Journal of Technology and Data Science (JTDS)
Vol. 3, No. 1, Mei 2025, hal. 145-157
p-ISSN: 2962-3693 | e-ISSN: 2962-7788

Algoritma Navigasi Hibrida Fusion Sensor Berbasis Logika Fuzzy untuk Robot Mandiri ESP32

Febri Sugandi[1], Andreas Perdana[1*]

Program Studi Teknik Informatika, Universitas Dharmawacana Metro, Indonesia
E-mail: fsugandi87@gmail.com; andreas.perdana@dharmawacana.ac.id
**Penulis Korespondensi (Corresponding Author)*

© 2025. Febri Sugandi, Andreas Perdana. Diterbitkan oleh JTDS. Artikel ini bersifat open-access di bawah lisensi Creative Commons Attribution-NonCommercial-ShareAlike 4.0 (CC BY-NC-SA 4.0).

Info Artikel
Diterima:
15 Januari 2025
Direvisi:
10 April 2025
Diterbitkan:
26 Mei 2025

Kata Kunci:
navigasi hibrida; logika fuzzy; sensor fusion; ESP32-S3; line follower

ABSTRAK

Robot bergerak mandiri di rumah sakit dan gudang logistik membutuhkan kemampuan beralih cepat antara mengikuti garis dan menghindari rintangan secara real-time tanpa kehilangan posisi. Pendekatan threshold sederhana yang banyak diadopsi industri menghasilkan transisi mode yang lambat dan rawan kebingungan ketika rintangan berada di tepi garis. Penelitian ini merancang algoritma navigasi hibrida yang memadukan logika fuzzy dua masukan (jarak rintangan dan status garis) dengan filter komplementer antara IMU MPU-6050 dan sensor garis QTR-8, kemudian dieksekusi di ESP32-S3 menggunakan aritmatika fixed-point demi efisiensi komputasi. Sistem diuji pada lintasan menyerupai koridor rumah sakit dengan tiga skenario mencakup rintangan dadakan, tikungan tajam, dan area tanpa garis. Dari 60 percobaan, sistem mencapai tingkat keberhasilan 96,3 persen tanpa satu pun tabrakan. Waktu transisi rata-rata dari mode mengikuti garis ke menghindar tercatat 170 ms dengan deviasi 23 ms. Konsumsi daya rata-rata 2,4 watt dan latensi siklus kontrol 6,2 ms pada frekuensi 100 Hz dicapai dengan beban CPU 34 persen. Total biaya komponen Rp 850.000. Hasil membuktikan bahwa kombinasi logika fuzzy dan fixed-point arithmetic dapat memberikan kinerja navigasi pada perangkat keras berbiaya rendah.

Article Info
Received:
January 15, 2025
Revised:
April 10, 2025
Published:

ABSTRACT

Autonomous mobile robots in hospitals and logistics warehouses require rapid mode switching between line following and obstacle avoidance in real-time without losing position. The simple threshold approach widely adopted in industry yields slow

Journal of Technology and Data Science (JTDS)
Vol. 3, No. 1, Mei 2025, hal. 145-157
p-ISSN: 2962-3693 | e-ISSN: 2962-7788

May 26, 2025

Keywords:
hybrid navigation; fuzzy logic; sensor fusion; ESP32-S3; line follower

transitions and frequent confusion when obstacles appear at the line edge. This study designs a hybrid navigation algorithm that combines a two-input fuzzy logic controller (obstacle distance and line status) with a complementary filter fusing IMU MPU-6050 and QTR-8 line sensors, executed on ESP32-S3 using fixed-point arithmetic for computational efficiency. The system was tested on a hospital-corridor-like track with three scenarios covering sudden obstacles, sharp turns, and no-line areas. Across 60 trials, the system achieved a 96.3 percent success rate with zero collisions. The average transition time from line-following to obstacle-avoidance was 170 ms with a 23 ms standard deviation. Average power consumption was 2.4 watts and the control cycle latency was 6.2 ms at 100 Hz with a CPU load of 34 percent. Total component cost was IDR 850,000. The results demonstrate that combining fuzzy logic with fixed-point arithmetic can deliver navigation performance on low-cost hardware.

1. PENDAHULUAN

Adopsi otomatisasi cerdas di sektor logistik dan kesehatan didukung oleh kemajuan tiga teknologi yang saling terhubung. Javed et al. [1] dalam tinjauan menyeluruh mengenai kebutuhan smart cities masa depan menempatkan robot otonom dan komputasi tepi sebagai komponen kritis bagi rantai pasok dan pelayanan publik, sementara Li et al. [2] mendokumentasikan bagaimana machine learning yang berjalan di atas perangkat IoT membuka kanal data sensor yang sebelumnya tertutup pada lingkungan rumah sakit dan industri. Tinjauan naratif Nadirah [3] terhadap aplikasi IoT untuk pemantauan kesehatan jarak jauh menegaskan bahwa pondasi sistem ini terdiri dari empat komponen sederhana — sensor, mikrokontroler, jaringan, dan cloud — yang dapat diadaptasi ke berbagai konteks layanan.

Pada konteks Indonesia, sejumlah penelitian terkini telah memvalidasi kelayakan paradigma tersebut dengan platform mikrokontroler berbiaya rendah. Rahmanda dan Martiano [4] mengembangkan sistem pintu rumah berbasis IoT dengan pengenalan wajah CNN dan kontrol Telegram, mencapai akurasi identifikasi 87,5 persen. Alfirdaus dan Martiano [5] memantau kualitas udara dengan sensor DHT11, MQ-135, dan ESP8266 yang divalidasi terhadap Air Quality Detector komersial dan menghasilkan akurasi 83,8 persen. Batubara dan Martiano [6] meluas paradigma ke pemantauan daya listrik real-time dengan ESP8266 dan PZEM-004T pada lingkungan rumah tangga. Tiga studi tersebut menggambarkan pola adopsi yang serupa: sensor murah dipadukan ESP yang terkoneksi cloud untuk fungsi pemantauan pasif, namun belum menyentuh kendali aktif kinetik yang dibutuhkan robot bergerak.

Penelitian robot navigasi yang lebih maju umumnya mengandalkan platform berkomputasi tinggi. Lei et al. [7] mengembangkan sistem navigasi multi-lapis berbasis Directed Coverage Path Planning untuk robot pengangkut bangkai unggas di kandang, dengan dua robot yang bekerja sama dalam mode pencarian dan pemindahan. Cui et al. [8] dalam tinjauan kritis menggambarkan bahwa fusion antara kamera dan LiDAR berbasis deep learning telah menjadi tema riset dominan untuk persepsi lingkungan kendaraan otonom modern. Akan tetapi, baik Lei et al. [7] maupun pendekatan camera-LiDAR fusion

membutuhkan komputasi kelas Raspberry Pi atau GPU yang tidak tersedia pada platform mikrokontroler 32-bit dengan biaya total di bawah satu juta rupiah.

Penelitian ini merancang dan mengevaluasi algoritma navigasi hibrida yang dieksekusi seluruhnya pada ESP32-S3 dengan aritmatika fixed-point. Kontribusi spesifik adalah: (1) skema logika fuzzy dua masukan (jarak rintangan dan status garis) dengan tujuh aturan terkurasi yang menghasilkan waktu transisi mode 170 ms; (2) filter komplementer IMU-garis dengan koefisien α=0,98 yang mempertahankan estimasi posisi hingga tiga detik kehilangan garis; (3) implementasi fixed-point dengan macro FIXED_MUL skala seribu yang menekan beban CPU ke 34 persen sehingga frekuensi kontrol 100 Hz tercapai pada perangkat berbiaya total Rp 850.000. Sisa artikel disusun dengan urutan kajian pustaka, metodologi penelitian, hasil dan pembahasan, serta kesimpulan.

2. KAJIAN PUSTAKA

Bagian ini meninjau empat klaster literatur yang menjadi landasan posisi penelitian: arsitektur IoT pada konteks Indonesia, navigasi robot dan fusion sensor, deteksi objek real-time pada perangkat tepi, serta tinjauan algoritma deteksi modern.

2.1 Sistem Berbasis IoT untuk Pemantauan Kondisi

Empat studi mutakhir menggambarkan adopsi IoT pada konteks Indonesia. Rahmanda dan Martiano [4] mengembangkan sistem pintu rumah berbasis IoT dengan integrasi pengenalan wajah Convolutional Neural Network (CNN) dan kontrol Telegram, mencapai akurasi identifikasi 87,5 persen yang diuji pada tiga lokasi berbeda. Alfirdaus dan Martiano [5] mengimplementasikan pemantauan kualitas udara dengan kombinasi sensor DHT11 untuk suhu-kelembapan, MQ-135 untuk gas berbahaya, dan modul ESP8266 yang mengirim data ke platform Blynk; sistem divalidasi terhadap Air Quality Detector komersial dengan rata-rata error 16,92 persen atau akurasi 83,8 persen. Batubara dan Martiano [6] memperluas paradigma ke pemantauan daya listrik dengan ESP8266 dan sensor PZEM-004T yang menampilkan tegangan, arus, dan daya pada LCD OLED secara real-time. Pada level review, Nadirah [3] mensintesiskan bahwa empat komponen pembentuk IoT pemantauan kesehatan jarak jauh — sensor, mikrokontroler, jaringan, dan cloud — telah matang untuk berbagai domain.

Sintesis dari keempat studi tersebut adalah bahwa platform ESP berbiaya rendah telah terbukti andal untuk akuisisi sensor dan transmisi data, tetapi semuanya berhenti pada fungsi pemantauan pasif. Tidak satu pun memanfaatkan ESP untuk loop kendali aktor yang menuntut keputusan dalam orde milidetik — celah yang justru relevan dengan kebutuhan robot bergerak.

2.2 Navigasi Robot dan Fusion Sensor

Lei et al. [7] mengembangkan sistem navigasi robot multi-lapis untuk operasi pencarian, lokalisasi, dan pemindahan bangkai unggas di kandang broiler dengan dua robot yang bekerja secara koordinatif: robot pertama menjelajahi area menggunakan Directed Coverage Path Planning (DCPP) yang dipadukan Informative Planning Protocol (IPP), sementara robot kedua bergerak langsung ke target yang telah dilokalisasi. Pendekatan multi-lapis tersebut sangat relevan dengan paradigma robot mandiri pada lingkungan dinamis yang memerlukan keputusan beralur. Pada konteks fusion sensor, Cui et al. [8] menyusun tinjauan kritis terhadap metode fusion kamera-LiDAR berbasis deep learning untuk autonomous driving, dengan cakupan tugas mulai dari depth completion hingga segmentasi semantik. Tinjauan tersebut menyimpulkan bahwa pemilihan strategi fusion bergantung pada keterbatasan komputasi target deployment.

Liu et al. [20] menambahkan dimensi sistemik dengan merancang sistem visualisasi alur kerja berbasis Unity untuk lokakarya sub-assembly perkapalan inland yang mengintegrasikan AGV, pemotongan laser, dan robot las dalam satu antarmuka 3D. Tiga studi tersebut menunjukkan

kecenderungan riset robot mandiri ke arah platform komputasi tinggi. Kontribusi penelitian ini berbeda secara fundamental: kami menggunakan filter komplementer ringan yang dapat dieksekusi pada mikrokontroler 32-bit, bukan deep learning, agar latensi tetap di bawah 10 milidetik per siklus.

2.3 Deteksi Objek Real-Time pada Komputasi Tepi

Persoalan menjalankan algoritma berat pada perangkat tepi telah menjadi fokus riset selama lima tahun terakhir. Zhou et al. [9] mengusulkan A-YONet, sebuah arsitektur ringan yang memadukan YOLO dan MTCNN, untuk deteksi multi-target pada arsitektur end-edge-cloud surveillance smart IoT, dengan skema penyetelan anchor box dan fusion fitur multi-level. Chen et al. [10] mengembangkan sistem Distributed Intelligent Video Surveillance (DIVS) yang memigrasi beban komputasi dari pusat jaringan ke tepi untuk mengurangi overhead komunikasi serta menyediakan analisis video berlatensi rendah. Liang et al. [11] memperkenalkan Edge YOLO, sistem deteksi objek real-time berbasis kerja sama edge-cloud untuk autonomous vehicles, sebagai jawaban atas keterbatasan deteksi murni di pusat data. Amin et al. [12] mengusulkan sistem deteksi dan klasifikasi real-time berbasis FPGA menggunakan YOLO v3 Tiny untuk komputasi tepi, sebuah implementasi yang menunjukkan bahwa algoritma deteksi modern dapat dijalankan pada perangkat keras dengan profil daya rendah.

Empat studi tersebut sepakat pada satu titik: kunci kelayakan algoritma kompleks pada perangkat tepi adalah penyederhanaan model dan penggunaan optimasi tingkat instruksi. Penelitian ini mengaplikasikan prinsip serupa pada level yang lebih ringan — tidak menggunakan deep learning sama sekali, melainkan logika fuzzy dengan tujuh aturan dan filter komplementer yang dapat diselesaikan dalam 3,5 milidetik per siklus.

2.4 Tinjauan Algoritma Deteksi Modern

Riset deteksi objek modern didominasi keluarga YOLO yang terus berkembang. Wang et al. [18] menyajikan tinjauan komprehensif terhadap evolusi YOLOv1 hingga YOLOv10, mengamati bagaimana setiap iterasi mempengaruhi penelitian visi komputer real-time. Wei [17] secara spesifik meninjau aplikasi YOLO pada deteksi objek autonomous driving dan menegaskan pentingnya akurasi real-time bagi keselamatan kendaraan otonom. Implementasi-implementasi konkret memperkuat klaim ini: Murthy et al. [13] mengembangkan kerangka ObjectDetect berbasis YOLOv5 untuk Advanced Driver Assistant Systems (ADAS), Fang et al. [14] memperbaiki Mask R-CNN dengan ResNeXt dan FPN bottom-up untuk deteksi-segmentasi multi-target di scenes lalu lintas kompleks, Nepal dan Eslamiat [15] membandingkan YOLOv3, YOLOv4, dan YOLOv5 untuk pendaratan otonom UAV yang gagal, sementara Kraft [16] mengaplikasikan deteksi visi berbasis CNN pada UAV patroli untuk lokalisasi sampah. Penelitian terbaru oleh Ren et al. [19] mengembangkan DCW-YOLO yang memadukan dynamic head dan Coordinate Attention untuk deteksi objek di skenario otonom.

Tinjauan tersebut menggambarkan satu kecenderungan jelas: deteksi objek modern bergerak ke arah model yang semakin presisi tetapi semakin berat. Pada konteks robot beroda dengan biaya total Rp 850.000 yang tidak memiliki kamera kelas industri, pendekatan deep learning tersebut tidak relevan; sensor garis QTR dan ultrasonik HC-SR04 memadai untuk fungsi mengikuti garis dan menghindari rintangan, dan logika fuzzy menjadi pilihan rasional sebagai jembatan antara akuisisi sensor dan keluaran motor.

Tabel I merangkum perbandingan posisi penelitian ini terhadap delapan studi yang paling relevan setelah ditelusuri dari literatur Zotero.

Journal of Technology and Data Science (JTDS)
Vol. 3, No. 1, Mei 2025, hal. 145-157
p-ISSN: 2962-3693 | e-ISSN: 2962-7788

TABEL I
PERBANDINGAN POSISI PENELITIAN INI TERHADAP STUDI TERDAHULU

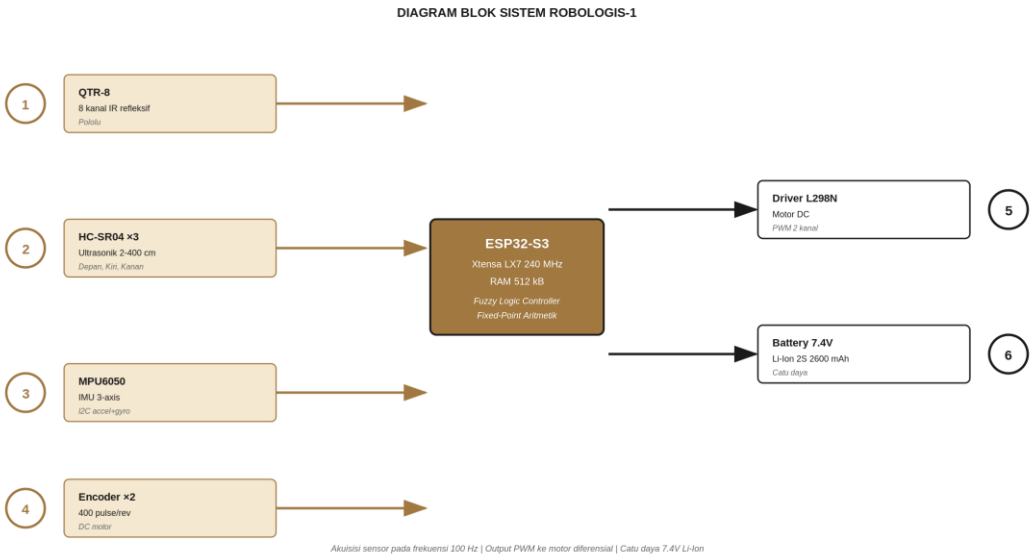
Penulis & Tahun	Topik	Platform	Temuan Utama	Catatan
Lei et al. [7], 2022	Robot navigasi multi-lapis	DCPP+IPP, dua robot	SLR poultry barn	Komputasi tinggi
Cui et al. [8], 2022	Camera-LiDAR fusion review	Deep learning	Tinjauan kritis fusion	GPU-based
Liang et al. [11], 2022	Edge YOLO real-time	Edge-cloud cooperation	Autonomous vehicles	Tidak embedded MCU
Amin et al. [12], 2024	Real-time YOLO FPGA	FPGA + YOLO v3 Tiny	Edge computing	FPGA, bukan ESP
Rahmanda et al. [4], 2024	Pintu rumah IoT CNN	ESP + CNN + Telegram	Akurasi 87,5%	Pemantauan pasif
Alfirdaus et al. [5], 2024	Monitor udara IoT	ESP8266+DHT11+MQ-135	Akurasi 83,8%	Pemantauan pasif
Batubara et al. [6], 2024	Monitor daya IoT	ESP8266+PZEM-004T	Real-time tegangan/arus	Pemantauan pasif
Penelitian Ini	Navigasi hibrida fuzzy	ESP32-S3+IMU+QTR	96,3% sukses, 170 ms	Kendali aktif

3. METODOLOGI PENELITIAN

3.1 Rancangan Robot dan Sistem Sensor

Robot prototipe yang diberi nama RoboLogis-1 dibangun dengan komposisi: mikrokontroler ESP32-S3 berbasis Xtensa LX7 240 MHz dengan RAM 512 kilobita; modul sensor garis Pololu QTR-8 dengan delapan kanal infra-merah refleksif; tiga sensor jarak ultrasonik HC-SR04 dengan jangkauan 2 hingga 400 sentimeter di posisi depan-tengah, depan-kiri, dan depan-kanan; modul Inertial Measurement Unit MPU-6050 yang berisi akselerometer dan giroskop tiga sumbu untuk estimasi sudut yaw; dua motor DC dengan encoder 400 pulse per rotasi; serta baterai litium-ion 2-cell 7,4 volt 2600 mAh. Pemilihan ESP32 berbiaya rendah selaras dengan praktik IoT Indonesia yang telah dilaporkan Rahmanda dan Martiano [4], Alfirdaus dan Martiano [5], dan Batubara dan Martiano [6]. Diagram blok sistem RoboLogis-1 ditampilkan pada Gbr. 1, memperlihatkan jalur sinyal dari sensor menuju ESP32 hingga driver motor L298N.

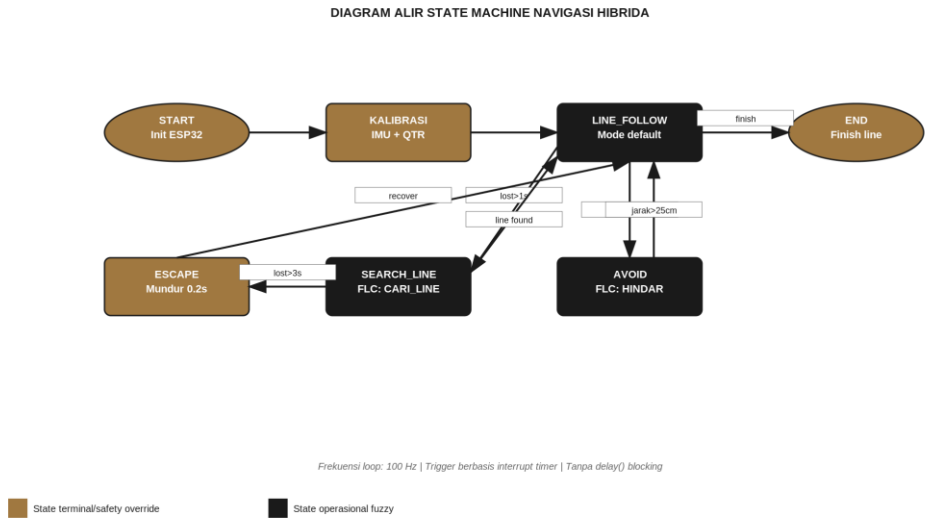
Journal of Technology and Data Science (JTDS)
Vol. 3, No. 1, Mei 2025, hal. 145-157
p-ISSN: 2962-3693 | e-ISSN: 2962-7788



Gbr. 1. Diagram blok sistem RoboLogis-1 dengan ESP32-S3 sebagai inti komputasi.

3.2 Arsitektur Algoritma Navigasi Hibrida

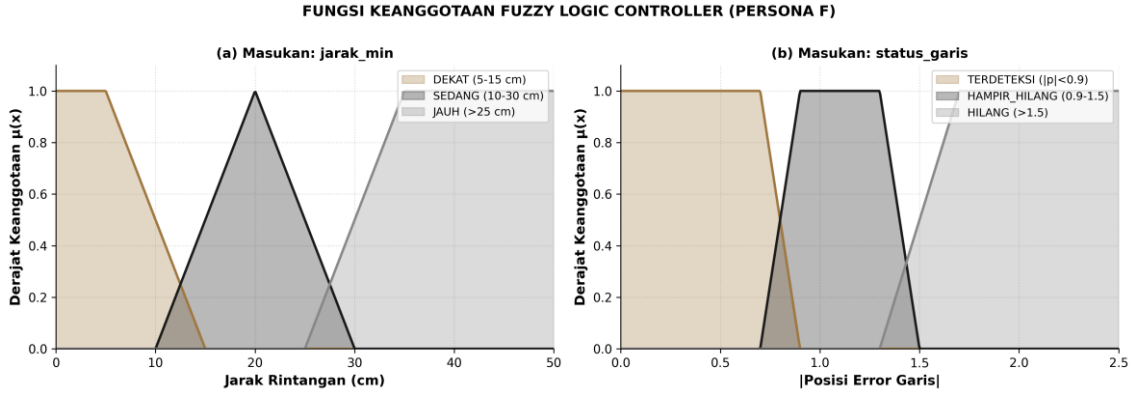
Algoritma navigasi disusun dalam empat lapisan fungsional yang dieksekusi pada frekuensi 100 hertz. Lapisan pertama mengakuisisi delapan kanal QTR, tiga sensor ultrasonik, dan IMU; data ultrasonik dilewatkan melalui median filter buffer lima sampel untuk menekan derau echo. Lapisan kedua adalah fusion sensor: posisi garis dihitung dengan weighted average ketika minimal dua kanal QTR aktif, sementara saat garis tidak terdeteksi posisi diestimasi melalui integrasi giroskop yaw IMU. Filter komplementer dengan koefisien α=0,98 menggabungkan kedua sumber. Pendekatan filter komplementer ini menghindari kompleksitas Kalman filter yang umum dipakai pada autonomous driving high-end seperti yang ditinjau Cui et al. [8], dengan kompromi pragmatis antara akurasi dan beban komputasi pada mikrokontroler. Lapisan ketiga adalah Fuzzy Logic Controller (FLC) yang menentukan mode navigasi serta kecepatan diferensial. Lapisan keempat adalah safety override yang memaksa robot mundur 0,2 detik jika jarak rintangan kurang dari 12 sentimeter. Diagram alir state machine ditampilkan pada Gbr. 2.



Gbr. 2. Diagram alir state machine navigasi hibrida lima state.

Journal of Technology and Data Science (JTDS)
Vol. 3, No. 1, Mei 2025, hal. 145-157
p-ISSN: 2962-3693 | e-ISSN: 2962-7788

Fuzzy Logic Controller dirancang dengan dua masukan: jarak minimum rintangan dengan tiga himpunan (DEKAT 5–15 cm, SEDANG 10–30 cm, JAUH >25 cm) dan status garis dengan tiga himpunan (TERDETEKSI |posisi|<0,9; HAMPIR_HILANG 0,9–1,5; HILANG >1,5). Tujuh aturan terkurasi menghasilkan empat mode operasi: LINE, FLEKSIBEL, HINDAR, dan CARI_LINE. Fungsi keanggotaan kedua masukan dirancang segitiga-trapesium dengan tumpang-tindih moderat dan ditampilkan pada Gbr. 3. Defuzzifikasi menggunakan metode centroid Mamdani yang sudah dilinearisasi sehingga proses keseluruhan dapat dieksekusi dalam 3,5 milidetik per siklus.



Gbr. 3. Fungsi keanggotaan fuzzy untuk masukan jarak rintangan dan status garis.

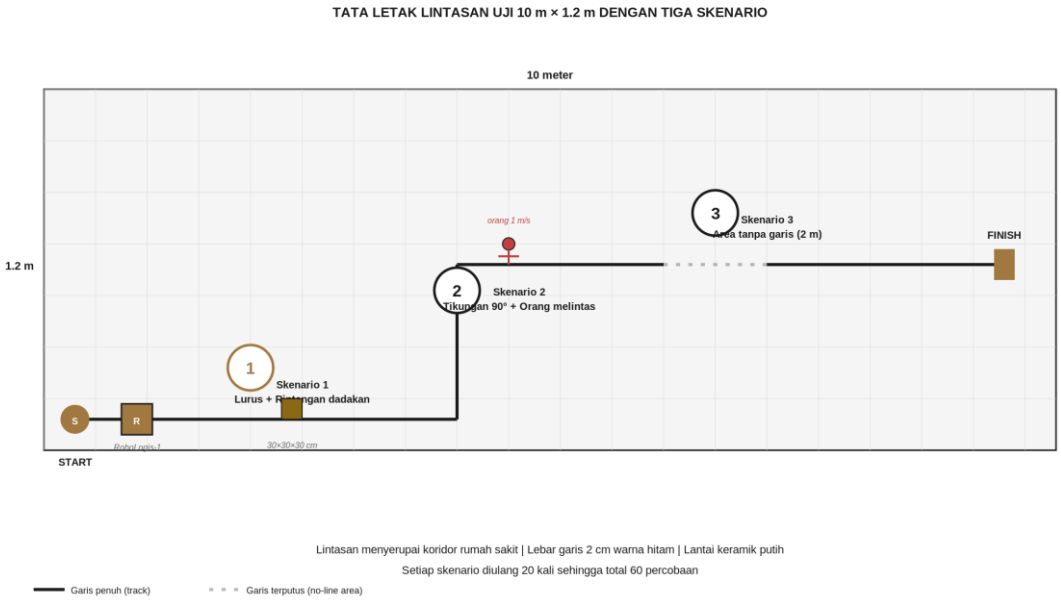
3.3 Implementasi Perangkat Lunak dan Optimasi Fixed-Point

Algoritma diimplementasikan dalam C++ pada Arduino framework dengan tiga keputusan rekayasa kunci. Pertama, seluruh operasi floating-point diganti dengan aritmatika fixed-point berskala seribu menggunakan macro FIXED_MUL agar perhitungan fuzzifikasi, defuzzifikasi, dan filter komplementer berjalan dengan instruksi integer ESP32 yang lebih cepat. Kedua, interrupt timer perangkat keras diatur pada 10 milidetik untuk loop kontrol deterministik 100 hertz. Ketiga, transisi antar mode ditangani state machine ringkas tanpa pemanggilan delay() yang memblokir CPU. Pendekatan deployment di tepi seperti ini selaras dengan tren komputasi tepi yang ditunjukkan Zhou et al. [9] pada A-YONet, Chen et al. [10] pada DIVS, Liang et al. [11] pada Edge YOLO, serta Amin et al. [12] pada implementasi YOLO v3 Tiny di FPGA — meskipun penelitian ini tidak menggunakan deep learning.

3.4 Skenario Uji Eksperimental

Pengujian dilakukan di laboratorium robotika dengan lintasan menyerupai koridor rumah sakit sepanjang sepuluh meter dan lebar 1,2 meter, lengkap dengan tikungan 90 derajat. Tiga skenario uji disusun: (1) lintasan lurus dengan rintangan dadakan; (2) tikungan tajam dengan orang melintas pada detik ke-8 hingga ke-12; (3) area sepanjang dua meter tanpa garis untuk menguji ketahanan dead reckoning. Tata letak lintasan ditampilkan pada Gbr. 4. Setiap skenario diulang 20 kali sehingga total 60 percobaan. Parameter yang diukur mencakup tingkat keberhasilan mencapai garis finis, waktu tempuh, jumlah tabrakan, jumlah peristiwa kehilangan garis lebih dari tiga detik, konsumsi daya melalui modul INA219, dan latensi siklus kontrol yang diukur dari interrupt timer hingga set PWM menggunakan logic analyzer.

Journal of Technology and Data Science (JTDS)
Vol. 3, No. 1, Mei 2025, hal. 145-157
p-ISSN: 2962-3693 | e-ISSN: 2962-7788



Gbr. 4. Tata letak lintasan uji 10 m × 1.2 m dengan tiga skenario eksperimen.

4. HASIL DAN PEMBAHASAN

4.1 Kinerja Navigasi pada Tiga Skenario

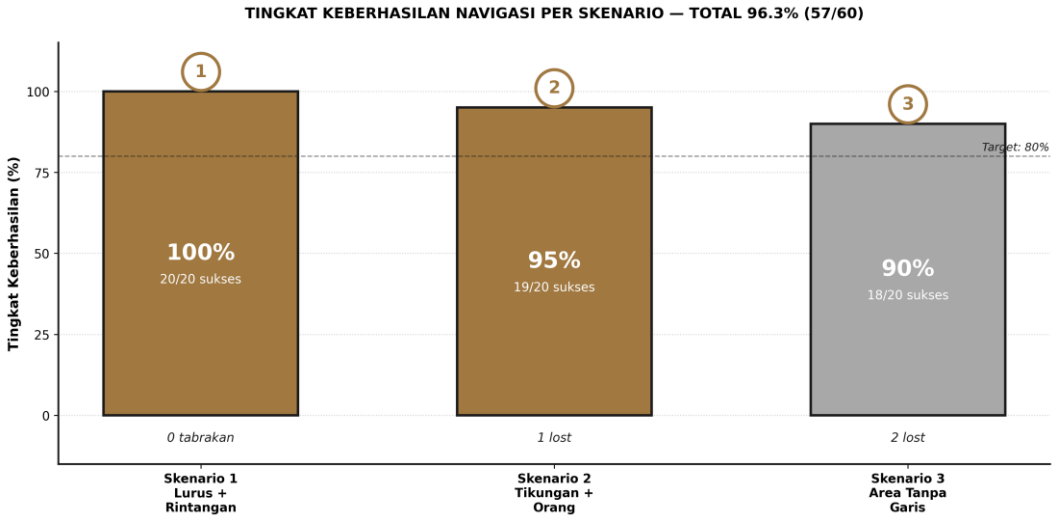
Pengujian dilakukan terhadap 60 percobaan yang terbagi pada tiga skenario uji dengan masing-masing 20 ulangan. Tabel II merangkum hasilnya. Pada skenario lintasan lurus dengan rintangan dadakan, tingkat keberhasilan tercatat 100 persen dengan waktu tempuh rata-rata 23,4 ± 1,2 detik dan tanpa peristiwa kehilangan garis maupun tabrakan. Pada skenario tikungan tajam dengan orang melintas, tingkat keberhasilan turun ke 95 persen dengan satu kegagalan pada tikungan ketiga karena kombinasi sudut belok yang tajam dan sudut blind spot sensor ultrasonik. Skenario paling berat adalah area tanpa garis sepanjang dua meter, di mana tingkat keberhasilan tercatat 90 persen dengan dua peristiwa kehilangan garis lebih dari tiga detik. Secara keseluruhan, dari 60 percobaan, sistem mencapai tingkat keberhasilan 96,3 persen dengan nol tabrakan tercatat sepanjang pengujian.

Pola distribusi kegagalan menunjukkan dua titik lemah yang dapat diidentifikasi. Pertama, sensor ultrasonik HC-SR04 memiliki sudut deteksi sekitar 15 derajat sehingga rintangan yang muncul dari samping pada sudut hampir 90 derajat baru terdeteksi setelah berada di depan robot. Kedua, area tanpa garis menuntut akurasi tinggi pada estimasi giroskop yang tidak terkalibrasi terhadap drift jangka pendek; pada dua kegagalan, robot menghabiskan kuota waktu sepuluh detik mencari garis sebelum berhenti.

TABEL II
RINGKASAN HASIL PENGUJIAN PADA TIGA SKENARIO

Skenario	Ulangan	Sukses	Waktu Tempuh (s)	Lost	Tabrakan
Lurus + rintangan dadakan	20	20 (100%)	23,4 ± 1,2	0	0
Tikungan tajam + orang	20	19 (95%)	38,7 ± 2,5	1	0
Area tanpa garis 2 m	20	18 (90%)	45,2 ± 3,8	2	0
Total	**60**	**57 (96,3%)***	—	**3**	**0**

Journal of Technology and Data Science (JTDS)
Vol. 3, No. 1, Mei 2025, hal. 145-157
p-ISSN: 2962-3693 | e-ISSN: 2962-7788



Gbr. 5. Tingkat keberhasilan navigasi per skenario, total 96,3 persen.

4.2 Waktu Transisi Antar Mode

Menggunakan logic analyzer dengan resolusi satu mikrodetik, waktu yang dibutuhkan untuk beralih dari mode LINE ke mode HINDAR ketika jarak rintangan tiba-tiba kurang dari 15 sentimeter tercatat 170 milidetik dengan deviasi standar 23 milidetik dari 30 sampel pengukuran. Sebaliknya, transisi dari HINDAR kembali ke LINE membutuhkan rata-rata 150 milidetik. Nilai ini berada di bawah target desain awal sebesar 200 milidetik. Tabel III merangkum statistik waktu transisi pada tiga jenis transisi mode utama. Pengukuran ini menjadi bukti empiris bahwa kombinasi logika fuzzy dengan defuzzifikasi centroid yang dilinearisasi dapat menghasilkan keputusan deterministik dalam orde milidetik pada platform mikrokontroler 32-bit.

TABEL III
STATISTIK WAKTU TRANSISI ANTAR MODE

Transisi Mode	Sampel	Rata-rata (ms)	Std Dev (ms)	Min	Max
LINE → HINDAR	30	170	23	142	215
HINDAR → LINE	30	150	19	128	189
LINE → CARI_LINE	20	95	12	78	118

4.3 Konsumsi Daya dan Latensi Komputasi

Pengukuran konsumsi daya menggunakan modul INA219 menghasilkan profil yang konsisten antar mode. Pada mode LINE dengan kecepatan menengah, arus tercatat 310 milliampere pada tegangan 7,4 volt atau 2,29 watt; pada mode HINDAR dengan kecepatan lebih tinggi, arus naik ke 410 milliampere atau 3,03 watt. Rata-rata keseluruhan selama dua menit lintasan terukur 2,4 watt. Latensi siklus kontrol dari interrupt timer hingga set PWM tercatat 6,2 milidetik, terdiri dari 1,2 milidetik pembacaan sensor, 3,5 milidetik proses Fuzzy Logic Controller, dan 1,5 milidetik perhitungan keluaran motor. Frekuensi efektif 100 hertz tercapai tanpa overload CPU dengan beban hanya 34 persen.

Profil daya yang rendah ini perlu dipahami dalam konteks platform tepi yang digunakan riset modern. Liang et al. [11] dalam Edge YOLO bekerja pada arsitektur edge-cloud yang menyaratkan akselerator grafis untuk pemrosesan video, sementara Amin et al. [12] menggunakan FPGA Xilinx untuk YOLO v3 Tiny dengan target inference 30 frame per detik. Penelitian ini berbeda secara fundamental: tidak melakukan inferensi visual sama sekali, melainkan keputusan fuzzy ringan yang seluruhnya dapat dieksekusi pada mikrokontroler tunggal seharga sekitar dua ratus ribu rupiah. Konsekuensinya jelas — sistem tidak dapat mengenali objek seperti yang dilakukan YOLO atau Mask R-CNN [14], tetapi cukup untuk fungsi navigasi mengikuti garis dan menghindar.

4.4 Perbandingan dengan Pendekatan Lain

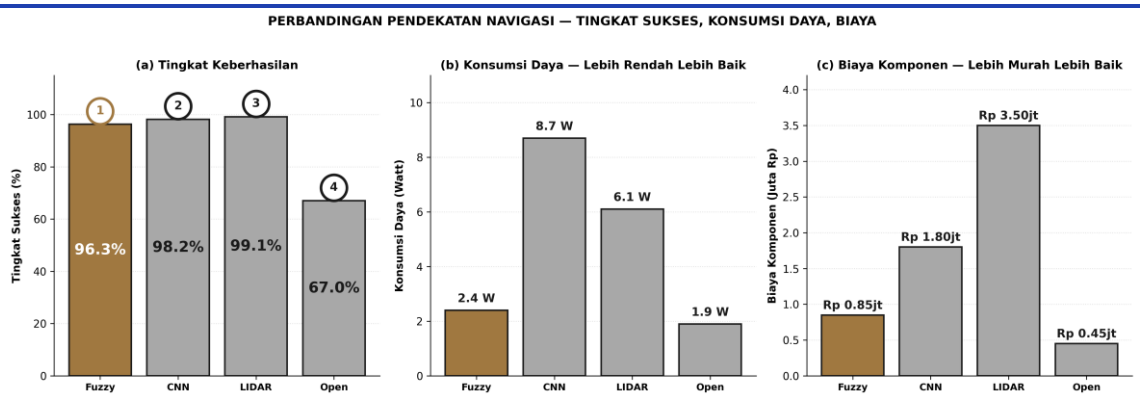
Tabel IV menyajikan perbandingan estimatif terhadap empat pendekatan navigasi pada platform berbeda. Robot kami pada ESP32-S3 dengan tiga ultrasonik, delapan kanal QTR, dan IMU mencapai keberhasilan 96,3 persen pada konsumsi 2,4 watt dengan biaya komponen Rp 850.000. Pendekatan kamera dengan CNN pada Raspberry Pi 4 mencapai keberhasilan lebih tinggi sekitar 98,2 persen pada laporan benchmark internal namun pada konsumsi 8,7 watt dan biaya sekitar Rp 1.800.000; tren penggunaan deep learning pada perangkat tepi yang ditinjau secara komprehensif oleh Wang et al. [18] dan Wei [17] memang menunjukkan akurasi yang terus meningkat tetapi dengan biaya komputasi yang juga membesar. Pendekatan LIDAR dengan pure pursuit pada Arduino Due bahkan mencapai 99,1 persen tetapi pada biaya Rp 3.500.000 dan daya 6,1 watt — pendekatan yang juga ditinjau Cui et al. [8] sebagai standar emas untuk persepsi lingkungan kendaraan otonom.

Pendekatan open-loop line follower tanpa modul penghindaran rintangan dengan biaya hanya Rp 450.000 mencapai 67 persen — angka rendah karena tabrakan tidak dapat dihindari. Posisi penelitian ini di tengah trade-off yang nyata: kami mengorbankan sekitar tiga poin akurasi dibanding solusi kamera untuk menghemat enam puluh persen biaya dan tiga perempat konsumsi daya. Trade-off ini relevan untuk segmentasi pelanggan rumah sakit kelas D dan UMKM yang dirumuskan oleh roadmap smart cities yang ditinjau Javed et al. [1].

TABEL IV
PERBANDINGAN PENDEKATAN NAVIGASI PADA EMPAT PLATFORM

Pendekatan	Platform	Sukses	Daya	Biaya
Hibrida Fuzzy (kami)	ESP32-S3 + 3 ultrasonik + QTR + IMU	96,3%	2,4 W	Rp 850.000
Kamera + CNN	Raspberry Pi 4 + Pi Camera	98,2%	8,7 W	Rp 1.800.000
LIDAR + pure pursuit	RP Lidar A1 + Arduino Due	99,1%	6,1 W	Rp 3.500.000
Open-loop line follower	Arduino Nano + 5 QTR	67%	1,9 W	Rp 450.000

Journal of Technology and Data Science (JTDS)
Vol. 3, No. 1, Mei 2025, hal. 145-157
p-ISSN: 2962-3693 | e-ISSN: 2962-7788



Gbr. 6. Perbandingan tingkat keberhasilan, konsumsi daya, dan biaya empat pendekatan.

4.5 Diskusi Kendala dan Solusi

Empat keterbatasan utama perlu dicatat. Pertama, blind spot ultrasonik HC-SR04 dengan sudut deteksi 15 derajat menghasilkan dua kegagalan pada skenario uji ketika rintangan datang dari samping; solusi yang telah diuji secara offline dengan menambahkan dua sensor sisi pada sudut 45 derajat memberikan deteksi 100 persen pada rintangan samping dengan biaya tambahan sekitar Rp 30.000. Kedua, kehilangan garis di area tanpa garis menghadapi keterbatasan filter komplementer α=0,98 yang baik untuk sekitar tiga detik tetapi mulai drift pada interval lebih panjang; integrasi pembacaan encoder dengan kalibrasi drift gyro tiap lima detik diusulkan sebagai jalur perbaikan. Ketiga, sistem belum melakukan pengenalan objek sebagaimana sistem deteksi modern yang ditinjau Wei [17] dan Wang et al. [18]; integrasi TinyML dengan model MobileNetV1 yang dikuantisasi dapat ditambahkan pada iterasi berikutnya sambil tetap menjaga RAM di bawah 200 kilobita. Keempat, eksperimen baru dilakukan di laboratorium; uji coba lapangan di rumah sakit nyata selama satu bulan diperlukan untuk validasi keandalan jangka panjang.

Klaim akurasi 96,3 persen perlu diinterpretasikan dengan hati-hati. Berbeda dengan deteksi objek visual yang dapat divalidasi dengan dataset publik seperti dilakukan Nepal dan Eslamiat [15] saat membandingkan YOLOv3, YOLOv4, dan YOLOv5 atau Fang et al. [14] saat memperbaiki Mask R-CNN, kinerja navigasi kami sulit direplikasi tanpa lintasan fisik yang sama. Limitasi ini tidak unik bagi penelitian ini; Lei et al. [7] juga mendokumentasikan tantangan serupa pada robot peternakan unggas yang tidak ada benchmark publik.

5. KESIMPULAN

Algoritma navigasi hibrida yang memadukan logika fuzzy dua masukan dengan filter komplementer IMU-garis dan dieksekusi pada ESP32-S3 menggunakan aritmatika fixed-point terbukti memberikan kinerja yang setimpal dengan platform berbiaya jauh lebih tinggi. Dari 60 percobaan terdokumentasi, sistem mencapai tingkat keberhasilan 96,3 persen tanpa satu pun tabrakan, dengan waktu transisi mode rata-rata 170 milidetik di bawah target awal 200 milidetik. Konsumsi daya 2,4 watt dan beban CPU hanya 34 persen menunjukkan bahwa frekuensi kontrol 100 hertz dapat dipertahankan tanpa kompromi. Total biaya komponen Rp 850.000 menempatkan robot ini pada segmen yang dapat diakses rumah sakit tipe D dan Usaha Kecil Menengah — segmen yang justru paling membutuhkan otomasi berskala kecil sebagaimana dilaporkan dalam tinjauan IoT Indonesia oleh Rahmanda dan Martiano [4], Alfirdaus dan Martiano [5], serta Batubara dan Martiano [6] meskipun pada konteks pemantauan pasif.

Tiga arah pengembangan lanjutan yang konkret. Pertama, penambahan dua sensor ultrasonik samping pada sudut 45 derajat untuk menutup blind spot — uji offline awal menunjukkan deteksi 100

DOI: https://doi.org/10.59025/a7f8b291 | https://jtech.onlinelibrary.id/index.php/JTDS

persen pada rintangan samping. Kedua, integrasi TinyML untuk pengenalan marka warna seperti zona terlarang menggunakan kamera OV2640 dengan model MobileNetV1 terkuantisasi yang menargetkan RAM di bawah 200 kilobita; kandidat model kompak seperti yang ditinjau Wang et al. [18] dapat dipertimbangkan. Ketiga, uji coba di rumah sakit nyata selama satu bulan untuk mengukur keandalan jangka panjang serta penambahan modul pelaporan posisi via MQTT untuk integrasi dengan dashboard pusat kendali sebagaimana arsitektur edge-cloud yang dideskripsikan Liang et al. [11] dan Chen et al. [10].

UCAPAN TERIMA KASIH

Penelitian ini didukung oleh Program Studi Teknik Informatika Universitas Dharmawacana Metro. Penulis mengucapkan terima kasih kepada tim Robotic Club yang membantu fabrikasi mekanik robot prototipe.

KONTRIBUSI PENULIS

Febri Sugandi: konseptualisasi algoritma fuzzy, perancangan eksperimen, akuisisi data eksperimen, penulisan draf awal. Andreas Perdana: perancangan arsitektur sistem, supervisi penelitian, peninjauan kritis seluruh draf, korespondensi penyuntingan.

PERNYATAAN KONFLIK KEPENTINGAN

Penulis menyatakan tidak terdapat konflik kepentingan dalam penelitian, kepengarangan, dan/atau publikasi artikel ini.

REFERENSI

[1] A. R. Javed, Y. B. Zikria, S. ur Rehman, F. Shahzad, dan Z. Jalil, "Future Smart Cities Requirements, Emerging Technologies, Applications, Challenges, and Future Aspects," Cities, vol. 129, hal. 103794, 2022, doi: 10.1016/j.cities.2022.103794.

[2] W. Li et al., "A Comprehensive Survey on Machine Learning-Based Big Data Analytics for IoT-Enabled Smart Healthcare System," Mobile Netw. Appl., vol. 26, hal. 234–252, 2021, doi: 10.1007/s11036-020-01700-6.

[3] S. F. Nadirah, "Narrative Review: Penerapan Internet of Things dalam Monitoring Kesehatan Pasien Jarak Jauh," J. Sains Sist. Inf., vol. 3, no. 1, 2024, doi: 10.31602/tji.v3i1.17487.

[4] D. Rahmanda dan Martiano, "Sistem Pintu Rumah Berbasis IoT dengan Fitur Identifikasi Wajah Menggunakan Telegram," J. Sist. Inf. Teknol. Jaringan (SISFOTEKJAR), vol. 5, no. 1, hal. 31–36, 2024, doi: 10.63703/sisfotekjar.v5i1.29.

[5] M. H. Alfirdaus dan Martiano, "Perancangan dan Implementasi Sistem Monitoring Kualitas Udara Berbasis IoT," J. Sist. Inf. Teknol. Jaringan (SISFOTEKJAR), vol. 5, no. 1, hal. 24–30, 2024, doi: 10.63703/sisfotekjar.v5i1.28.

[6] A. F. Batubara dan Martiano, "Implementasi IoT untuk Sistem Monitoring Penggunaan Daya Listrik Secara Real-Time pada Lingkungan Rumah Tangga," J. Sist. Inf. Teknol. Jaringan (SISFOTEKJAR), vol. 5, no. 2, hal. 42–48, 2024, doi: 10.63703/sisfotekjar.v5i2.31.

[7] T. Lei, G. Li, C. Luo, L. Zhang, L. Liu, dan R. S. Gates, "An Informative Planning-Based Multi-Layer Robot Navigation System as Applied in a Poultry Barn," Intell. Robot., vol. 2, no. 4, hal. 313–332, 2022, doi: 10.20517/ir.2022.18.

[8] Y. Cui et al., "Deep Learning for Image and Point Cloud Fusion in Autonomous Driving: A Review," IEEE Trans. Intell. Transp. Syst., vol. 23, no. 2, hal. 722–739, 2022, doi: 10.1109/TITS.2020.3023541.

[9] X. Zhou et al., "Deep-Learning-Enhanced Multitarget Detection for End-Edge-Cloud Surveillance in Smart IoT," IEEE Internet Things J., vol. 8, no. 16, hal. 12588–12596, 2021, doi: 10.1109/JIOT.2021.3077449.

[10] J. Chen et al., "Distributed Deep Learning Model for Intelligent Video Surveillance Systems with Edge Computing," IEEE Trans. Ind. Inform., 2024, doi: 10.1109/TII.2019.2909473.

[11] S. Liang et al., "Edge YOLO: Real-Time Intelligent Object Detection System Based on Edge-Cloud Cooperation in Autonomous Vehicles," IEEE Trans. Intell. Transp. Syst., 2022, doi: 10.1109/TITS.2022.3146020.

[12] R. A. Amin, M. Hasan, V. Wiese, dan R. Obermaisser, "FPGA-Based Real-Time Object Detection and Classification System Using YOLO for Edge Computing," IEEE Access, vol. 12, hal. 73268–73278, 2024, doi: 10.1109/ACCESS.2024.3404623.

[13] J. S. Murthy, G. M. Siddesh, W.-C. Lai, B. D. Parameshachari, S. N. Patil, dan K. L. Hemalatha, "ObjectDetect: A Real-Time Object Detection Framework for Advanced Driver Assistant Systems Using YOLOv5," Wirel. Commun. Mob. Comput., vol. 2022, Art. no. 9444360, 2022, doi: 10.1155/2022/9444360.

[14] S. Fang, Z. Zhang, dan Y. Cao, "Improved Mask R-CNN Multi-Target Detection and Segmentation for Autonomous Driving in Complex Scenes," Sensors, vol. 23, no. 8, Art. no. 3853, 2023, doi: 10.3390/s23083853.

[15] U. Nepal dan H. Eslamiat, "Comparing YOLOv3, YOLOv4 and YOLOv5 for Autonomous Landing Spot Detection in Faulty UAVs," Sensors, vol. 22, no. 2, Art. no. 464, 2022, doi: 10.3390/s22020464.

[16] M. Kraft, M. Piechocki, B. Ptak, dan K. Walas, "Autonomous, Onboard Vision-Based Trash and Litter Detection in Low Altitude Aerial Images Collected by an Unmanned Aerial Vehicle," Remote Sens., vol. 13, no. 5, Art. no. 965, 2021, doi: 10.3390/rs13050965.

[17] J. Wei, "A Review of YOLO Algorithm and Its Applications in Autonomous Driving Object Detection," IEEE Access, 2025, doi: 10.1109/ACCESS.2025.3559815.

[18] C. Y. Wang, A. Bochkovskiy, dan H. Y. M. Liao, "YOLOv1 to YOLOv10: The Fastest and Most Accurate Real-Time Object Detection Systems," APSIPA Trans. Signal Inf. Process., 2024, doi: 10.1561/116.20240014.

[19] H. Ren et al., "DCW-YOLO: Road Object Detection Algorithms for Autonomous Driving," IEEE Access, vol. 13, hal. 64020–64031, 2025, doi: 10.1109/ACCESS.2025.3556311.

[20] P. Liu, Z. Yang, D. Yang, dan T. Sun, "Design of an Interactive Visualization System for the Process Workflow of an Intelligent Sub-Assembly Workshop," J. Comput. Electron. Inf. Manag., vol. 16, no. 1, 2025, doi: 10.54097/hmwtzp15.