968 sebagai pengembangan dari algoritma Dijkstra [12]. A* menggabungkan kekuatan pencarian menyeluruh seperti Dijkstra dengan efisiensi heuristik seperti algoritma Best-First Search. A* menggunakan fungsi evaluasi

$$f(n) = g(n) + h(n) \quad (1)$$

di mana $g(n)$ merupakan biaya aktual dari node awal ke node saat ini, sedangkan $h(n)$ adalah estimasi heuristik dari node saat ini ke tujuan.

Kelebihan utama dari A* adalah kemampuannya menemukan jalur optimal apabila fungsi heuristiknya admissible (tidak melebihi biaya sebenarnya ke tujuan) dan consistent (memenuhi segitiga ketat dalam graf). Hal ini menjadikannya pilihan yang sangat baik untuk navigasi robot dalam lingkungan yang diketahui dan terstruktur. Dibandingkan dengan algoritma Dijkstra yang tidak memiliki arahan heuristik, A* mampu mengurangi jumlah node yang dievaluasi sehingga lebih efisien dalam hal waktu komputasi. Di sisi lain, jika dibandingkan dengan Greedy Best-First Search yang hanya mengandalkan $h(n)$, A* tidak mudah terjebak dalam pencarian lokal dan memberikan solusi yang lebih akurat.

Dalam penelitian ini, A* digunakan untuk merencanakan lintasan robot Pioneer 3DX pada lingkungan virtual berbasis citra top-view. Grid navigasi dibentuk dari hasil segmentasi citra menggunakan teknik thresholding warna, yang menghasilkan representasi rintangan dan area bebas secara visual. A* kemudian mencari lintasan dari posisi awal robot ke posisi tujuan berdasarkan peta grid tersebut. Untuk pengujian dan analisis, fungsi heuristik pada A* juga dapat dimodifikasi, seperti dengan memberikan bobot lebih besar pada estimasi heuristik untuk melihat dampaknya terhadap jumlah node yang dikunjungi dan panjang jalur.

Simulasi V-Rep

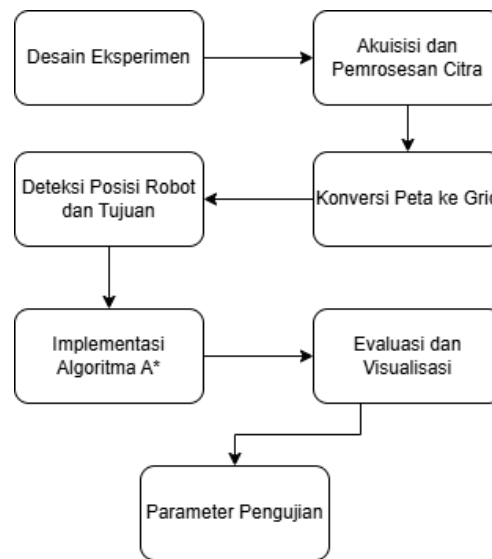
Virtual Robot Experimentation Platform (V-REP) [13], yang kini dikenal sebagai CoppeliaSim, merupakan simulator robot 3D yang mendukung pemodelan dan pemrograman robotik secara visual dan modular. V-REP banyak digunakan dalam pendidikan, penelitian, dan pengembangan karena fleksibilitas dan kapabilitasnya dalam menyimulasikan berbagai jenis robot, sensor, dan lingkungan kompleks secara realistis. Platform ini menyediakan remote API yang memungkinkan integrasi dengan bahasa pemrograman seperti Python, C++, dan MATLAB, sehingga pengguna dapat mengendalikan robot secara dinamis dalam simulasi.

Dalam konteks penelitian ini, V-REP digunakan untuk menyimulasikan robot Pioneer 3DX dalam lingkungan virtual yang dibangun menyerupai arena uji. Robot Pioneer 3DX merupakan robot mobile dua roda diferensial yang banyak digunakan dalam studi navigasi dan kontrol karena desainnya yang sederhana namun fungsional. Pada model simulasi, Pioneer dilengkapi dengan tanda visual berupa lingkaran warna di atasnya, yang memungkinkan pendeteksian posisi dan orientasi melalui citra top-view. Sementara itu, posisi tujuan ditandai dengan kotak berwarna hijau, dan rintangan divisualisasikan sebagai sekat berwarna abu-abu.

Keunggulan V-REP dalam mendukung input dari kamera virtual memungkinkan pendekatan pengolahan citra seperti segmentasi warna untuk digunakan dalam sistem perencanaan jalur. Hal ini menggantikan kebutuhan akan sensor fisik, dan menjadikan seluruh sistem dapat dikembangkan dan diuji secara software-saja. Pendekatan ini sangat berguna dalam fase prototipe atau awal penelitian, sebelum sistem diimplementasikan pada robot fisik. Dengan memanfaatkan simulasi dari V-REP, seluruh proses path planning – mulai dari pembacaan posisi robot dan goal, konversi ke grid, pencarian jalur menggunakan A*, hingga visualisasi lintasan – dapat dikembangkan, diuji, dan dianalisis secara menyeluruh.

METODOLOGI

Pada bagian metodologi ini akan dijelaskan mengenai bagaimana penelitian dilakukan, mulai dari persiapan data, proses pengolahan, hingga pengujian hasil seperti yang terlihat pada Gambar 1 berikut.



Gambar 1. Metodologi Penelitian

Desain Eksperimen

Penelitian ini menggunakan pendekatan simulasi dengan bantuan perangkat lunak CoppeliaSim untuk memodelkan robot Pioneer 3-DX dalam lingkungan virtual. Arena simulasi dilengkapi kamera atas untuk menangkap citra peta lingkungan secara real-time. Citra yang diambil digunakan sebagai dasar pemrosesan untuk membentuk representasi grid navigasi dan posisi robot serta tujuan.

Akuisisi dan Pemrosesan Citra

Gambar diambil dari kamera top-view di CoppeliaSim dan dibaca dalam format BGR. Warna spesifik digunakan sebagai penanda untuk mendeteksi:

1. Dinding (abu-abu dengan nilai RGB sekitar [104,104,104])
2. Posisi robot (lingkaran merah muda atau merah tua)
3. Posisi tujuan (kotak hijau)

Proses thresholding dilakukan menggunakan fungsi `cv2.inRange()` untuk menghasilkan mask biner yang mengindikasikan area terhalang dan area bebas. Hasil threshold kemudian dikonversi ke dalam grid navigasi 0/1 untuk digunakan dalam algoritma A*.

Konversi Peta ke Grid

Citra yang telah ditreshold dikonversi menjadi grid dengan dimensi sesuai resolusi gambar (tanpa resize). Nilai 1 pada grid menandakan rintangan, sedangkan nilai 0 menandakan area bebas. Grid ini digunakan sebagai input utama untuk algoritma perencanaan jalur.

Deteksi Posisi Robot dan Tujuan

Posisi robot ditentukan berdasarkan centroid dari lingkaran berwarna pada tubuh robot yang terdeteksi dari kontur terbesar. Demikian pula posisi tujuan diidentifikasi dari kotak berwarna hijau. Kedua titik tersebut kemudian dikonversi menjadi koordinat grid untuk dijadikan input awal dan akhir dalam proses path planning.

Implementasi Algoritma A*

Algoritma A* digunakan untuk menentukan jalur optimal dari posisi robot ke tujuan pada grid navigasi. Fungsi evaluasi menggunakan: $f(n)=g(n)+w \cdot h(n)$
 w adalah bobot yang dapat disesuaikan. Jalur hasil A* divisualisasikan di atas citra asli untuk menunjukkan hasil rencana lintasan.

Evaluasi dan Visualisasi

Setelah jalur ditemukan, lintasan digambar ulang di atas citra asli menggunakan warna tertentu. Selain itu, jumlah node yang dikunjungi (visited node) dicatat sebagai salah satu indikator efisiensi algoritma. Sebagai tambahan, peta heatmap node yang dikunjungi juga divisualisasikan untuk menunjukkan intensitas eksplorasi grid.

Parameter Pengujian

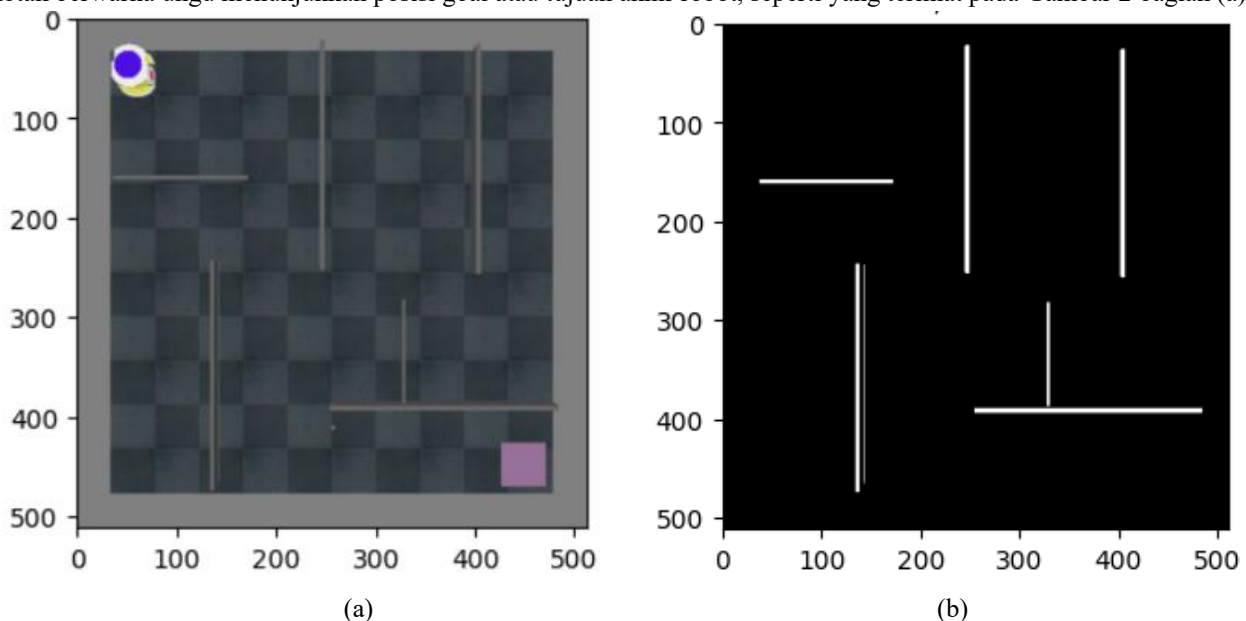
Untuk melihat pengaruh bobot heuristik, dilakukan pengujian dengan variasi nilai bobot w seperti 0.10, 0.5 1, 3, 5. Hasil dibandingkan berdasarkan jumlah node yang dikunjungi dan panjang lintasan yang dihasilkan.

HASIL DAN PEMBAHASAN

Pada bagian ini dipaparkan hasil implementasi algoritma A*(A-Star) dalam proses perencanaan jalur (*path planning*) pada robot Pioneer P3DX dalam simulasi lingkungan V-REP. Proses dimulai dari pengolahan citra arena untuk mendeteksi rintangan, posisi awal robot, dan posisi tujuan (goal), kemudian dilanjutkan dengan konversi ke dalam bentuk grid sebagai representasi peta untuk keperluan pencarian jalur. Setelah itu, algoritma A* dijalankan untuk menghasilkan lintasan optimal dari posisi awal menuju tujuan, sekaligus mencatat node yang dikunjungi selama proses pencarian. Hasil visualisasi lintasan, peta heatmap node yang dieksplorasi, serta pengujian terhadap beberapa variasi parameter heuristik akan ditampilkan dan dibahas lebih lanjut dalam bagian ini untuk mengevaluasi efisiensi serta efektivitas pendekatan yang digunakan.

Visualisasi Tahapan Sistem

Berikut merupakan arena simulasi citra RGB top-view yang merupakan citra hasil tangkapan kamera top-view dari simulasi lingkungan di V-REP/CoppeliaSim. Arena ini terdiri dari Robot Pioneer P3DX ditandai dengan lingkaran warna dibagian atas robot untuk mendeteksi keberadaan robot, area berwarna abu-abu merupakan dinding atau rintangan, serta kotak berwarna ungu menunjukkan posisi goal atau tujuan akhir robot, seperti yang terlihat pada Gambar 2 bagian (a).

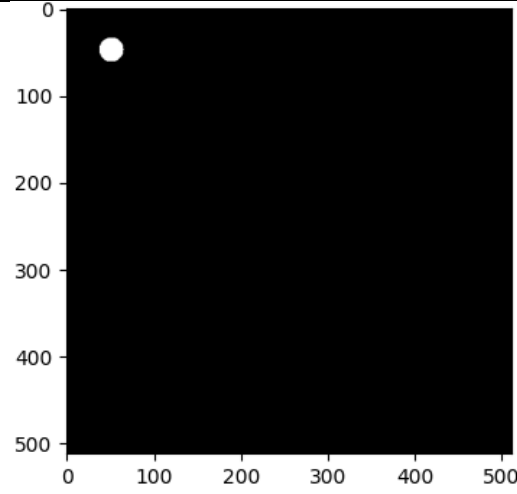
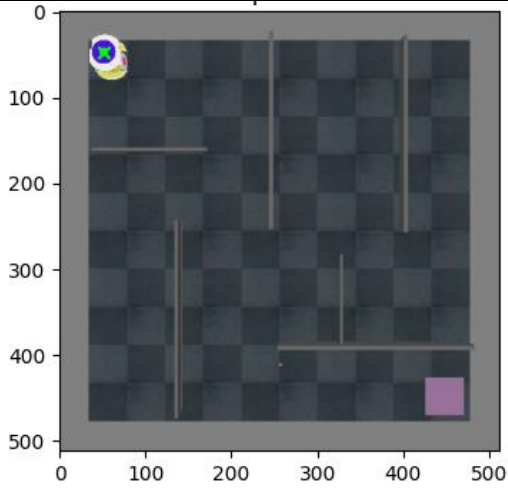
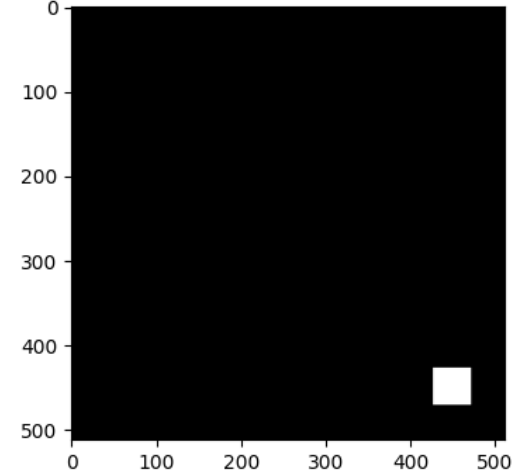
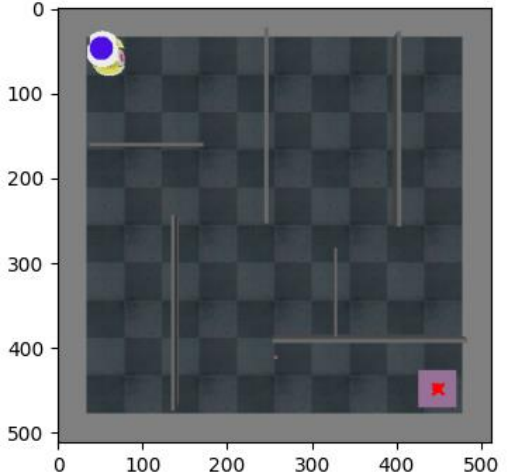


Gambar 2. (a) Gambar Hasil dari Kamera Top-View, (b) Hasil Tresholding Dinding Halangan

Pada Gambar 2 bagian (b) menunjukkan hasil proses thresholding berdasarkan warna dinding (abu-abu dengan nilai RGB sekitar [104,104,104]). Proses ini menghasilkan mask biner, di mana Area hitam terdeteksi sebagai area bebas untuk navigasi serta area putih terdeteksi sebagai rintangan.

Teknik tresholding juga digunakan dalam mengidentifikasi titik keberadaan robot dan goal, dengan warna biru (nilai RGB sekitar [229,15,78]) terdeteksi dan posisinya berada pada koordinat [51, 47] serta warna ungu (nilai RGB sekitar [151, 113, 151]) terdeteksi dan posisinya berada pada koordinat [448, 448] seperti yang terlihat pada Tabel 1 berikut.

Tabel 1. Hasil Proses Tresholding Posisi Robot dan Posisi Goal

Mask Treshold	Deteksi Titik Pada Gambar Asli	Posisi
		[51, 47]
		[448, 448]

Visualisasi Jalur Hasil A* Path Planning dan Heatmap Node yang Dikunjungi

Pengujian dilakukan untuk melihat pengaruh variasi bobot heuristik w terhadap perilaku pencarian jalur algoritma A*. Fokus utama adalah pada panjang jalur yang terbentuk dan jumlah simpul (node) yang dikunjungi selama proses pencarian. Semua pengujian dilakukan pada arena dan posisi awal sampai tujuan yang sama.

Berikut merupakan visualisasi hasil jalur hasil dari A*.

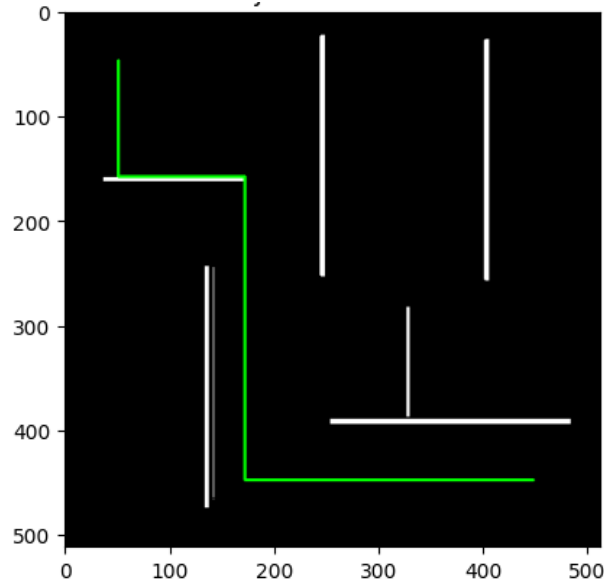
1. bobot w sebesar 0.10

Ringkasan hasil dari pengujian pertama terlihat pada Tabel 2 berikut.

Tabel 2. Hasil dari bobot sebesar 0.10

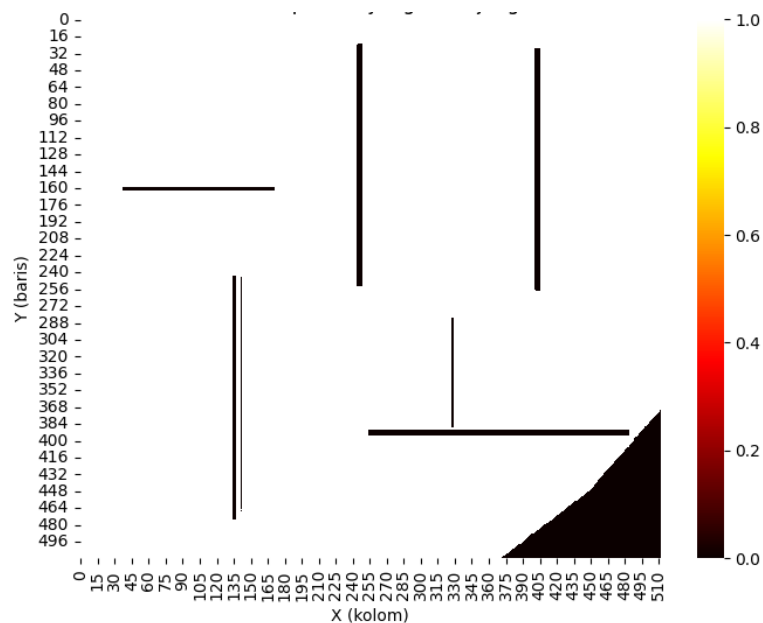
Panjang Jalur	798
Total Node yang Dikunjungi (Visited Node)	247728

Gambar jalur akhir seperti yang terlihat pada Gambar 3 berikut



Gambar 3. Jalur Akhir yang Terbentuk dengan Bobot Sebesar 0.10

Jalur yang dihasilkan mengikuti jalur optimal, tetapi pencarian berlangsung sangat luas seperti yang terlihat pada Gambar 4 berikut.



Gambar 4. Heatmap Node yang Kunjungan dengan Bobot 0.10

Heatmap menunjukkan bahwa hampir seluruh area grid dieksplorasi yang menunjukkan distribusi pencarian yang luas dan intensitas tinggi di berbagai area. Nilai w yang terlalu kecil membuat A^* bekerja seperti Dijkstra, yang melakukan pencarian ke semua arah tanpa mempertimbangkan arah tujuan. Hal ini menyebabkan efisiensi pencarian sangat rendah (visited node sangat besar), meskipun hasil jalur tetap optimal.

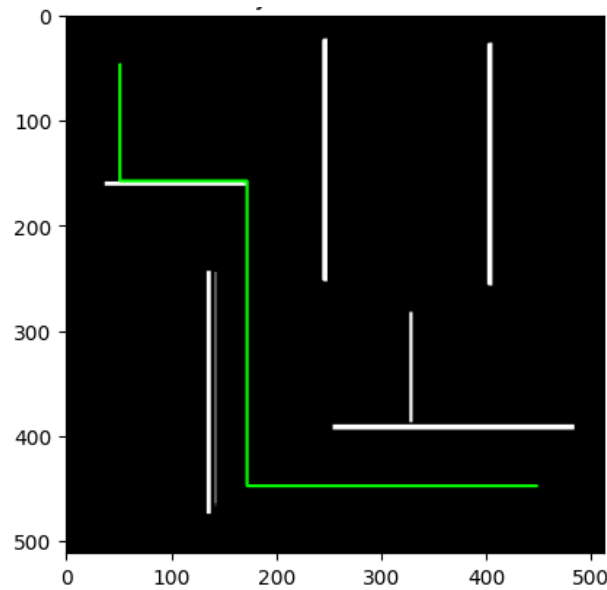
2. bobot w sebesar 0.5

Ringkasan hasil dari pengujian kedua terlihat pada Tabel 3 berikut.

Tabel 3. Hasil dari Bobot Sebesar 0.5

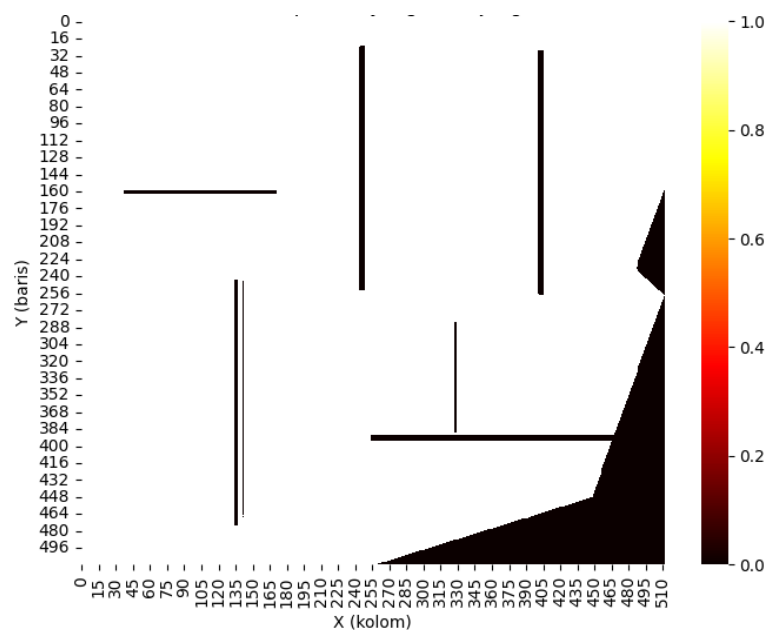
Panjang Jalur	798
Total Node yang Dikunjungi (Visited Node)	239370

Nilai bobot masih tergolong rendah, sehingga algoritma A* tetap mengeksplor cukup luas seperti pada $w = 0.1$. Jalur yang dihasilkan tetap sama seperti pada jalur dengan bobot w sebesar 0.10. Gambar jalur akhir seperti yang terlihat pada Gambar 5 berikut



Gambar 5. Jalur Akhir yang Terbentuk dengan Bobot Sebesar 0.5

Heatmap menunjukkan cakupan pencarian masih sangat luas, meskipun sedikit lebih fokus dibanding $w = 0.1$. Gambar heatmap node yang dikunjungi terlihat pada Gambar 6.



Gambar 6. Heatmap Node yang Kunjungan dengan Bobot 0.5

Dari hasil pengujian yang telah dilakukan, masih terlalu kecil untuk memberikan arah pencarian yang kuat ke goal tetapi sedikit lebih efisien dibanding $w = 0.1$ tapi belum cukup signifikan.

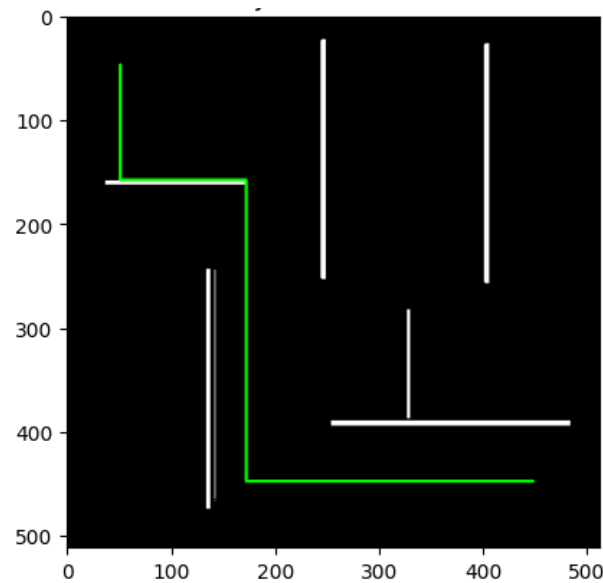
3. bobot w sebesar 1

Ringkasan hasil dari pengujian ketiga terlihat pada Tabel 4 berikut.

Tabel 4. Hasil dari Bobot Sebesar 1

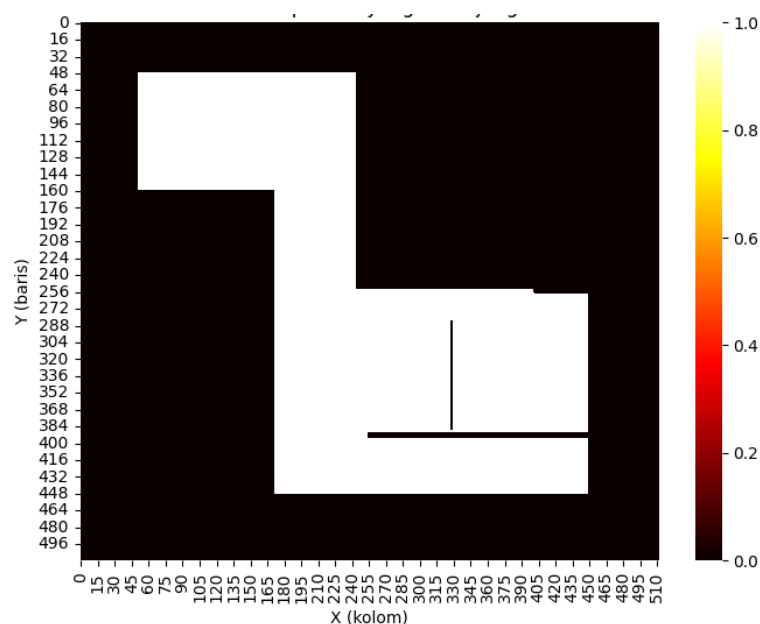
Panjang Jalur	798
Total Node yang Dikunjungi (Visited Node)	80991

Pada $w = 1$, algoritma A* menggunakan heuristik dengan bobot seimbang. Ini adalah bentuk klasik A*, dan sudah mulai memperlihatkan pengurangan signifikan dalam jumlah node yang dikunjungi. Gambar jalur akhir seperti yang terlihat pada Gambar 7 berikut



Gambar 7. Jalur Akhir yang Terbentuk dengan Bobot Sebesar 1

Jalur tetap sama panjang, tetapi heatmap sudah lebih terarah ke jalur yang menuju goal, dengan eksplorasi yang tidak lagi menyebar ke seluruh arena. Ini menunjukkan bahwa fungsi heuristik mulai berperan dalam membatasi ruang eksplorasi. Gambar heatmap node yang dikunjungi terlihat pada Gambar 8 berikut



Gambar 8. Heatmap Node yang Kunjungan dengan Bobot 1

4. bobot w sebesar 3 dan 5

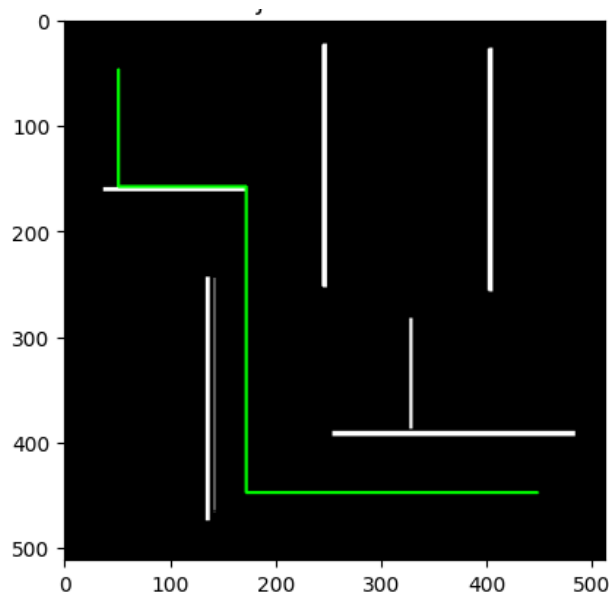
Ringkasan hasil dari pengujian keempat terlihat pada Tabel 7 berikut.

Tabel 7. Hasil dari bobot sebesar 1

Panjang Jalur	798
Total Node yang Dikunjungi (Visited Node)	799

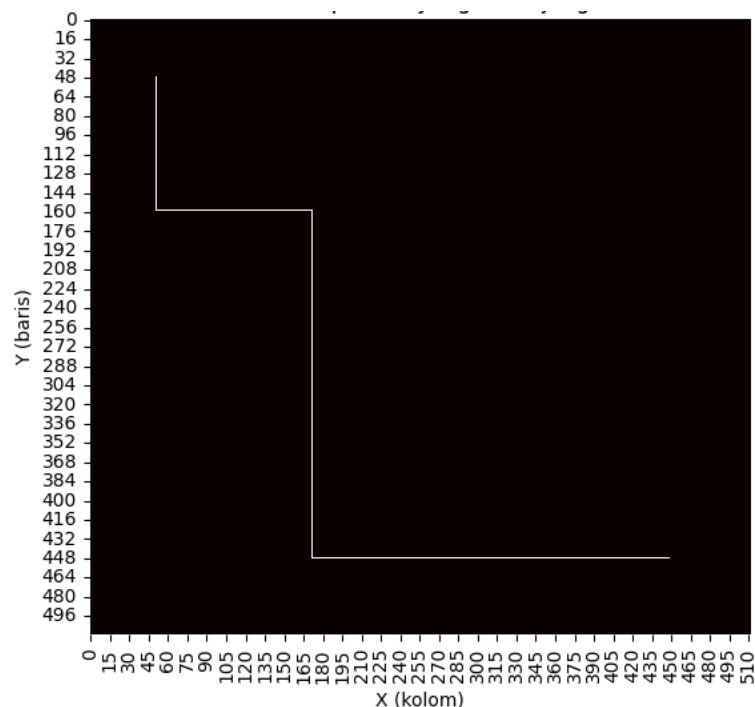
Jalur yang ditemukan tetap identik seperti pengujian sebelumnya. Perubahan besar terjadi pada $w \geq 3$. Dengan

heuristik yang lebih dominan, A* berperilaku semakin greedy, langsung menuju goal dengan sangat efisien. Gambar jalur akhir seperti yang terlihat pada Gambar 9 berikut



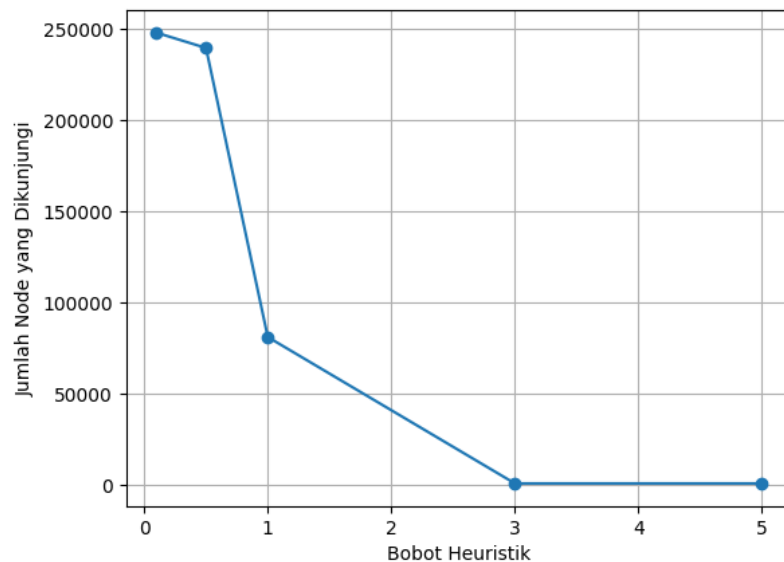
Gambar 9. Jalur Akhir yang Terbentuk dengan Bobot Sebesar 3 dan 5

Heatmap yang dihasilkan sangat sempit dengan hanya mengikuti jalur yang mendekati garis lurus ke goal. Gambar heatmap node yang dikunjungi seperti yang terlihat pada Gambar 10 berikut



Gambar 10. Heatmap Node yang Kunjungan dengan Bobot 3 dan 5

Bobot heuristik yang besar membuat A* semakin mirip dengan Greedy Best-First Search. A* menjadi sangat agresif menuju tujuan, mengeksplorasi jauh lebih sedikit node. Namun karena tidak ada banyak cabang atau rintangan memutar, jalur tetap optimal. Ini menunjukkan bahwa dalam arena sederhana, peningkatan w dapat sangat mengurangi eksplorasi tanpa mengorbankan kualitas jalur.



Gambar 11. Grafik Variasi Bobot Heuristik

Gambar 11. Grafik Perbandingan Jumlah Node yang Dikunjungi dengan Bobot Heuristik menunjukkan hubungan antara variasi bobot heuristik dan jumlah node yang dieksplorasi oleh algoritma A* dalam proses pencarian jalur dari posisi awal ke tujuan. Terlihat bahwa ketika bobot heuristik berada pada nilai rendah (0 hingga 1), jumlah node yang dikunjungi sangat tinggi, bahkan mencapai lebih dari 200.000 simpul. Hal ini disebabkan karena pada bobot heuristik rendah, A* cenderung mengeksplorasi hampir seluruh area pencarian, mirip dengan perilaku algoritma Dijkstra yang tidak menggunakan estimasi arah menuju goal. Akibatnya, ruang pencarian menjadi luas dan waktu komputasi pun meningkat.

Namun, ketika bobot heuristik dinaikkan ke nilai 2 hingga 5, terjadi penurunan jumlah node yang sangat signifikan. Pada nilai bobot heuristik ≥ 3 , algoritma A* menjadi lebih bersifat greedy — ia lebih fokus ke arah tujuan dan mengurangi eksplorasi ke area yang tidak menjanjikan. Ini menyebabkan jumlah node yang dikunjungi turun drastis hingga mendekati nol pada grafik. Artinya, pencarian menjadi jauh lebih efisien dari sisi komputasi dan ruang pencarian. Hal ini menguatkan bahwa penambahan bobot heuristik dapat mengarahkan algoritma A* untuk lebih cepat mencapai tujuan dengan biaya eksplorasi lebih rendah. Akan tetapi, perlu dicatat bahwa efisiensi ini dapat mengorbankan optimalitas jalur, karena semakin greedy sebuah algoritma, semakin besar kemungkinan ia mengabaikan jalur terpendek yang sebenarnya.

KESIMPULAN DAN SARAN

Penelitian ini berhasil mengimplementasikan algoritma A* untuk perencanaan jalur robot mobile Pioneer P3DX dalam lingkungan simulasi V-REP dengan pendekatan berbasis pengolahan citra. Melalui deteksi warna pada citra top-view, posisi robot, tujuan, dan rintangan berhasil diidentifikasi dengan teknik segmentasi warna dan thresholding. Hasil segmentasi kemudian dikonversi ke dalam representasi grid biner yang menjadi input bagi algoritma A* dalam mencari lintasan optimal.

Eksperimen menunjukkan bahwa algoritma A* mampu menemukan jalur yang valid dari posisi awal ke tujuan, dengan performa yang sangat dipengaruhi oleh parameter bobot heuristik. Saat bobot heuristik rendah (misalnya 1), algoritma mengeksplorasi banyak node (mirip Dijkstra), sedangkan pada bobot tinggi (misalnya 5 atau lebih), jumlah node yang dikunjungi turun drastis karena arah pencarian lebih fokus ke goal. Visualisasi berupa heatmap node dan grafik jumlah node yang dikunjungi memperjelas efisiensi dan perilaku algoritma dalam berbagai kondisi. Jalur yang dihasilkan tetap valid, meskipun dalam beberapa kasus terjadi kompromi antara efisiensi eksplorasi dan optimalitas panjang jalur.

Dengan pendekatan berbasis visual dan integrasi dengan simulator V-REP, penelitian ini membuktikan bahwa sistem navigasi dapat dibangun secara fleksibel dan efisien tanpa ketergantungan pada sensor fisik atau peta yang telah ditentukan sebelumnya.

Untuk pengembangan selanjutnya, disarankan untuk menguji algoritma ini pada lingkungan yang lebih dinamis, misalnya dengan menambahkan rintangan bergerak atau arena yang berubah selama simulasi. Perlu dilakukan perbandingan

performa dengan algoritma lain seperti Dijkstra, Greedy Best-First Search, atau algoritma sampling seperti RRT agar dapat mengetahui kelebihan dan kekurangan A* dalam berbagai skenario. Serta implementasi dapat diperluas dengan menggabungkan machine learning atau teknik pengenalan objek agar segmentasi tidak bergantung pada warna statis, namun bisa adaptif terhadap perubahan kondisi pencahayaan.

DAFTAR PUSTAKA

- [1] F. A. Pratama, A. S. Sahay, and N. Nugrahaningsih, "Perbandingan Algoritma Dijkstra dan A-Star dalam Pencarian Rute Terpendek Sekolah Menengah Atas (SMA) di Kota Palangka Raya Berbasis Website," vol. 5, pp. 1–3, 2025.
- [2] Y. H. Nuryoso, P. Pradjoko, and L. Lelah, "Implementasi Algoritma A-Star Untuk Mencari Rute Terpendek Angkutan Umum Kota (Studi Kasus Pada Rute Angkutan Umum Kota di Kota Sukabumi)," *JSTIE (Jurnal Sarj. Tek. Inform.*, vol. 8, no. 1, p. 21, 2020, doi: 10.12928/jstie.v8i1.13922.
- [3] R. D. Saktia Purnama *et al.*, "Implementasi Penggunaan Algoritma Greedy Best First Search Untuk Menentukan Rute Terpendek Dari Cilacap Ke Yogyakarta," *J. Inform. dan Tek. Elektro Terap.*, vol. 12, no. 2, 2024, doi: 10.23960/jitet.v12i2.4068.
- [4] R. Dwi Septiana, D. Abisono Punkastyo, and N. Nugroho, "Algoritma Greedy dan Algoritma A* Untuk Penentuan Cost Pada Routing Jaringan," *KLIK Kaji. Ilm. Inform. dan Komput. Implementasi*, vol. 3, no. 2, pp. 181–187, 2022, [Online]. Available: <https://djournals.com/klik>.
- [5] E. Rohmer, S. P. N. Singh, and M. Freese, "V-REP: A versatile and scalable robot simulation framework," *IEEE Int. Conf. Intell. Robot. Syst.*, pp. 1321–1326, 2013, doi: 10.1109/IROS.2013.6696520.
- [6] G. Montenegro *et al.*, "Modeling and Control of a Spherical Robot in the CoppeliaSim Simulator," *Sensors*, vol. 22, no. 16, 2022, doi: 10.3390/s22166020.
- [7] R. Amirullah, A. Rusdina, and D. Darlis, "Implementasi Sistem Path Planning dan Routing untuk Mobile Robot Berbasis Visible Light Communication Implementation of Path Planning and Routing System Based-on Visible Light Communication for Mobile Robot," vol. 8, no. 5, p. 4283, 2021.
- [8] Luluk Indah Safitri, Julian Sahertian, and Danang Wahyu Widodo, "Implementasi Algoritma Path Planning A* Pada Base Station Robot Sepak Bola Beroda," *Gener. J.*, vol. 7, no. 3, pp. 56–63, 2023, doi: 10.29407/gj.v7i3.20545.
- [9] D. F. Adryady, A. Prasetyo, H. P. Shatyaziamawan, and A. S. Priambodo, "Implementasi Algoritma Path Planning dan Mapping Arena pada Mobile Robot," *J. Tek. Elektro dan Komput. TRIAC*, vol. 9, no. 2, pp. 41–45, 2022, [Online]. Available: <https://journal.trunojoyo.ac.id/triac/article/view/13215>.
- [10] H. Zhao, H. Zhou, and G. Yang, "Research on Global Path Planning of Artificial Intelligence Robot Based on Improved Ant Colony Algorithm," *J. Phys. Conf. Ser.*, vol. 1744, no. 2, 2021, doi: 10.1088/1742-6596/1744/2/022032.
- [11] R. Amelia, V. S. Meliala, S. N. Harahap, M. H. Al-majid, and P. Harliana, "OPTIMALISASI RUTE TRANSPORTASI MENGGUNAKAN ALGORITMA GRAF (STUDI KASUS: JARINGAN TRANSPORTASI PERKOTAAN)," vol. 8, no. 6, pp. 12625–12632, 2024.
- [12] D. O. Pugas, M. Somantri, K. I. Satoto, and N. Nilsson, "Pencarian Rute Terpendek Menggunakan Algoritma Dijkstra dan Astar (A*) pada SIG Berbasis Web untuk Pemetaan Pariwisata Kota Sawahlunto," *Pencarian Rute Terpendek Menggunakan Algoritma Dijkstra dan Astar pada SIG Berbas. Web untuk Pemetaan Pariwisata Kota Sawahlunto*, vol. 13, no. 1, pp. 27–32, 2011.
- [13] E. Apriaskar, F. Fahmizal, I. Cahyani, and A. Mayub, "Autonomous Mobile Robot based on BehaviourBased Robotic using V-REP Simulator–Pioneer P3-DX Robot," *J. Rekayasa Elektr.*, vol. 16, no. 1, 2020, doi: 10.17529/jre.v16i1.15081.