



Evaluasi Kinerja Navigasi Inspeksi Melingkar Otonom Berbasis *Finite State Machine* pada UAV dengan Variasi Jarak di Webots

Rizal Meifan^{1*}, Fikra Titan Syifa²

¹⁻²Program Studi Teknik Elektro, Universitas Telkom Kampus Purwokerto, Indonesia

Email: rizalmeifan@telkomniversity.ac.id¹, fikras@telkomniversity.ac.id²

*Penulis korespondensi: rizalmeifan@telkomniversity.ac.id

Abstract. *Autonomous aerial inspection requires reliable navigation mechanisms to execute sequential flight operations while maintaining trajectory accuracy and returning to the initial position. This study evaluates a Finite State Machine (FSM)-based autonomous circular inspection navigation system implemented in the Webots simulation environment. The proposed navigation system consists of sequential states including takeoff, forward navigation, hovering, circular inspection, return navigation, and landing. The effect of inspection distance on navigation performance was investigated using five target-radius configurations of 3, 5, 7, 10, and 15 m, with three repetitions for each configuration. Performance was evaluated based on mission completion time, altitude root mean square error (RMSE), circular trajectory radius RMSE, yaw RMSE, and return-to-home (RTH) error. The results show that increasing the inspection radius leads to longer mission completion times, ranging from 88.61 s at 3 m to 350.54 s at 15 m. In contrast, altitude RMSE, radius RMSE, and yaw RMSE generally decrease as the inspection radius increases, with the lowest values obtained at 15 m of 0.1230 m, 0.1830 m, and 0.1806 rad, respectively. The RTH error exhibits a non-monotonic response, with the lowest value of 0.1615 m obtained at a 5 m inspection radius. These results indicate that inspection distance affects different aspects of autonomous navigation performance, particularly the trade-off between mission duration and trajectory-tracking accuracy. The findings demonstrate the feasibility of FSM-based autonomous circular inspection navigation in Webots and provide a simulation-based baseline for further robustness evaluation and real-world implementation.*

Keywords: *Autonomous Navigation; Circular Inspection; Finite State Machine; UAV; Webots.*

Abstrak. Navigasi udara otonom untuk kegiatan inspeksi memerlukan mekanisme navigasi yang andal agar mampu menjalankan rangkaian operasi penerbangan secara berurutan dengan tetap mempertahankan akurasi lintasan dan kemampuan kembali ke posisi awal. Penelitian ini mengevaluasi sistem navigasi inspeksi melingkar otonom berbasis *Finite State Machine* (FSM) yang diimplementasikan pada lingkungan simulasi Webots. Sistem navigasi yang dikembangkan terdiri atas beberapa keadaan (*state*) secara berurutan, yaitu *takeoff*, navigasi maju, *hovering*, inspeksi melingkar, navigasi kembali, dan *landing*. Pengaruh jarak inspeksi terhadap kinerja navigasi dievaluasi menggunakan lima konfigurasi radius target, yaitu 3, 5, 7, 10, dan 15 m, dengan tiga kali pengulangan pada setiap konfigurasi. Kinerja sistem dievaluasi berdasarkan waktu penyelesaian misi, *root mean square error* (RMSE) ketinggian, RMSE radius lintasan melingkar, RMSE *yaw*, dan *return-to-home* (RTH) *error*. Hasil penelitian menunjukkan bahwa peningkatan radius inspeksi menyebabkan waktu penyelesaian misi semakin lama, dari 88,61 detik pada radius 3 m menjadi 350,54 detik pada radius 15 m. Sebaliknya, nilai RMSE ketinggian, RMSE radius lintasan, dan RMSE *yaw* secara umum menurun seiring peningkatan radius, dengan nilai terendah pada radius 15 m masing-masing sebesar 0,1230 m, 0,1830 m, dan 0,1806 rad. Sementara itu, RTH *error* menunjukkan pola yang tidak monoton, dengan nilai terendah sebesar 0,1615 m pada radius 5 m. Hasil tersebut menunjukkan bahwa variasi jarak inspeksi memengaruhi beberapa aspek kinerja navigasi otonom, terutama adanya kompromi antara durasi penyelesaian misi dan akurasi pelacakan lintasan. Penelitian ini menunjukkan kelayakan penerapan navigasi inspeksi melingkar berbasis FSM pada lingkungan simulasi Webots serta memberikan dasar evaluasi berbasis simulasi untuk pengujian ketahanan sistem dan implementasi pada UAV secara nyata.

Kata kunci: *Finite State Machine; Inspeksi Melingkar; Navigasi Otonom; UAV; Webots.*

1. LATAR BELAKANG

Perkembangan teknologi robotika dan sistem otonom telah mendorong pemanfaatan *Unmanned Aerial Vehicle* (UAV) pada berbagai aplikasi, termasuk pemantauan, surveilans, inspeksi, dan pengumpulan informasi di lingkungan yang sulit dijangkau manusia. Pada aplikasi inspeksi, UAV tidak hanya dituntut untuk mencapai suatu lokasi tujuan, tetapi juga

harus mampu menjalankan rangkaian perilaku penerbangan secara sistematis, mulai dari lepas landas, navigasi menuju area inspeksi, pelaksanaan pola inspeksi, kembali ke posisi awal, hingga melakukan pendaratan. Kompleksitas rangkaian tersebut membutuhkan mekanisme pengendalian yang mampu mengatur perubahan perilaku UAV secara terstruktur dan konsisten.

Salah satu pendekatan yang banyak digunakan untuk mengorganisasi perilaku sistem robotika adalah *Finite State Machine* (FSM), yang merepresentasikan perilaku robot ke dalam sejumlah state dengan kondisi transisi yang terdefinisi secara jelas. FSM menawarkan struktur kendali yang modular, deterministik, relatif mudah dianalisis, serta memiliki beban komputasi yang ringan sehingga sesuai untuk eksekusi *real time* pada platform dengan sumber daya terbatas (Artigas et al., 2024). Berbagai penelitian telah menunjukkan penerapan FSM pada robot berjalan (Hussain et al., 2019), navigasi dan penghindaran rintangan (Plaza et al., 2020), pengendalian formasi multi-robot melalui *positioning*, *alignment*, *moving*, dan *dispersion* (Navarro et al., 2009), serta pengaturan perilaku kendaraan otonom berbasis transisi kejadian (Wang et al., 2023). FSM juga telah diterapkan untuk mengorganisasi perilaku mesin otonom (Rossander & Lideskog, 2023) dan dikombinasikan dengan *subsumption architecture* serta sistem berbasis pengetahuan untuk meningkatkan kinerja navigasi (Othman & Rad, 2020). Berbagai penerapan tersebut menunjukkan bahwa FSM sesuai digunakan sebagai mekanisme pengendalian tingkat tinggi pada sistem robot otonom yang memiliki tahapan operasi terstruktur dan membutuhkan respons deterministik.

Karakteristik tersebut menjadikan *Finite State Machine* (FSM) relevan untuk misi inspeksi yang memiliki urutan operasi yang jelas. Correll dan Martinoli (2009), misalnya, menggunakan struktur FSM dalam sistem inspeksi multi-robot untuk mengatur perubahan perilaku robot selama proses pencarian dan inspeksi. Prorok et al. (2011) juga memodelkan perilaku robot menggunakan deterministic FSM dalam konteks sistem robotik terdistribusi untuk melakukan aktivitas inspeksi dan kunjungan terhadap area tertentu. Pada aplikasi *surveillance* berbasis *swarm robotics*, Brasiel dan Lima (2024) menggunakan FSM untuk merepresentasikan perilaku robot dalam menjalankan tugas *surveillance*. Selanjutnya, Hejase dan Özgüner (2022) mengembangkan kerangka otomasi operasi *load-haul-dump* dengan memanfaatkan FSM sebagai mekanisme pengambilan keputusan tingkat tinggi, di mana setiap tahapan operasi direpresentasikan sebagai *state* dan kondisi tertentu digunakan sebagai pemicu perpindahan antar *state*. Pendekatan tersebut diintegrasikan dengan pengendali *path tracking* dan divalidasi melalui simulasi. Temuan-temuan tersebut menunjukkan bahwa FSM memiliki

potensi sebagai arsitektur kendali tingkat tinggi pada sistem robot otonom yang memiliki beberapa tahapan operasi.

Dalam konteks inspeksi menggunakan UAV, salah satu pola navigasi yang dapat digunakan adalah *circular inspection*, yaitu UAV bergerak mengelilingi suatu titik pusat dengan radius tertentu. Pola ini memungkinkan sensor UAV memperoleh sudut pandang dari berbagai arah terhadap objek atau area yang diamati. Secara geometris, lintasan melingkar dapat direpresentasikan melalui koordinat pusat dan radius lintasan. Namun, kemampuan UAV dalam mempertahankan lintasan tersebut dapat dipengaruhi oleh berbagai parameter, salah satunya adalah jarak atau radius inspeksi. Perubahan radius menyebabkan perubahan panjang lintasan yang harus ditempuh serta karakteristik gerakan UAV selama melakukan inspeksi. Penelitian Moreno et al. (2023) menunjukkan bahwa pengujian *path tracking* pada lintasan berbentuk lingkaran dapat digunakan untuk mengevaluasi kemampuan sistem dalam mempertahankan lintasan yang telah ditentukan. Hal ini menunjukkan pentingnya evaluasi kuantitatif terhadap kesalahan lintasan dalam pengujian sistem navigasi.

Selain akurasi lintasan, sistem inspeksi otonom juga perlu dievaluasi berdasarkan keberhasilan penyelesaian keseluruhan misi. Cheng et al. (2022) mengembangkan Robot *Navigation Analysis Platform* (RoNAP) untuk menganalisis performa algoritma navigasi robot dengan mempertimbangkan informasi posisi, jarak terhadap tujuan, dan keberhasilan penyelesaian tugas. Pendekatan evaluasi semacam ini penting karena performa navigasi tidak hanya ditentukan oleh kemampuan robot mengikuti lintasan, tetapi juga oleh kemampuan sistem menyelesaikan seluruh rangkaian operasi yang telah ditentukan.

Pengembangan dan pengujian sistem navigasi UAV secara langsung pada perangkat fisik memiliki keterbatasan berupa kebutuhan biaya, waktu, serta risiko kerusakan perangkat ketika terjadi kegagalan algoritma. Oleh karena itu, lingkungan simulasi dapat digunakan sebagai tahap awal untuk mengembangkan dan mengevaluasi algoritma navigasi. Webots merupakan simulator robotika yang menyediakan lingkungan tiga dimensi, model fisika, sensor, aktuator, dan berbagai model robot yang dapat digunakan untuk melakukan eksperimen secara sistematis. Collins et al. (2021) menempatkan Webots sebagai salah satu simulator fisika yang banyak digunakan dalam aplikasi robotika, sementara Baumann dan Martinoli (2022) menunjukkan pemanfaatan simulator dengan tingkat fidelitas tinggi untuk memodelkan perilaku robot secara realistis. Webots juga telah digunakan untuk pengembangan dan pengujian sistem robotika pada berbagai aplikasi, termasuk pertanian, *swarm robotics*, navigasi, dan UAV (Emmi et al., 2013; Rasouli et al., 2024; Uğurlu et al., 2022).

Pada penelitian UAV, Uğurlu et al. (2022) menggunakan Webots untuk mengembangkan dan mengevaluasi perencanaan lintasan UAV berbasis *deep reinforcement learning*. Sementara itu, Trihinas et al. (2021) memanfaatkan Webots sebagai bagian dari pengujian aplikasi drone berbasis machine learning. Penggunaan simulator dalam penelitian-penelitian tersebut menunjukkan bahwa lingkungan simulasi dapat digunakan untuk mengevaluasi perilaku UAV sebelum sistem diterapkan pada lingkungan nyata. Mañas-Álvarez et al. (2023) juga menunjukkan kemampuan Webots dalam mendukung simulasi sistem *cyber-physical* dengan robot virtual, sehingga memungkinkan eksperimen dilakukan secara terkontrol dan berulang.

Meskipun penelitian sebelumnya telah banyak membahas penerapan FSM untuk pengendalian robot, penggunaan Webots untuk simulasi UAV, serta evaluasi *path tracking*, evaluasi khusus terhadap kinerja FSM pada keseluruhan misi *circular inspection* UAV dengan variasi jarak inspeksi masih perlu dikaji secara lebih terukur. Sebagian penelitian berfokus pada perilaku tertentu, seperti navigasi, formasi, *obstacle avoidance*, atau *path tracking*, sedangkan penelitian ini mempertimbangkan rangkaian misi secara utuh yang terdiri atas *takeoff*, *forward navigation*, *circular inspection*, *return navigation*, dan *landing*. Dengan demikian, evaluasi tidak hanya diarahkan pada bentuk lintasan selama fase *circular inspection*, tetapi juga mencakup kemampuan UAV menyelesaikan keseluruhan misi dan kembali ke posisi awal.

Penelitian ini menggunakan lima variasi jarak inspeksi, yaitu 3 m, 5 m, 7 m, 10 m, dan 15 m, dengan masing-masing konfigurasi dilakukan sebanyak tiga kali pengulangan. Variasi tersebut digunakan untuk menganalisis pengaruh jarak terhadap beberapa parameter kinerja navigasi, meliputi waktu penyelesaian misi, kesalahan ketinggian, kesalahan radius lintasan melingkar, kesalahan orientasi *yaw*, serta kesalahan posisi UAV terhadap posisi awal setelah menyelesaikan misi. Data setiap percobaan direkam dari simulasi Webots dan selanjutnya dianalisis menggunakan statistik deskriptif untuk memperoleh karakteristik performa pada setiap variasi jarak.

Berdasarkan permasalahan tersebut, penelitian ini bertujuan mengevaluasi kinerja sistem navigasi otonom UAV berbasis *Finite State Machine* pada misi *circular inspection* menggunakan Webots dengan variasi jarak 3 m, 5 m, 7 m, 10 m, dan 15 m. Hasil penelitian diharapkan dapat memberikan gambaran mengenai pengaruh jarak inspeksi terhadap akurasi lintasan, kestabilan navigasi, waktu penyelesaian misi, dan kemampuan UAV kembali ke posisi awal. Selain itu, penelitian ini diharapkan dapat menjadi dasar untuk pengembangan dan pengujian lebih lanjut pada UAV fisik.

2. KAJIAN TEORITIS

Autonomous UAV Navigation

Navigasi otonom merupakan salah satu aspek fundamental dalam pengembangan sistem robotika, termasuk *Unmanned Aerial Vehicle* (UAV). Sistem navigasi memungkinkan robot menentukan dan menjalankan pergerakan menuju tujuan berdasarkan informasi lingkungan dan kondisi internal robot. Pada UAV, navigasi melibatkan beberapa aspek, antara lain estimasi posisi, pengendalian sikap, pengaturan ketinggian, serta pengendalian pergerakan. Penggunaan sensor seperti *Global Positioning System* (GPS), *Inertial Measurement Unit* (IMU), kompas, dan sensor lainnya dapat digunakan untuk memperoleh informasi posisi dan orientasi UAV.

Dalam penelitian navigasi UAV berbasis pembelajaran penguatan, Uğurlu et al. menggunakan Webots untuk membangun lingkungan tiga dimensi dan mengembangkan sistem perencanaan lintasan untuk robot udara. Penelitian tersebut menunjukkan bahwa lingkungan simulasi dapat digunakan untuk mengembangkan dan mengevaluasi algoritma navigasi UAV sebelum diterapkan pada lingkungan nyata. Selain itu, Trihinas et al. menggunakan Webots sebagai lingkungan pengujian untuk aplikasi UAV berbasis machine learning, yang menunjukkan pemanfaatan simulator untuk mengevaluasi perilaku aplikasi drone secara sistematis.

Pada penelitian ini, navigasi UAV tidak diarahkan pada perencanaan lintasan berbasis pembelajaran, tetapi menggunakan pendekatan berbasis *Finite State Machine* (FSM). Sistem navigasi dirancang untuk menjalankan rangkaian misi secara berurutan, mulai dari *takeoff*, navigasi menuju area inspeksi, pelaksanaan lintasan melingkar, kembali menuju posisi awal, hingga landing

Finite State Machine pada Sistem Robotika

Finite State Machine (FSM) merupakan pendekatan pemodelan perilaku sistem yang membagi operasi menjadi sejumlah keadaan (*state*) dan aturan perpindahan (*transition*) berdasarkan kondisi tertentu. Pendekatan ini memungkinkan perilaku sistem yang kompleks diuraikan menjadi beberapa kondisi operasi yang lebih sederhana dan terstruktur. Dalam sistem robotika, FSM telah digunakan untuk mengorganisasi perilaku robot pada berbagai aplikasi, seperti navigasi, penghindaran rintangan, pengendalian formasi, *surveillance*, dan koordinasi robot.

Navarro et al. menggunakan FSM untuk mengimplementasikan pengendalian formasi pada sistem multi-robot. Sistem tersebut membagi perilaku robot ke dalam beberapa keadaan, seperti *positioning*, *alignment*, *moving*, dan *dispersion*, dengan perpindahan keadaan ditentukan berdasarkan kondisi posisi robot. Pendekatan serupa juga digunakan oleh Hussain

et al. pada sistem kendali *walking machine*, dengan FSM digunakan untuk mengorganisasi perilaku robot berdasarkan kondisi navigasi dan respons terhadap lingkungan.

Plaza et al. menerapkan FSM pada sistem robotika berbasis ROS untuk mengorganisasi perilaku robot, termasuk perilaku bergerak dan menghindari rintangan. Sementara itu, Brasiel dan Lima menggunakan FSM untuk merepresentasikan perilaku robot e-puck dalam aplikasi surveillance berbasis swarm robotics yang disimulasikan menggunakan Webots. Rossander dan Lideskog juga menunjukkan penerapan FSM pada sistem mesin otonom, dengan perilaku sistem dibagi menjadi sejumlah keadaan yang memiliki fungsi dan kondisi transisi tertentu.

Berdasarkan penelitian tersebut, FSM sesuai digunakan pada sistem yang memiliki urutan operasi yang jelas dan bersifat deterministik. Pada penelitian ini, FSM digunakan sebagai pengendali tingkat tinggi (*high-level controller*) untuk menentukan mode operasi UAV. Setiap *state* merepresentasikan satu tahapan misi penerbangan, sedangkan transition ditentukan berdasarkan kondisi yang telah ditetapkan.

FSM pada penelitian ini terdiri atas beberapa keadaan utama, yaitu *Takeoff*, *Forward Navigation*, *Hover*, *Circular Inspection*, *Return Navigation*, dan *Landing*. Struktur tersebut memungkinkan keseluruhan misi inspeksi dijalankan secara berurutan dan dapat dievaluasi berdasarkan kinerja setiap tahapan.

Circular Inspection Navigation

Circular inspection navigation merupakan pola pergerakan UAV dengan mengikuti lintasan melingkar terhadap suatu titik pusat inspeksi. Pola ini memungkinkan UAV melakukan observasi dari berbagai arah terhadap objek atau area yang menjadi target inspeksi. Dalam konteks penelitian ini, titik pusat digunakan sebagai referensi geometris untuk menentukan lintasan UAV selama fase circular inspection.

Secara ideal, lintasan lingkaran pada bidang horizontal dapat direpresentasikan menggunakan persamaan parametrik:

$$x(t) = x_c + r \cos(\theta(t)) \quad (1)$$

$$y(t) = y_c + r \sin(\theta(t)) \quad (2)$$

Dengan x_c dan y_c merupakan koordinat titik pusat inspeksi, r merupakan radius lintasan, sedangkan θ merupakan sudut posisi UAV terhadap titik pusat.

Pendekatan lintasan melingkar berkaitan dengan permasalahan *path tracking*, yaitu kemampuan robot untuk mempertahankan pergerakan aktual agar tetap mendekati lintasan yang telah ditentukan. Moreno et al. misalnya, mengevaluasi *path tracking* pada robot pertanian menggunakan beberapa bentuk lintasan, termasuk lintasan melingkar. Hasil tersebut

menunjukkan bahwa lintasan berbentuk lingkaran dapat digunakan sebagai salah satu skenario untuk mengevaluasi kemampuan sistem dalam mempertahankan jalur yang ditentukan.

Pada penelitian ini, radius lintasan r menjadi variabel eksperimen. Lima nilai jarak inspeksi digunakan, yaitu 3 m, 5 m, 7 m, 10 m, dan 15 m. Setiap konfigurasi dilakukan sebanyak tiga kali pengulangan. Variasi tersebut digunakan untuk mengevaluasi pengaruh jarak inspeksi terhadap akurasi lintasan dan kinerja keseluruhan misi UAV.

Dengan demikian, evaluasi tidak hanya mempertimbangkan apakah UAV berhasil menyelesaikan lintasan melingkar, tetapi juga mempertimbangkan radius *trajectory error*, *yaw error*, *altitude error*, waktu penyelesaian misi, serta *return to home error*.

Webots sebagai Lingkungan Simulasi

Webots merupakan lingkungan simulasi robotika tiga dimensi yang menyediakan model robot, sensor, aktuator, lingkungan fisik, serta *physics engine* untuk melakukan pengujian sistem robotika. Collins et al. menempatkan Webots sebagai salah satu simulator robotika yang banyak digunakan untuk pemodelan dan pengujian sistem robot, khususnya karena kemampuannya dalam mensimulasikan berbagai sensor, aktuator, dan interaksi fisik.

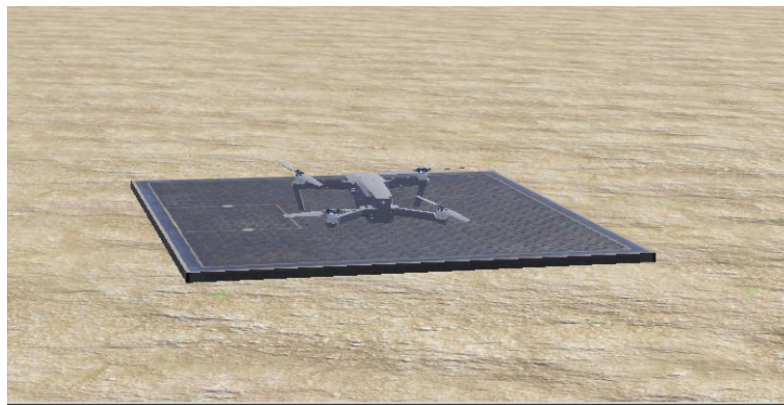
Baumann dan Martinoli menjelaskan bahwa Webots termasuk simulator dengan tingkat fidelitas yang tinggi dan dapat digunakan untuk pemodelan mikroskopis sistem robotika. Kemampuan tersebut memungkinkan karakteristik robot dan interaksi dengan lingkungan direpresentasikan dalam lingkungan simulasi sebelum dilakukan implementasi pada robot fisik.

Webots telah digunakan secara luas dalam berbagai penelitian robotika sebagai lingkungan simulasi untuk pengembangan, pengujian, dan evaluasi sistem robot otonom. Emmi et al. (2013) memanfaatkan lingkungan simulasi untuk pengembangan sistem robot pada aplikasi pertanian presisi, sedangkan Mañas-Álvarez et al. (2023) menggunakan Webots dalam penelitian *cyber-physical systems* yang melibatkan robot virtual dan robot nyata. Pada aplikasi UAV, Uğurlu et al. (2022) memanfaatkan Webots untuk membangun lingkungan simulasi tiga dimensi dalam pengembangan dan evaluasi perencanaan lintasan robot udara, sementara Trihinas et al. (2021) menggunakan Webots sebagai testing suite untuk menguji aplikasi UAV berbasis *machine learning*. Dalam konteks navigasi berbasis pembelajaran, Iskandar dan Kovács (2024) menggunakan Webots untuk memodelkan lingkungan tiga dimensi sebagai media pelatihan algoritma *reinforcement learning*, dengan mengembangkan kemampuan algoritma *Proximal Policy Optimization* (PPO) dalam menangani *continuous state* yang diperoleh dari sensor inframerah (IR) serta mengendalikan kecepatan motor robot e-puck. Berbagai penelitian tersebut menunjukkan bahwa Webots memiliki fleksibilitas dalam

mendukung simulasi sistem robotika pada berbagai skenario, mulai dari robot pertanian, *cyber-physical systems*, navigasi UAV, hingga pembelajaran berbasis *reinforcement learning*, sehingga relevan digunakan sebagai lingkungan simulasi untuk pengembangan dan evaluasi sistem navigasi UAV otonom dalam penelitian ini.

Berdasarkan kemampuan tersebut, Webots digunakan dalam penelitian ini sebagai platform simulasi untuk mengimplementasikan UAV dan menguji pengendali FSM. Lingkungan simulasi dirancang agar UAV dapat menjalankan keseluruhan rangkaian misi, yaitu *takeoff*, *forward navigation*, *hover*, *circular inspection*, *return navigation*, dan *landing*.

Penggunaan simulasi memungkinkan setiap konfigurasi jarak diuji secara berulang dengan kondisi eksperimen yang relatif terkontrol. Data posisi, ketinggian, orientasi, waktu, dan parameter navigasi lainnya direkam selama proses simulasi dan disimpan dalam format CSV untuk selanjutnya dianalisis secara kuantitatif.

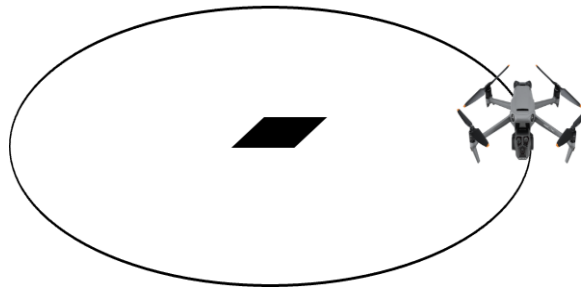


Gambar 1. Tampilan Robot UAV dalam Simulasi Webots.

3. METODE PENELITIAN

Desain Penelitian

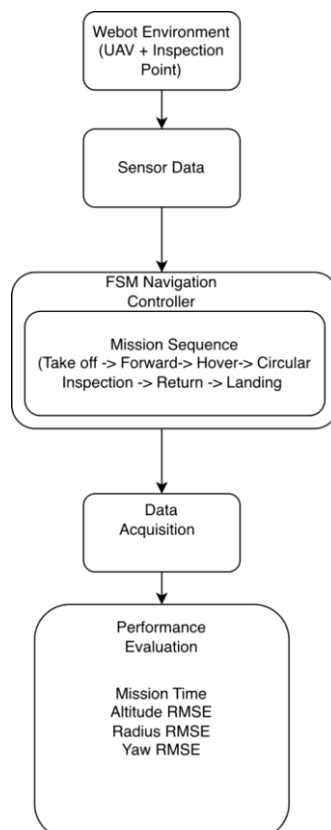
Penelitian ini menggunakan pendekatan eksperimental berbasis simulasi. Eksperimen dilakukan dengan menjalankan UAV secara otonom pada lingkungan Webots menggunakan pengendali FSM. Variabel bebas penelitian adalah jarak navigasi/inspeksi, sedangkan variabel terikat berupa metrik performa navigasi. Proses pengambilan data dilakukan sebanyak tiga kali pengulangan pada setiap variasi jarak pengujian. Pengujian dilakukan pada lima variasi jarak, yaitu 3, 5, 7, 10, dan 15 meter.



Gambar 2. Metode Pengambilan Data

Arsitektur Sistem

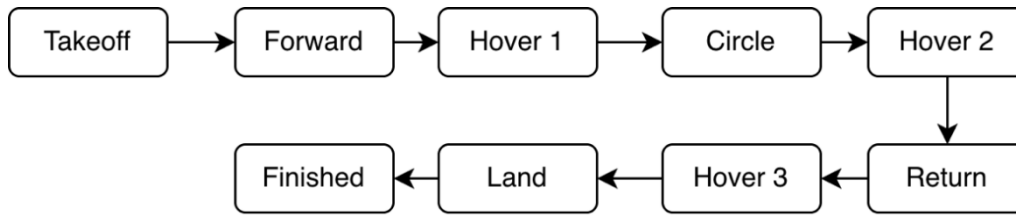
Secara umum sistem penelitian terdiri atas sensor, pengendali FSM, pengendali penerbangan, aktuator propeller, dan sistem perekaman data. Alur sistem dapat dijelaskan sebagai berikut. Sensor UAV menyediakan informasi posisi, ketinggian, orientasi, dan kecepatan sudut. Data sensor kemudian digunakan oleh FSM untuk menentukan state aktif. Berdasarkan state tersebut, sistem menghasilkan masukan kendali berupa *roll*, *pitch*, *yaw*, dan *vertical* input. Masukan tersebut diterapkan pada empat motor propeller UAV di dalam lingkungan Webots. Seluruh parameter penerbangan direkam ke dalam file CSV untuk analisis lebih lanjut.



Gambar 3. Arsitektur Sistem Navigasi UAV berbasis FSM.

Finite State Machine

FSM yang digunakan terdiri atas sembilan keadaan yaitu :



Gambar 4. Urutan tahapan algoritma *Finite State Machine* (FSM) pada UAV.

Pada keadaan *takeoff*, UAV meningkatkan ketinggian hingga mencapai target *altitude*. Setelah ketinggian mencapai kondisi yang ditentukan, UAV berpindah ke keadaan *forward*.

Pada keadaan *forward*, UAV bergerak menuju area inspeksi berdasarkan jarak yang ditentukan. Setelah jarak terpenuhi, UAV masuk ke keadaan *hover 1* untuk melakukan stabilisasi selama interval waktu tertentu.

Keadaan *circle* merupakan fase utama penelitian. UAV mempertahankan jarak terhadap titik pusat sambil melakukan rotasi lintasan hingga mencapai satu putaran penuh atau 360°. Setelah orbit selesai, UAV masuk ke *hover 2*, kemudian berpindah ke *return* untuk kembali menuju titik awal. Setelah mencapai jarak kembali yang ditentukan, UAV melakukan *hover 3* kemudian berpindah ke *land*. Setelah *altitude* mendekati nol, sistem berpindah ke keadaan *finished* dan seluruh motor dihentikan.

Kendali Circular Inspection

Pada fase *circle*, radius aktual terhadap titik pusat dihitung menggunakan persamaan 3 dan kesalahan radius dihitung sesuai dengan persamaan 4.

$$r_{actual} = \sqrt{(x - x_c)^2 + (y - y_c)^2} \quad (3)$$

$$e_r = r_{actual} - r_{target} \quad (4)$$

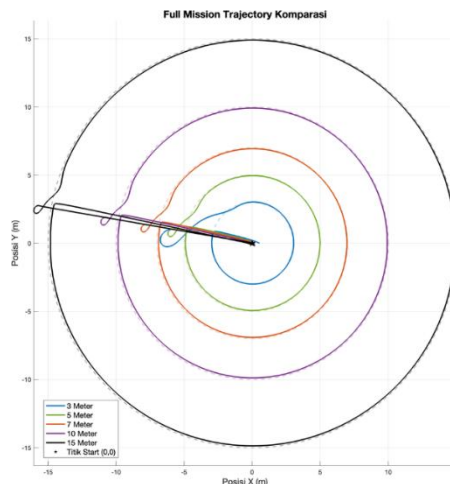
Pengendali kemudian memberikan koreksi terhadap UAV untuk mempertahankan radius mendekati nilai target. Orientasi UAV terhadap pusat inspeksi ditentukan menggunakan persamaan 5. Selanjutnya kesalahan yaw dihitung berdasarkan perbedaan antara orientasi target dan orientasi aktual. Penyelesaian satu orbit ditentukan berdasarkan akumulasi perubahan sudut UAV terhadap titik pusat hingga mencapai 360 derajat.

$$\psi_{target} = \text{atan2}(y_c - y, x_c - x) \quad (5)$$

4. HASIL DAN PEMBAHASAN

Validasi Keseluruhan Misi

Hasil simulasi menunjukkan bahwa sistem FSM digunakan untuk menjalankan misi secara berurutan mulai dari *takeoff*, navigasi maju, *hover*, inspeksi melingkar, *return*, dan *landing*. Setiap fase ditentukan berdasarkan kondisi sensor dan parameter navigasi yang telah ditetapkan.



Gambar 3. Trajectory Keseluruhan Misi UAV pada Lingkungan Webots.

Gambar 3 menunjukkan perbandingan lintasan penuh (*full mission trajectory*) UAV pada lima variasi jarak inspeksi, yaitu 3, 5, 7, 10, dan 15 m. Lintasan mencakup rangkaian pergerakan UAV dari titik awal menuju area inspeksi, pelaksanaan lintasan melingkar, hingga pergerakan kembali menuju area titik awal. Secara visual, peningkatan jarak inspeksi menghasilkan peningkatan radius lintasan secara konsisten, sehingga konfigurasi 3 m menghasilkan lintasan paling kecil, sedangkan konfigurasi 15 m menghasilkan lintasan paling besar. Pada *fase circular inspection*, seluruh konfigurasi menunjukkan pola lintasan yang mendekati bentuk lingkaran dengan pusat di sekitar titik referensi (0,0). Hal ini menunjukkan bahwa sistem navigasi berbasis FSM mampu menjalankan pola pergerakan yang sesuai dengan parameter jarak inspeksi yang ditentukan.

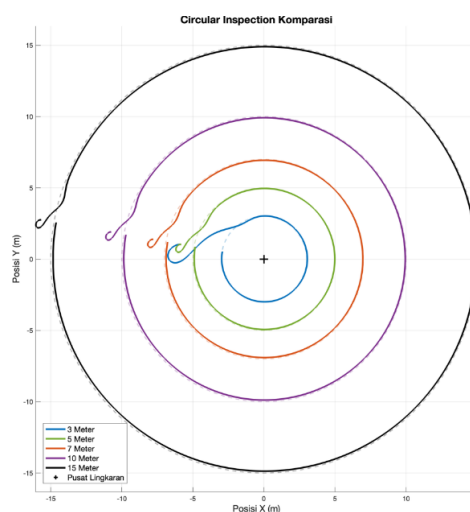
Selain pola lintasan melingkar, Gambar 3 juga memperlihatkan adanya segmen lintasan pada awal dan akhir misi yang menghubungkan titik awal UAV dengan area inspeksi. Segmen tersebut merepresentasikan proses navigasi menuju dan kembali dari area inspeksi, sehingga evaluasi *trajectory* tidak hanya mencerminkan keberhasilan UAV dalam melakukan manuver melingkar, tetapi juga keseluruhan pelaksanaan misi. Meskipun secara visual lintasan aktual terlihat mendekati lintasan yang diharapkan, tingkat penyimpangan *trajectory* tidak dapat ditentukan hanya berdasarkan visualisasi. Oleh karena itu, analisis kuantitatif selanjutnya dilakukan menggunakan radius RMSE untuk mengevaluasi akurasi lintasan melingkar, serta

mission time, *altitude RMSE*, *yaw RMSE*, dan *RTH error* untuk mengevaluasi performa navigasi UAV secara keseluruhan.

Circular Inspection Trajectory

Gambar 4 menunjukkan perbandingan lintasan aktual UAV terhadap lintasan referensi pada proses *circular inspection* dengan variasi radius 3, 5, 7, 10, dan 15 m. Garis putus-putus menunjukkan lintasan referensi berbentuk lingkaran, sedangkan garis penuh menunjukkan lintasan aktual UAV hasil simulasi. Secara umum, lintasan aktual pada seluruh konfigurasi mampu mengikuti pola lingkaran dan mempertahankan pusat lintasan pada titik yang sama. Hal ini menunjukkan bahwa pengendali berbasis *Finite State Machine* (FSM) mampu mengarahkan UAV untuk menyelesaikan fase inspeksi melingkar pada berbagai jarak yang diuji. Perbedaan antara lintasan aktual dan referensi relatif kecil pada sebagian besar bagian lintasan, terutama pada konfigurasi 7, 10, dan 15 m. Dengan demikian, perubahan radius inspeksi tidak menyebabkan kegagalan UAV dalam mengikuti pola lintasan yang telah ditentukan.

Perbedaan lintasan paling terlihat pada bagian awal dan akhir lintasan, khususnya pada sisi kiri grafik. Bagian tersebut tidak sepenuhnya membentuk lingkaran karena merupakan transisi UAV dari fase pendekatan menuju *circular inspection* dan selanjutnya keluar dari lintasan untuk menjalankan fase kembali (*return navigation*). Pada jarak 3 m, deviasi relatif lebih terlihat karena radius yang kecil menyebabkan perubahan posisi terhadap pusat lingkaran menjadi lebih sensitif terhadap kesalahan navigasi. Sementara itu, pada radius yang lebih besar, lintasan aktual terlihat semakin mendekati lintasan referensi, meskipun tetap terdapat deviasi kecil sepanjang lintasan.



Gambar 4. Perbandingan Lintasan Ideal dan Lintasan Aktual pada Fase *Circular Inspection*.

Evaluasi Performa Variasi jarak

Evaluasi kinerja sistem dilakukan untuk mengetahui pengaruh variasi jarak inspeksi terhadap performa navigasi UAV. Pengujian dilakukan pada lima variasi target radius, yaitu 3, 5, 7, 10, dan 15 m, dengan tiga kali pengulangan pada setiap variasi. Parameter yang digunakan untuk mengevaluasi kinerja sistem meliputi mission completion time, altitude RMSE, radius RMSE, yaw RMSE, dan *return-to-home* (RTH) error. Nilai rata-rata dari setiap parameter pengujian disajikan pada Tabel 1.

$$L = 2\pi r \quad (6)$$

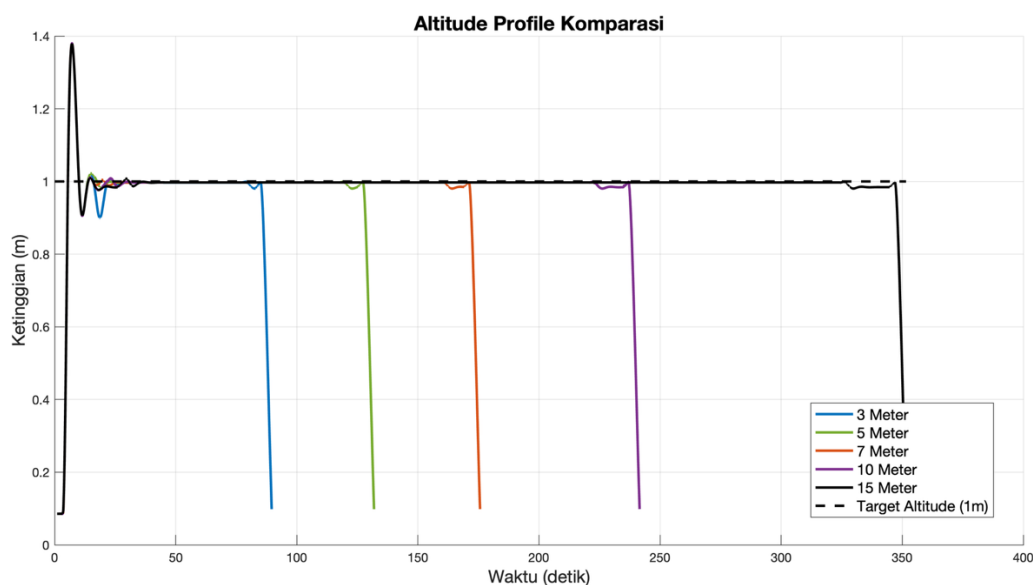
Berdasarkan Tabel 1, waktu penyelesaian misi meningkat seiring dengan bertambahnya jarak inspeksi. Waktu misi berturut-turut sebesar 88,61 s, 130,86 s, 174,64 s, 240,52 s, dan 350,54 s untuk jarak 3, 5, 7, 10, dan 15 m. Peningkatan ini menunjukkan bahwa radius inspeksi yang lebih besar menghasilkan lintasan yang lebih panjang sehingga UAV membutuhkan waktu lebih lama untuk menyelesaikan seluruh rangkaian misi. Secara khusus, peningkatan waktu misi dari 3 m ke 15 m mencapai sekitar empat kali lipat. Pola tersebut sesuai dengan karakteristik geometris lintasan melingkar, karena panjang lintasan ideal meningkat secara linear terhadap radius menurut hubungan persamaan 6. Dengan L merupakan panjang lintasan dan r merupakan radius lintasan. Oleh karena itu, peningkatan radius menyebabkan UAV menempuh lintasan yang lebih panjang dan pada akhirnya meningkatkan waktu penyelesaian misi. Hasil ini menunjukkan bahwa *mission completion time* merupakan parameter penting dalam mengevaluasi efisiensi operasional sistem ketika radius area inspeksi diperbesar.

Tabel 1. Rata-rata Performa Navigasi UAV pada Berbagai jarak.

Jarak (m)	Mission Time (s)	Altitude RMSE (m)	Radius RMSE (m)	Yaw RMSE (rad)	RTH Error (m)
3	88.61	0.2452	0.9442	0.3699	0.5300
5	130.86	0.2012	0.2481	0.2992	0.1615
7	174.64	0.1741	0.2208	0.2577	0.1891
10	240.52	0.1485	0.1996	0.2187	0.2154
15	350.54	0.1230	0.1830	0.1806	0.2703

Performa pengendalian ketinggian UAV dievaluasi menggunakan *altitude root mean square error* (RMSE). Pada seluruh skenario pengujian, ketinggian UAV ditetapkan sebesar 1 m di atas permukaan sebagai ketinggian referensi. Profil ketinggian UAV selama pelaksanaan misi pada berbagai variasi radius inspeksi ditunjukkan pada Gambar 5. Berdasarkan hasil pengujian, UAV mampu mempertahankan ketinggian di sekitar nilai referensi selama pelaksanaan misi. Nilai altitude RMSE menunjukkan kecenderungan menurun secara konsisten, yaitu dari 0.2452 m pada radius 3 m menjadi 0.1230 m pada radius 15 m. Penurunan nilai RMSE menunjukkan bahwa deviasi ketinggian UAV terhadap ketinggian referensi semakin kecil pada radius inspeksi yang lebih besar.

Secara keseluruhan, nilai altitude RMSE mengalami penurunan sebesar sekitar 49.8% dari radius 3 m hingga 15 m. Hasil tersebut menunjukkan bahwa sistem pengendalian ketinggian mampu mempertahankan kestabilan altitude pada seluruh variasi radius yang diuji. Meskipun demikian, penurunan nilai RMSE tersebut tidak secara langsung menunjukkan bahwa peningkatan radius inspeksi menyebabkan performa pengendalian *altitude* menjadi lebih baik. Perbedaan nilai *error* dapat dipengaruhi oleh beberapa faktor, seperti karakteristik dinamika UAV, durasi penerbangan, perubahan kondisi gerak selama mengikuti lintasan, serta respons sistem kontrol terhadap masing-masing konfigurasi lintasan. Dengan demikian, hasil ini menunjukkan bahwa sistem mampu mempertahankan ketinggian referensi sebesar 1 m secara konsisten pada seluruh skenario pengujian.



Gambar 5. Perbandingan ketinggian UAV selama pelaksanaan misi pada berbagai variasi jarak inspeksi.

Akurasi UAV dalam mempertahankan lintasan inspeksi melingkar dievaluasi menggunakan radius RMSE. Nilai radius RMSE diperoleh dari perbedaan antara radius aktual UAV terhadap radius target selama fase circular inspection. Hasil pengujian menunjukkan bahwa nilai radius RMSE menurun secara signifikan dari 0.9442 m pada radius 3 m menjadi 0.2481 m pada radius 5 m. Selanjutnya, nilai tersebut terus menurun menjadi 0.2208, 0.1996, dan 0.1830 m masing-masing pada radius 7, 10, dan 15 m.

Kecenderungan tersebut menunjukkan bahwa kemampuan UAV dalam mempertahankan radius lintasan relatif meningkat seiring dengan bertambahnya jarak inspeksi. Pada radius 3 m, nilai *error* yang relatif besar menunjukkan bahwa deviasi posisi UAV terhadap lintasan melingkar memiliki pengaruh yang lebih signifikan. Sebaliknya, pada radius 15 m, radius

RMSE hanya sebesar 0.1830 m. Dengan demikian, sistem mampu mempertahankan lintasan dengan deviasi radial yang relatif kecil meskipun radius inspeksi diperbesar.

Jika dibandingkan antara radius 3 m dan 15 m, radius RMSE mengalami penurunan sekitar 80.6%. Hasil ini menunjukkan bahwa peningkatan radius tidak selalu menyebabkan penurunan akurasi lintasan. Dalam konfigurasi eksperimen ini, justru diperoleh kecenderungan bahwa lintasan dengan radius yang lebih besar dapat dipertahankan dengan deviasi radial yang lebih rendah. Temuan tersebut menjadi salah satu hasil penting dalam evaluasi performa sistem navigasi berbasis variasi jarak.

Stabilitas orientasi UAV selama pelaksanaan misi dievaluasi menggunakan *yaw* RMSE. Berdasarkan Tabel 1, nilai *yaw* RMSE mengalami penurunan secara konsisten dari 0.3699 rad pada radius 3 m menjadi 0.1806 rad pada radius 15 m. Nilai tersebut menunjukkan adanya pengurangan deviasi orientasi UAV terhadap orientasi referensi ketika radius inspeksi diperbesar.

Secara kuantitatif, *yaw* RMSE mengalami penurunan sekitar 51.2% antara radius 3 m dan 15 m. Kecenderungan ini menunjukkan bahwa sistem mampu mempertahankan orientasi UAV secara relatif lebih stabil pada lintasan dengan radius yang lebih besar. Nilai *error* yang lebih rendah pada radius yang lebih besar juga konsisten dengan hasil radius RMSE, yang menunjukkan bahwa konfigurasi lintasan dengan radius lebih besar menghasilkan performa tracking yang relatif lebih baik pada eksperimen yang dilakukan.

Kemampuan UAV untuk kembali menuju titik awal setelah menyelesaikan misi dievaluasi menggunakan *return to home* (RTH) *error*. Berbeda dengan parameter *altitude* RMSE, *radius* RMSE, dan *yaw* RMSE yang menunjukkan kecenderungan menurun, nilai RTH *error* memiliki pola yang tidak monoton. Nilai RTH *error* masing-masing sebesar 0.5300, 0.1615, 0.1891, 0.2154, dan 0.2703 m untuk radius 3, 5, 7, 10, dan 15 m.

Error terkecil diperoleh pada radius 5 m, yaitu sebesar 0.1615 m, sedangkan *error* terbesar diperoleh pada radius 3 m, yaitu sebesar 0.5300 m. Setelah mencapai nilai minimum pada radius 5 m, RTH *error* mengalami peningkatan secara bertahap hingga mencapai 0.2703 m pada *radius* 15 m. Pola ini menunjukkan bahwa peningkatan radius inspeksi tidak memiliki hubungan linear terhadap akurasi *return to home*. Performa RTH kemungkinan dipengaruhi oleh akumulasi *error* posisi selama pelaksanaan misi, kondisi akhir lintasan inspeksi, serta respons sistem navigasi ketika melakukan transisi dari fase inspeksi menuju fase kembali ke titik awal.

Secara keseluruhan, nilai RTH *error* masih berada di bawah 0.6 m pada seluruh konfigurasi pengujian. Hal tersebut menunjukkan bahwa UAV tetap mampu kembali

mendekati posisi *home* setelah menyelesaikan misi inspeksi pada seluruh variasi radius yang diuji.

5. KESIMPULAN

Penelitian ini telah mengevaluasi kinerja navigasi UAV otonom dalam menyelesaikan misi inspeksi melingkar berbasis *Finite State Machine* (FSM) pada lingkungan simulasi Webots. Pengujian dilakukan pada lima variasi radius, yaitu 3, 5, 7, 10, dan 15 m, dengan tiga kali pengulangan pada setiap konfigurasi. Hasil pengujian menunjukkan bahwa sistem mampu menyelesaikan seluruh skenario misi dengan baik. *Mission completion time* meningkat dari 88.61 s pada radius 3 m menjadi 350.54 s pada radius 15 m, yang menunjukkan adanya konsekuensi peningkatan cakupan area terhadap waktu penyelesaian misi.

Dari aspek akurasi navigasi, sistem menunjukkan performa yang relatif stabil pada seluruh variasi radius. *Altitude* RMSE menurun dari 0.2452 m menjadi 0.1230 m, *yaw* RMSE menurun dari 0.3699 rad menjadi 0.1806 rad, sedangkan radius RMSE menurun dari 0.9442 m menjadi 0.1830 m pada peningkatan radius 3 hingga 15 m. Sementara itu, *RTH error* menunjukkan pola non monotonik dengan nilai terendah sebesar 0.1615 m pada radius 5 m dan tertinggi sebesar 0.5300 m pada radius 3 m. Hasil tersebut menunjukkan bahwa sistem FSM mampu mempertahankan ketinggian, orientasi, dan lintasan melingkar dengan deviasi yang relatif kecil pada berbagai variasi radius, meskipun peningkatan radius menyebabkan waktu operasi menjadi lebih panjang.

Penelitian selanjutnya disarankan untuk meningkatkan jumlah pengulangan dan menerapkan analisis statistik guna memperoleh evaluasi performa yang lebih kuat. Pengujian juga dapat dikembangkan dengan variasi kecepatan, ketinggian, parameter yaw, serta penambahan gangguan seperti angin dan *noise* sensor untuk menguji *robustness* sistem. Selain itu, perbandingan dengan metode navigasi atau pengendalian lainnya dapat dilakukan untuk memperoleh konfigurasi yang lebih optimal. Sebagai tahap lanjutan, sistem berbasis simulasi Webots perlu divalidasi pada UAV fisik di lingkungan nyata untuk mengevaluasi pengaruh faktor-faktor aktual seperti turbulensi, ketidakpastian sensor, dan dinamika UAV terhadap performa navigasi.

DAFTAR REFERENSI

Artigas, M. I., Rodrigues, R. T., Vanderseypen, L., & Bruyninckx, H. (2024). Software patterns and data structures for the runtime coordination of robots, with a focus on real-time execution performance. *Frontiers in Robotics and AI*, 11, 1363041. <http://dx.doi.org/10.3389/frobt.2024.1363041>.

- Baumann, C., & Martinoli, A. (2022). Spatial microscopic modeling of collective movements in multi-robot systems: Design choices and calibration. *Frontiers in Robotics and AI*, 9, 961053. <http://dx.doi.org/10.3389/frobt.2022.961053>.
- Baumann, C., Perolini, J., Tourki, E., & Martinoli, A. (2024). Hybrid flock–formation control algorithms. In *Lecture Notes in Computer Science*, 523–536. http://dx.doi.org/10.1007/978-3-031-51497-5_37.
- Brasiel, H. C., & Lima, D. A. (2024). Swarm robotics surveillance control with Ant Cellular Automata Model in the Cerrado Biome for preserving biodiversity. *Journal on Interactive Systems*, 15(1), 375–387. <http://dx.doi.org/10.5753/jis.2024.3796>.
- Cheng, C., Duan, S., He, H., Li, X., & Chen, Y. (2022). A generalized robot navigation analysis platform (RoNAP) with visual results using multiple navigation algorithms. *Sensors*, 22(23), 9036. <http://dx.doi.org/10.3390/s22239036>.
- Collins, J., Chand, S., Vanderkop, A., & Howard, D. (2021). A review of physics simulators for robotic applications. *IEEE Access*, 9, 51416–51431. <http://dx.doi.org/10.1109/ACCESS.2021.3068769>.
- Correll, N., & Martinoli, A. (2009). Multirobot inspection of industrial machinery. *IEEE Robotics & Automation Magazine*, 16(1), 103–112. <http://dx.doi.org/10.1109/MRA.2008.931633>.
- Emmi, L., Paredes-Madrid, L., Ribeiro, Á., Pájaros, G., & Santos, P. G. de. (2013). Fleets of robots for precision agriculture: A simulation environment. *Industrial Robot: The International Journal of Robotics Research and Application*, 40(1), 41–58. <http://dx.doi.org/10.1108/01439911311294246>.
- Hejase, B., & Özgüner, Ü. (2022). Physics-based simulation and automation of a load-haul-dump operation for an articulated dump truck. *Vehicles*, 4(1), 167–181. <http://dx.doi.org/10.3390/vehicles4010011>.
- Hussain, R., Zielińska, T., & Hexel, R. (2019). Finite state automaton based control system for walking machines. *International Journal of Advanced Robotic Systems*, 16(3), 1–10. <http://dx.doi.org/10.1177/1729881419853182>.
- Iskandar, A., & Kovács, B. (2024). Investigating the impact of curriculum learning on reinforcement learning for improved navigational capabilities in mobile robots. *Inteligencia Artificial*, 27(73), 163–176. <http://dx.doi.org/10.4114/intartif.vol27iss73pp163-176>.
- Mañas-Álvarez, F.-J., Guinaldo, M., Dormido, R., & Dormido-Canto, S. (2023). Scalability of cyber-physical systems with real and virtual robots in ROS 2. *Sensors*, 23(13), 6073. <http://dx.doi.org/10.3390/s23136073>.
- Moreno, I. J., Ouardani, D., Chaparro-Arce, D., & Cardenas, A. (2023). Real-time hardware-in-the-loop emulation of path tracking in low-cost agricultural robots. *Vehicles*, 5(3), 894–913. <http://dx.doi.org/10.3390/vehicles5030049>.
- Navarro, I., Pugh, J., Martinoli, A., & Matía, F. (2009). A distributed scalable approach to formation control in multi-robot systems. In *Lecture Notes in Computer Science*, 203–214. http://dx.doi.org/10.1007/978-3-642-00644-9_18.
- Othman, K. M., & Rad, A. B. (2020). Sequential localizing and mapping: A navigation strategy via enhanced subsumption architecture. *Sensors*, 20(17), 4815. <http://dx.doi.org/10.3390/s20174815>.

- Plaza, J. M., Perdices, E., García-Pérez, L., & Fernández-Conde, J. (2020). A ROS-based open tool for intelligent robotics education. *Applied Sciences*, 10(21), 7419. <http://dx.doi.org/10.3390/app10217419>.
- Prorok, A., Correll, N., & Martinoli, A. (2011). Multi-level spatial modeling for stochastic distributed robotic systems. *The International Journal of Robotics Research*, 30(5), 574–589. <http://dx.doi.org/10.1177/0278364910399521>.
- Rasouli, S., Dautenhahn, K., & Nehaniv, C. L. (2024). Simulation of a bio-inspired flocking-based aggregation behaviour in swarm robotics. *Biomimetics*, 9(11), 668. <http://dx.doi.org/10.3390/biomimetics9110668>.
- Rossander, M., & Lideskog, H. (2023). Design and implementation of a control system for an autonomous reforestation machine using finite state machines. *Forests*, 14(7), 1340. <http://dx.doi.org/10.3390/f14071340>.
- Trihinas, D., Agathocleous, M., Avogian, K., & Katakis, I. (2021). FlockAI: A testing suite for ML-driven drone applications. *Future Internet*, 13(12), 317. <http://dx.doi.org/10.3390/fi13120317>.
- Uğurlu, H. İ., Pham, H. X., & Kayacan, E. (2022). Sim-to-real deep reinforcement learning for safe end-to-end planning of aerial robots. *Robotics*, 11(5), 109. <http://dx.doi.org/10.3390/robotics11050109>.
- Wang, Y., Chu, D., Li, H., & Lu, L. (2023). Formation control for connected and autonomous vehicles based on distributed consensus embedded with risk potential field. *IEEE Access*, 11, 45618–45631. <http://dx.doi.org/10.1109/ACCESS.2023.3273610>.