

Real-time Multi-Level Wireless Control Model based on IoT for Wheeled Robots

Budi Rahmani^{1*}, Ajib Alfarizi², Ilmi³, Miftahuddin⁴, M. Agus Wahyudi⁵, Melyana⁶, Mahrita⁷
 Teknik Informatika, STMIK Banjarbaru, Banjarbaru, Indonesia
 *e-mail *Corresponding Author*: budirahmani@gmail.com

Abstract

Demands regarding a wireless control system for all-wheeled soccer-playing robots in the 2023 Indonesian robot contest require significant changes to the STMIK Banjarbaru's robot design. Therefore, the team designed and tested the robot control program using an ATmega2560 microcontroller and a laptop on top of the robot. The wheeled robot used another laptop as a controller outside the robot. This research reprogrammed the soccer player's wheeled robot prototype from the microcontroller controller side regarding the robot's movement using a holonomic model. Then, the microcontroller is programmed to receive commands to move the robot from the laptop installed on the robot via serial communication with a USB cable. This study also created an application in the form of a GUI (Graphical User Interface) on the laptop on the robot. Then, it is also ready to be controlled wirelessly from the laptop outside the robot. The test results show that this research has overcome the main problems related to response time and consistency in carrying out essential motor functions, including forward, backward, sliding, and rotating movements on the robot. Additionally, this experiment successfully met the challenges of performing specialized functions such as motor dribbling and ball kicking with a significant level of success. Wireless test results using access point frequencies of 2.4 GHz and 5.8 GHz also produced exemplary achievements, demonstrating the robot's ability to communicate efficiently in various environments.

Keywords: *ATmega2560 microcontroller; Robot; Holonomic; Wireless control; Response time*

Abstrak

Tuntutan terkait sistem kendali nirkabel atas semua robot beroda pemain bola di kontes robot Indonesia tahun 2023 menuntut perubahan signifikan terhadap rancangan robot oleh tim di STMIK Banjarbaru. Karenanya tim telah merancang bangun sekaligus mengujicoba program pengendali robot menggunakan mikrokontroler ATmega2560 serta laptop yang diletakkan di atas robot. Selain itu juga digunakan laptop lain sebagai pengendali di luar robot. Adapun yang dilakukan adalah dengan memrogram ulang prototipe robot beroda pemain bola dari sisi pengendali mikrokontroler terkait gerakan robot dengan model *holonomic*. Kemudian Mikrokontroler diprogram untuk juga bisa menerima perintah menggerakkan robot dari laptop yang dipasang di robot melalui komunikasi serial dengan kabel USB. Selanjutnya, pada laptop yang ada di robot dibuatkan aplikasi berupa GUI (*Graphical User Interface*) untuk kemudian juga siap dikendalikan secara nirkabel dari laptop di luar robot. Hasil pengujian menunjukkan bahwa penelitian ini berhasil mengatasi permasalahan utama terkait waktu respon dan konsistensi dalam menjalankan fungsi motorik dasar, termasuk pergerakan maju, mundur, geser, dan berputar pada robot. Selain itu, eksperimen ini berhasil menghadapi tantangan dalam menjalankan fungsi khusus seperti motor *dribble* dan penendang bola dengan tingkat keberhasilan yang signifikan. Hasil pengujian nirkabel menggunakan frekuensi akses poin 2.4 GHz dan 5.8 GHz juga menghasilkan pencapaian yang baik, menunjukkan kemampuan robot untuk berkomunikasi secara efisien dalam berbagai lingkungan.

Kata kunci: *Mikrokontroler ATmega2560; Robot; Holonomic; Kendali nirkabel; Response time*

1. Pendahuluan

Dalam perkembangan teknologi robotika, penggunaan robot beroda dengan sistem penggerak holonomik telah menjadi tren yang signifikan [1][2]. Robot-robot semacam itu mampu melakukan gerakan omnidireksional dan manuver yang kompleks dengan

menggunakan roda yang dapat berputar independen [3]–[7]. Namun, dalam lingkungan yang dinamis dan kompleks, kendali yang efisien dan tepat pada robot beroda ini masih merupakan tantangan yang belum terselesaikan [8].

Salah satu pendekatan yang menjanjikan adalah penggunaan sistem kendali nirkabel berbasis *Internet of Things* (IoT) pada robot beroda secara real-time [9]–[11]. Model kendali nirkabel multi level pada robot semacam ini memungkinkan penggunaan jaringan nirkabel untuk mengirimkan perintah kendali dari perangkat eksternal ke robot secara *real-time*, sehingga memungkinkan interaksi yang lebih cepat dan responif antara pengguna dan robot [12].

Tuntutan terkait kendali nirkabel terhadap robot di kasus ini juga sesuai dengan aturan di Kontes Robot Indonesia, khususnya pada divisi KRSBI-Beroda. Di divisi ini, robot yang dilombakan harus bebas dari proses pengoperasian secara manual. Artinya setelah robot disiapkan di arena, kemudian tidak boleh ada lagi sentuhan secara fisik ke robot, baik oleh anggota tim maupun mendamping, bahkan juri sekalipun [13], [14].

Namun, saat ini masih ada beberapa masalah yang perlu diatasi dalam pengembangan dan implementasi model kendali nirkabel multi level pada robot beroda secara real-time. Salah satunya adalah masalah latensi dan keandalan komunikasi antara perangkat pengirim dan penerima, karena penggunaan jaringan nirkabel yang rentan terhadap gangguan dan kehilangan sinyal [15].

Selain itu, integrasi yang efektif antara sistem kendali nirkabel dengan sistem penggerak holonomik juga merupakan tantangan tersendiri. Sistem kendali harus mampu mengoptimalkan gerakan robot beroda secara real-time agar dapat melakukan manuver dengan akurasi tinggi dan menghindari tabrakan atau kegagalan sistem [16]. Oleh karena itu, penelitian ini bertujuan untuk mengoptimalkan sistem komunikasi nirkabel untuk mengendalikan gerakan robot beroda pemain bola yang dirancang bangun. Adapun yang dikerjakan adalah mengembangkan dan mengimplementasikan model kendali nirkabel multi level berbasis IoT pada robot beroda secara real-time. Dengan demikian, penelitian ini diharapkan dapat meningkatkan kinerja dan kemampuan robot beroda tersebut dalam menjalankan tugas-tugas yang kompleks dan interaktif dalam lingkungan yang dinamis.

2. Tinjauan Pustaka

Robot beroda dengan sistem penggerak holonomik telah menjadi tren yang signifikan dalam perkembangan teknologi robotika. Sistem ini memungkinkan robot untuk melakukan gerakan omnidireksional dan manuver yang kompleks dengan menggunakan roda yang dapat berputar independen. Dengan kemampuan ini, robot beroda dapat dengan mudah bergerak dalam ruang yang terbatas dan mengubah arah dengan cepat tanpa perlu mengubah orientasi keseluruhan robot. Kelebihan ini membuat robot beroda secara real-time menjadi pilihan yang menarik untuk aplikasi yang membutuhkan mobilitas tinggi dan manuver yang presisi [6], [17].

Meskipun robot beroda secara real-time memiliki potensi yang besar, kendali yang efisien dan tepat pada robot semacam itu masih menjadi tantangan yang belum terselesaikan. Dalam lingkungan yang dinamis dan kompleks, robot harus mampu merespon dengan cepat terhadap perintah kendali dan mengoptimalkan gerakan-gerakan yang diperlukan. Oleh karena itu, pengembangan sistem kendali yang dapat memenuhi kebutuhan ini menjadi krusial untuk meningkatkan kinerja robot beroda [3].

Salah satu pendekatan yang cukup memungkinkan adalah penggunaan sistem kendali nirkabel berbasis *Internet of Things* (IoT) pada robot beroda secara real-time [10]. Model kendali nirkabel multi level pada robot semacam ini memungkinkan penggunaan jaringan nirkabel untuk mengirimkan perintah kendali dari perangkat eksternal ke robot secara real-time. Dengan menggunakan jaringan nirkabel, interaksi antara pengguna dan robot dapat menjadi lebih cepat dan responif.

Beberapa contoh penggunaan sistem kendali nirkabel berbasis IoT pada robot beroda secara real-time diantaranya adalah robot yang memonitor kondisi pasien di rumah pada masa pandemi covid-19 yang lalu. Menggunakan robot berbasis IoT ini, perawat tidak harus mendatangi dan melihat serta memonitor kondisi pasien di ruang perawatan. Proses pendatangi pasien dilakukan jika memang kondisi pasien berdasarkan alat pemonitor yang dipasang pada pasien menunjukkan hal yang memprihatinkan [18].

Ada pula sebuah robot beroda yang secara real-time dapat digunakan sebagai robot pelayan di industri perhotelan atau restoran. Dalam penelitian [19] robot secara real-time dapat

dikendalikan secara nirkabel melalui sistem berbasis IoT. Melalui aplikasi atau perangkat pengendali yang terhubung ke jaringan internet, pengguna dapat memberikan perintah dan mengontrol gerakan robot secara real-time dalam sebuah rumah. Robot ini lebih kepada robot pembantu yang beroperasi secara otonom dan dapat dipantau secara nirkabel via. Jika dioperasikan di restoran pengguna dapat memesan makanan atau minuman melalui aplikasi khusus yang terhubung ke sistem kendali nirkabel berbasis IoT. Robot akan menerima pesanan tersebut dan menggunakan sistem holonomic drive untuk bergerak secara lancar menuju meja pelanggan dengan cepat dan akurat. Ini menggabungkan teknologi IoT, sistem kendali nirkabel, dan sistem penggerak holonomik untuk memberikan pengalaman pelayanan yang inovatif dan efisien [20].

Robot asisten pribadi yang dilengkapi secara real-time dapat dikendalikan melalui sistem kendali nirkabel berbasis IoT. Pengguna dapat menggunakan perangkat pintar seperti smartphone atau speaker pintar yang terhubung ke jaringan internet untuk memberikan perintah dan mengontrol robot. Robot ini dapat membantu dengan tugas-tugas sehari-hari, seperti membersihkan rumah, mengatur jadwal, atau memberikan informasi yang diperlukan. Sistem kendali nirkabel memungkinkan interaksi yang mudah dan responif antara pengguna dan robot, meningkatkan efisiensi dan kemudahan penggunaan [21].

Penggunaan sistem kendali nirkabel berbasis IoT pada robot beroda secara real-time memberikan fleksibilitas, kemudahan penggunaan, dan interaksi real-time antara pengguna dan robot. Hal ini membuka potensi aplikasi yang luas di berbagai bidang, termasuk eksplorasi, perhotelan, hiburan, dan banyak lagi. Tuntutan terkait kendali nirkabel terhadap robot dalam kasus ini juga sesuai dengan aturan di Kontes Robot Indonesia, khususnya pada divisi KRSBI-Beroda [22]. Di divisi ini, robot yang dilombakan harus bebas dari proses pengoperasian secara manual. Setelah robot disiapkan di arena, tidak boleh ada lagi sentuhan fisik ke robot, baik oleh anggota tim maupun pihak lain, termasuk juri. Oleh karena itu, pengembangan dan implementasi model kendali nirkabel multi level pada robot beroda secara real-time menjadi penting untuk memenuhi persyaratan kontes dan meningkatkan kualitas kompetisi robotika.

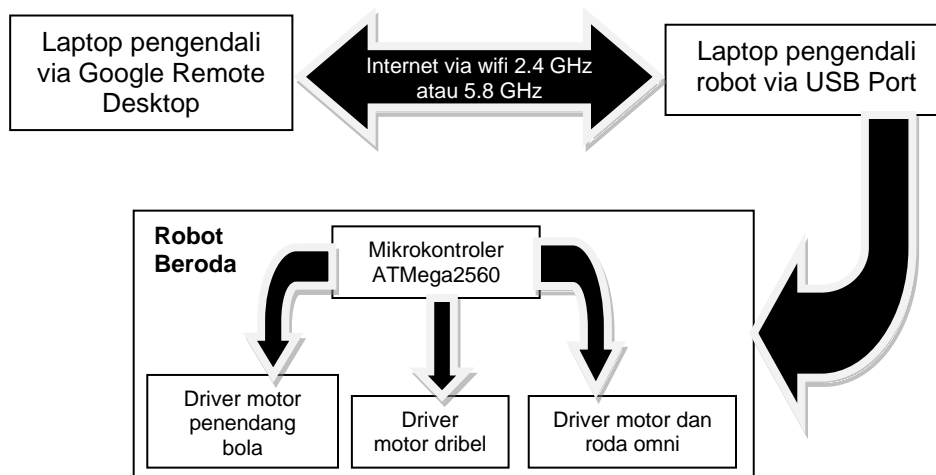
Tantangan dalam Implementasi Kendali Nirkabel: Meskipun pendekatan kendali nirkabel berbasis IoT menjanjikan, masih ada beberapa masalah yang perlu diatasi dalam pengembangan dan implementasi model kendali nirkabel multi level pada robot beroda secara real-time. Salah satu masalah utama adalah masalah latensi dan keandalan komunikasi antara perangkat pengirim dan penerima. Keterbatasan jaringan nirkabel yang rentan terhadap gangguan dan kehilangan sinyal dapat menghambat responivitas dan stabilitas sistem kendali. Selain itu, integrasi yang efektif antara sistem kendali nirkabel dengan sistem penggerak holonomik juga merupakan tantangan tersendiri, di mana sistem kendali harus mampu mengoptimalkan gerakan robot beroda dengan akurasi tinggi agar dapat melakukan manuver yang presisi dan menghindari tabrakan atau kegagalan sistem.

Penelitian ini telah mengembangkan dan mengimplementasikan model kendali nirkabel multi level berbasis IoT pada robot beroda dengan *holonomic drive system* secara *real-time*. Penelitian terkait terdahulu [7] dan [5] belum mengimplementasikan sistem kendali secara nirkabel via internet (IoT) dan hanya sampai proses pengolahan citra di laptop dan pemberian perintah dari laptop ke mikrokontroler. Hasil penelitian ini diharapkan dapat mengatasi masalah-masalah yang ada dan meningkatkan kinerja serta kemampuan robot beroda dalam menjalankan tugas-tugas yang kompleks dan interaktif dalam lingkungan yang dinamis. Hasil penelitian diharapkan berkontribusi pada pengembangan teori pengendalian robot secara nirkabel berbasis IoT serta menjadi referensi bagi penelitian lanjutan.

3. Metodologi

Sebelum proses pengendalian secara nirkabel berbasis IoT diimplementasikan pada robot beroda di penelitian ini, maka sebelumnya peneliti dan tim robot menggunakan fasilitas *Refreebox* standar kontes robot internasional dan nasional. Namun karena koneksi secara nirkabel ke perangkat mikrokontroler secara langsung belum dimungkinkan karena keterbatasan waktu dan sumber daya programmer, maka mode tersebut tidak digunakan. Sementara yang digunakan adalah mode kendali komputer (laptop) ke mikrokontroler. Laptop sebagai *main controller* atau pengendali utama dan mikrokontroler yang terhubung langsung ke perangkat driver motor dan lain-lain adalah sebagai *secondary controller*.

Penelitian kali ini mengembangkan apa yang sudah dibangun sebelumnya terkait *main controller* dan *secondary controller*. Pengembangan yang dilakukan *main controller* adalah pada *user interface* aplikasi dengan ragam kendali gerakan robot yang bisa diberikan ke *secondary controller* (Mikokontroler). Gambar 1 memperlihatkan blok diagram sistem kendali multi level berbasis IoT yang dikembangkan pada penelitian ini. *Google Remote Desktop* sebagai media mengoperasikan robot secara nirkabel via internet yang terkoneksi dengan jaringan 2.4 GHz atau 5.8 Ghz. Kedua koneksi tersebut diuji coba dan dianalisis secara terpisah.



Gambar 1. Blok diagram sistem kendali multi level berbasis IoT

Pseudocode dari program yang ada di mikrokontroler adalah:

```

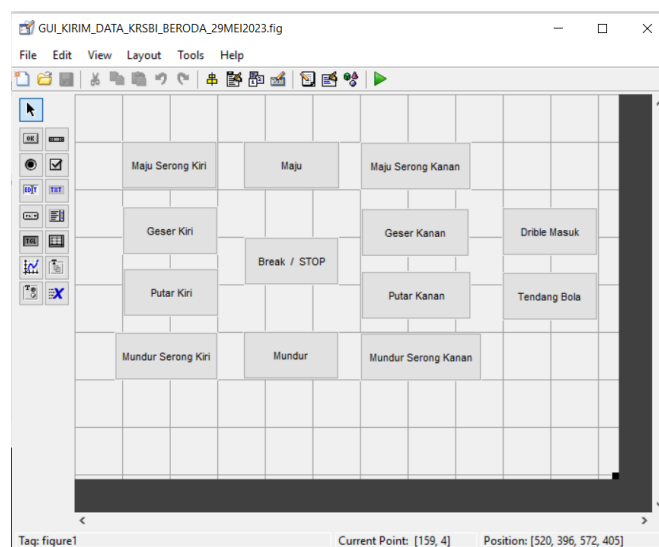
Initialize pins for motor control, PWM, and other peripherals
Function cobaMotorku(inA1, inB1, inA2, inB2, inA3, inB3, inA4, inB4, pwm):
  Set the direction and PWM for each motor based on the given input
Function maju(pwm):
  Call cobaMotorku with appropriate parameters to move the robot forward
Function mundur(pwm):
  Call cobaMotorku with appropriate parameters to move the robot backward
Function geserKiri(pwm):
  Call cobaMotorku with appropriate parameters to move the robot to the left
Function geserKanan(pwm):
  Call cobaMotorku with appropriate parameters to move the robot to the right
Function belokKiri(pwm):
  Call cobaMotorku with appropriate parameters to turn the robot to the left
Function belokKanan(pwm):
  Call cobaMotorku with appropriate parameters to turn the robot to the right
Function majuSerongKiri(pwm):
  Call cobaMotorku with appropriate parameters to move the robot diagonally forward to the left
Function majuSerongKanan(pwm):
  Call cobaMotorku with appropriate parameters to move the robot diagonally forward to the right
Function mundurSerongKiri(pwm):
  Call cobaMotorku with appropriate parameters to move the robot diagonally backward to the left
Function mundurSerongKanan(pwm):
  Call cobaMotorku with appropriate parameters to move the robot diagonally backward to the right
Function breakMotor():
  Stop all motors by setting their control pins HIGH
Function dribelKeluar():
  Activate the dribel mechanism to extend the dribel out
Function dribelMasuk():
  Activate the dribel mechanism to retract the dribel in
  
```

Function dribbleBreak():
 Stop the dribble mechanism by setting its control pins HIGH
Function penandangON():
 Activate the kicking mechanism by setting the appropriate control pins
Function penandangOFF():
 Deactivate the kicking mechanism by setting the appropriate control pins
Function dribbleBolaIn():
 Call *dribbleMasuk()* and *penandangOFF()* to prepare for ball intake
Function tendangBola():
 Call *dribbleKeluar()* and *penandangON()* to initiate ball kicking
Function bacaDataSerialMatlab():
 Read serial data from MATLAB and execute corresponding commands to control the robot
Function setup():
 Initialize the LCD, serial communication, and pin modes
Function loop():
 Continuously read and process data from MATLAB to control the robot

Adapun algoritme dari *pseducode* yang disampaikan sebelumnya adalah:

Initialize hardware (pin modes, PWM, etc.)
Setup the LCD and print a welcome message
Define movement functions (e.g., maju, mundur, geserKanan, geserKiri, belokKanan, belokKiri, majuSerongKanan, majuSerongKiri, mundurSerongKanan, mundurSerongKiri)
Define dribbling functions (e.g., dribbleKeluar, dribbleMasuk, dribbleBreak)
Define kicking function (e.g., tendangBola)
Define function to stop all motors (e.g., breakMotor)
Define function to read data from MATLAB and execute corresponding actions (e.g., bacaDataSerialMatlab)
In the Loop:
 Call *bacaDataSerialMatlab* function to check for incoming commands from MATLAB and execute the corresponding robot actions

Desain GUI pada *main controller* diperlihatkan pada Gambar 2. Desain antar muka ini dibuat menggunakan aplikasi Matlab 2013a. Pada dasarnya antar muka yang dibuat ini akan melakukan sinkronisasi komunikasi ke mikrokontroler melalui jalur USB ke Arduino board yang digunakan (Arduino Mega2560). Jika nomor port serial sudah sama atau disamakan, maka tidak ada pesan *error* yang muncul ketika program *interface* ini dijalankan.



Gambar 2. Desain GUI menggunakan Matlab

Penelitian ini mendokumentasikan pengujian berbagai fungsi yang dijalankan pada mikrokontroler ATmega2560 untuk mengendalikan robot dalam bentuk tabel-tabel hasil pengujian. Tabel-tabel tersebut berisi informasi tentang fungsi-fungsi yang diuji, reaksi gerak robot yang diharapkan saat menjalankan setiap fungsi, dan hasil pengamatan reaksi gerak robot secara keseluruhan pada setiap pengujian. Pengujian dilakukan untuk memvalidasi bahwa mikrokontroler dan perangkat keras robot berfungsi dengan benar. Kemudian juga untuk memastikan bahwa reaksi robot sesuai dengan yang diinginkan saat berbagai fungsi dijalankan. Hasil pengamatan dapat digunakan untuk mengidentifikasi masalah dan melakukan perbaikan jika ada reaksi yang tidak sesuai dengan yang diharapkan. Pengujian lain juga memastikan bahwa mikrokontroler ATmega2560 pada robot merespon dengan benar terhadap instruksi yang diberikan baik melalui GUI secara langsung di laptop pada robot maupun pada laptop yang mengendalikannya secara nirkabel menggunakan jaringan wifi dengan frekuensi 2,4 GHz dan 5,8 GHz. Respon robot diharapkan benar saat diberi instruksi untuk bergerak maju, mundur, geser, atau berputar. Selain itu juga akan dicatat waktu respon dari saat tombol ditekan.

Berikutnya Gerak Serong, guna memastikan apakah robot dapat bergerak serong dengan benar, baik ke kanan maupun ke kiri. Selain itu juga akan dicatat waktu respon dari saat tombol ditekan. Kemudian Stop/Break dan Dribble Masuk, guna memastikan semua motor roda berhenti saat tombol "Stop/Break" ditekan. Selanjutnya juga memastikan apakah motor dribble bergerak untuk menarik bola saat tombol "Dribble Masuk" ditekan. Pada bagian ini juga akan dicatat terkait waktu respon dari masing-masing instruksi. Berikutnya adalah Tendang Bola, dan disini ditujukan untuk memastikan robot tidak bergerak, tetapi motor penendang bergerak untuk mendorong bola. Disini juga akan dicatat waktu respon setelah tombol "Tendang Bola" ditekan.

4. Hasil dan Pembahasan

Pengujian yang dilakukan pada penelitian ini ditunjukkan pada tabel-tabel berikut ini:

Tabel 1 . Hasil Pengujian *Function* di mikrokontroler ATmega2560

No	Function di mikrokontroler ATmega2560 yang dijalankan secara individu	Reaksi gerak robot yang diharapkan	Hasil pengamatan reaksi gerak robot secara keseluruhan pada masing-masing pengujian				
			Pengujian ke-1	Pengujian ke-2	Pengujian ke-3	Pengujian ke-4	Pengujian ke-5
1	Maju selama 2 detik	Bergerak Maju selama 2 detik	Robot berhasil bergerak maju selama 2 detik	Robot berhasil bergerak maju selama 2 detik	Robot berhasil bergerak maju selama 2 detik	Robot berhasil bergerak maju selama 2 detik	Robot berhasil bergerak maju selama 2 detik
2	Mundur selama 2 detik	Bergerak Mundur selama 2 detik	Robot berhasil bergerak mundur selama 2 detik	Robot berhasil bergerak mundur selama 2 detik	Robot berhasil bergerak mundur selama 2 detik	Robot berhasil bergerak mundur selama 2 detik	Robot berhasil bergerak mundur selama 2 detik
3	Geser Kanan selama 2 detik	Bergeser Kanan selama 2 detik	Robot berhasil bergeser ke kanan selama 2 detik	Robot berhasil bergeser ke kanan selama 2 detik	Robot berhasil bergeser ke kanan selama 2 detik	Robot berhasil bergeser ke kanan selama 2 detik	Robot berhasil bergeser ke kanan selama 2 detik
4	Geser Kiri selama 2 detik	Bergeser Kiri selama 2 detik	Robot berhasil bergeser ke kiri selama 2 detik	Robot berhasil bergeser ke kiri selama 2 detik	Robot berhasil bergeser ke kiri selama 2 detik	Robot berhasil bergeser ke kiri selama 2 detik	Robot berhasil bergeser ke kiri selama 2 detik
5	Maju Serong Kanan selama 2 detik	Bergerak Maju Serong Kanan selama 2 detik	Robot berhasil bergerak maju serong kanan selama 2 detik	Robot berhasil bergerak maju serong kanan selama 2 detik	Robot berhasil bergerak maju serong kanan selama 2 detik	Robot berhasil bergerak maju serong kanan selama 2 detik	Robot berhasil bergerak maju serong kanan selama 2 detik
6	Maju Serong Kiri selama 2 detik	Bergerak Maju Serong Kiri selama 2 detik	Robot berhasil bergerak maju serong kiri selama 2 detik	Robot berhasil bergerak maju serong kiri selama 2 detik	Robot berhasil bergerak maju serong kiri selama 2 detik	Robot berhasil bergerak maju serong kiri selama 2 detik	Robot berhasil bergerak maju serong kiri selama 2 detik
7	Mundur Serong Kanan selama 2 detik	Bergerak Mundur Serong Kanan selama 2 detik	Robot berhasil bergerak mundur serong kanan selama 2 detik	Robot berhasil bergerak mundur serong kanan selama 2 detik	Robot berhasil bergerak mundur serong kanan selama 2 detik	Robot berhasil bergerak mundur serong kanan selama 2 detik	Robot berhasil bergerak mundur serong kanan selama 2 detik
8	Mundur Serong Kiri selama 2 detik	Bergerak Mundur Serong Kiri selama 2 detik	Robot berhasil bergerak mundur serong kiri selama 2 detik	Robot berhasil bergerak mundur serong kiri selama 2 detik	Robot berhasil bergerak mundur serong kiri selama 2 detik	Robot berhasil bergerak mundur serong kiri selama 2 detik	Robot berhasil bergerak mundur serong kiri selama 2 detik
9	Berputar ke Kanan selama 2 detik	Berputar ke Kanan selama 2 detik	Robot berhasil berputar ke kanan selama 2 detik	Robot berhasil berputar ke kanan selama 2 detik	Robot berhasil berputar ke kanan selama 2 detik	Robot berhasil berputar ke kanan selama 2 detik	Robot berhasil berputar ke kanan selama 2 detik
10	Berputar ke Kiri selama 2 detik	Berputar ke Kiri selama 2 detik	Robot berhasil berputar ke kiri selama 2 detik	Robot berhasil berputar ke kiri selama 2 detik	Robot berhasil berputar ke kiri selama 2 detik	Robot berhasil berputar ke kiri selama 2 detik	Robot berhasil berputar ke kiri selama 2 detik
11	Motor dribble bola ke arah	Motor dribble bola arah	Motor dribble telah aktif	Motor dribble telah aktif	Motor dribble telah aktif	Motor dribble telah aktif	Motor dribble telah aktif

	dalam dan penandang bola arah dalam	masuk hidup	mengarahkan bola masuk	mengarahkan bola masuk	mengarahkan bola masuk	mengarahkan bola masuk	mengarahkan bola masuk
12	Motor dribble bola ke arah luar dan motor penandang bola bergerak ke arah luar (menendang)	Motor penandang mengarah keluar dan motor dribble juga mengarah ke luar	Motor dribble telah aktif mengarahkan bola keluar bersamaan dengan penandang bola aktif mengarah keluar	Motor dribble telah aktif mengarahkan bola keluar bersamaan dengan penandang bola aktif mengarah keluar	Motor dribble telah aktif mengarahkan bola keluar bersamaan dengan penandang bola aktif mengarah keluar	Motor dribble telah aktif mengarahkan bola keluar bersamaan dengan penandang bola aktif mengarah keluar	Motor dribble telah aktif mengarahkan bola keluar bersamaan dengan penandang bola aktif mengarah keluar
13	Stop/Break	Motor semua roda berhenti	Keempat roda robot berhenti berputar	Keempat roda robot berhenti berputar	Keempat roda robot berhenti berputar	Keempat roda robot berhenti berputar	Keempat roda robot berhenti berputar
14	Dribble break	Motor dribble berhenti	Kedua motor dribble berhenti berputar	Kedua motor dribble berhenti berputar	Kedua motor dribble berhenti berputar	Kedua motor dribble berhenti berputar	Kedua motor dribble berhenti berputar

Tabel 1 menunjukkan hasil pengujian fungsi di mikrokontroler ATmega2560 terkait keberhasilan Fungsi Motorik. Robot mampu melakukan berbagai gerakan seperti maju, mundur, geser, berputar, dan berhenti sesuai dengan perintah dari mikrokontroler ATmega2560. Keberhasilan ini menunjukkan kemampuan mikrokontroler untuk mengendalikan motor dengan akurat. Kemudian konsistensi hasil diperlihatkan pada pengujian yang diulang-ulang (Pengujian ke-1 hingga Pengujian ke-5). Konsistensi ini penting untuk memastikan bahwa robot dapat diandalkan dalam berbagai situasi. Selanjutnya waktu eksekusi, setiap fungsi dijalankan selama 2 detik, dan robot berhasil mematuhi durasi yang diharapkan. Ini menunjukkan bahwa mikrokontroler dapat mengontrol waktu eksekusi dengan akurat, yang penting dalam aplikasi robotik yang memerlukan gerakan yang tepat waktu.

Fungsi Khusus seperti motor dribble dan penandang bola juga berhasil dijalankan. Ini menunjukkan kemampuan mikrokontroler untuk mengkoordinasikan berbagai perangkat keras seperti motor dribble dan motor penandang.

Tabel 2. Pengujian **Function** di mikrokontroler ATmega2560 berdasarkan perintah dari Laptop via *command prompt* di Matlab

No	Function di mikrokontroler ATmega2560 yang dijalankan secara individu di command prompt Matlab	Reaksi gerak robot yang diharapkan	Hasil pengamatan reaksi gerak robot secara keseluruhan	Waktu respon reaksi gerak robot yang diharapkan setelah tombol ditekan (detik)
1	Maju	Bergerak Maju	Robot berhasil bergerak maju	0.5
2	Mundur	Bergerak Mundur	Robot berhasil bergerak mundur	0.5
3	Geser Kanan	Bergeser Kanan	Robot berhasil bergeser ke kanan	0.5
4	Geser Kiri	Bergeser Kiri	Robot berhasil bergeser ke kiri	0.5
5	Maju Serong Kanan	Bergerak Maju Serong Kanan	Robot berhasil bergerak maju serong kanan	0.5
6	Maju Serong Kiri	Bergerak Maju Serong Kiri	Robot berhasil bergerak maju serong kiri	0.5
7	Mundur Serong Kanan	Bergerak Mundur Serong Kanan	Robot berhasil bergerak mundur serong kanan	0.5
8	Mundur Serong Kiri	Bergerak Mundur Serong Kiri	Robot berhasil bergerak mundur serong kiri	0.5
9	Berputar ke Kanan	Berputar ke Kanan	Robot berhasil berputar ke kanan	1
10	Berputar ke Kiri	Berputar ke Kiri	Robot berhasil berputar ke kiri	1
11	Break/Stop roda	Semua motor roda berhenti	Motor dribble telah aktif mengarahkan bola masuk	1
12	Dribble masuk	Robot tidak bergerak, motor dribble bergerak untuk menarik bola ke arah robot	Motor dribble telah aktif mengarahkan bola keluar bersamaan dengan penandang bola aktif mengarah keluar	1.5
13	Tendang bola	Robot tidak bergerak, motor penandang bergerak untuk mendorong bola keluar dari robot	Keempat roda robot berhenti berputar	1.5
			Rerata Waktu Eksekusi (detik)	0.77

Tabel 2 berisi hasil pengujian berbagai fungsi yang dijalankan pada mikrokontroler ATmega2560, berdasarkan perintah yang diberikan via laptop melalui *command prompt* Matlab. Berdasarkan hasil pengujian yang didapatkan maka untuk fungsi motorik Dasar (Maju, Mundur, Geser Kanan, Geser Kiri, Maju Serong Kanan, Maju Serong Kiri, Mundur Serong Kanan, Mundur Serong Kiri, Berputar ke Kanan, Berputar ke Kiri) berhasil menjalankan semua fungsi motorik dasar sesuai dengan perintah dari mikrokontroler ATmega2560. Waktu respon reaksi gerak robot yang diharapkan adalah sekitar 0.5 detik, yang menunjukkan respon yang cepat terhadap perintah. Tabel 2 ini memberikan bukti empiris bahwa mikrokontroler ATmega2560 berhasil dalam mengendalikan robot dan menjalankan berbagai fungsi yang diberikan dengan akurat. Respon yang cepat, sesuai dengan harapan, dan waktu eksekusi yang baik membuat mikrokontroler ini menjadi platform yang kuat untuk pengembangan robotik.

Tabel 3. Pengujian perintah dari *user interface* pada program di laptop ke mikrokontroler

No	Tombol di program diklik	Reaksi gerak robot yang diharapkan	Hasil pengamatan reaksi gerak robot secara keseluruhan	Waktu respon reaksi gerak robot yang diharapkan setelah tombol ditekan
1	Maju	Bergerak Maju	Robot berhasil bergerak maju	1
2	Mundur	Bergerak Mundur	Robot berhasil bergerak mundur	1
3	Geser Kanan	Bergeser Kanan	Robot berhasil bergeser ke kanan	0.5
4	Geser Kiri	Bergeser Kiri	Robot berhasil bergeser ke kiri	0.5
5	Maju Serong Kanan	Bergerak Maju Serong Kanan	Robot berhasil bergerak maju serong kanan	0.5
6	Maju Serong Kiri	Bergerak Maju Serong Kiri	Robot berhasil bergerak maju serong kiri	0.5
7	Mundur Serong Kanan	Bergerak Mundur Serong Kanan	Robot berhasil bergerak mundur serong kanan	0.5
8	Mundur Serong Kiri	Bergerak Mundur Serong Kiri	Robot berhasil bergerak mundur serong kiri	0.5
9	Berputar ke Kanan	Berputar ke Kanan	Robot berhasil berputar ke kanan	1
10	Berputar ke Kiri	Berputar ke Kiri	Robot berhasil berputar ke kiri	1
11	Break/Stop	Semua motor roda berhenti	Motor drible telah aktif mengarahkan bola masuk	1
12	Drible masuk	Robot tidak bergerak, motor drible bergerak untuk menarik bola ke arah robot	Motor drible telah aktif mengarahkan bola keluar bersamaan dengan penandang bola aktif mengarah keluar	1.5
13	Tendang bola	Robot tidak bergerak, motor penandang bergerak untuk mendorong bola keluar dari robot	Keempat roda robot berhenti berputar	1.5
Rerata Waktu Eksekusi (detik)				0.85

Tabel 3 menunjukkan hasil pengujian atas perintah ke robot via GUI yang dibuat. Beberapa poin menunjukkan bahwa waktu respon rata-rata robot terhadap perintah tombol diklik adalah 0.85 detik. Ini adalah indikasi bahwa sistem kontrol robot merespon dengan cepat terhadap perintah. Beberapa fungsi memiliki waktu respon yang lebih lama (1 hingga 1.5 detik) seperti fungsi berputar dan fungsi drible masuk. Ini mungkin disebabkan oleh kompleksitas pergerakan yang melibatkan koordinasi motor yang lebih banyak. Secara keseluruhan, hasil

yang diperoleh ini dapat digunakan sebagai dasar untuk penelitian lebih lanjut dalam pengembangan sistem kontrol robotik dan perbaikan kinerja robot dalam aplikasi yang lebih kompleks.

Tabel 4 menunjukkan berisi hasil pengujian yang melibatkan pengiriman perintah dari laptop pengendali yang terkoneksi secara nirkabel ke user interface pada laptop yang diletakkan pada robot, yang juga terhubung ke mikrokontroler melalui access point 2.4 GHz. Terkait hasil pengamatan di Tabel 4 yaitu bahwa rata-rata waktu respon robot terhadap perintah nirkabel adalah sekitar 1.35 detik. Waktu respon ini sedikit lebih lambat dibandingkan dengan pengujian sebelumnya yang dilakukan secara lokal (1 detik). Hal ini dapat disebabkan oleh keterlambatan dalam mentransmisikan perintah nirkabel dan menerima mereka di robot. Meskipun responnya masih relatif cepat, peningkatan waktu respon perlu diperhitungkan dalam penggunaan aplikasi yang memerlukan respon yang sangat cepat.

Fungsi dribble masuk dan tendang bola memiliki waktu respon yang sedikit lebih lama (2 detik). Waktu respon yang lebih lambat dalam fungsi ini mungkin disebabkan oleh tingkat kompleksitas tugas, termasuk koordinasi antara motor dribble dan motor penendang. Hal ini sesuai dengan ekspektasi dalam tugas yang melibatkan manipulasi objek (bola).

Hasil pengujian menunjukkan kinerja robot ketika dikendalikan secara nirkabel dengan respon yang cepat adalah nilai positif, tetapi peningkatan waktu respon dalam beberapa fungsi harus dipertimbangkan dalam pengembangan dan penggunaan aplikasi robotik yang spesifik. Selain itu, pengujian nirkabel membuka pintu untuk pertimbangan keamanan tambahan dalam pengembangan sistem kontrol.

Tabel 4. Pengujian perintah dari laptop pengendali secara nirkabel ke *user interface* pada laptop yang diletakkan pada robot dan terkoneksi juga ke mikrokontroler dengan *access poin 2.4 Ghz*

No	Tombol di program diklik secara nirkabel dari laptop pengendali	Reaksi gerak robot yang diharapkan	Hasil pengamatan reaksi gerak robot secara keseluruhan	Waktu respon reaksi gerak robot yang diharapkan setelah tombol ditekan oleh laptop kedua via <i>Google Remote Desktop</i>
1	Maju	Bergerak Maju	Robot berhasil bergerak maju	1.5
2	Mundur	Bergerak Mundur	Robot berhasil bergerak mundur	1.5
3	Geser Kanan	Bergeser Kanan	Robot berhasil bergeser ke kanan	1
4	Geser Kiri	Bergeser Kiri	Robot berhasil bergeser ke kiri	1
5	Maju Serong Kanan	Bergerak Maju Serong Kanan	Robot berhasil bergerak maju serong kanan	1
6	Maju Serong Kiri	Bergerak Maju Serong Kiri	Robot berhasil bergerak maju serong kiri	1
7	Mundur Serong Kanan	Bergerak Mundur Serong Kanan	Robot berhasil bergerak mundur serong kanan	1
8	Mundur Serong Kiri	Bergerak Mundur Serong Kiri	Robot berhasil bergerak mundur serong kiri	1
9	Berputar ke Kanan	Berputar ke Kanan	Robot berhasil berputar ke kanan	1.5
10	Berputar ke Kiri	Berputar ke Kiri	Robot berhasil berputar ke kiri	1.5
11	Break/Stop	Semua motor roda berhenti	Motor dribble telah aktif mengarahkan bola masuk	1.5
12	Dribble masuk	Robot tidak bergerak, motor dribble bergerak untuk menarik bola ke arah robot	Motor dribble telah aktif mengarahkan bola keluar bersamaan dengan penendang bola aktif mengarah keluar	2
13	Tendang bola	Robot tidak bergerak, motor penendang bergerak untuk mendorong bola keluar dari robot	Keempat roda robot berhenti berputar	2
Rerata Waktu Eksekusi (detik)				1.35

Tabel 5 menunjukkan hasil pengujian respon robot terhadap perintah yang dikirimkan dari laptop pengendali secara nirkabel melalui *Google Remote Desktop*. Beberapa hal diantaranya adalah terkait respon waktu. Rata-rata waktu respon robot terhadap perintah nirkabel adalah 0.77 detik. Respon ini relatif cepat dan menunjukkan bahwa sistem pengendalian nirkabel memiliki kinerja yang baik dalam mengirim perintah ke robot dan menggerakkan robot sesuai dengan instruksi yang diberikan. Respon yang cepat adalah nilai positif dalam pengendalian robot.

Hasil pengujian juga menunjukkan bahwa robot memberikan respon yang konsisten terhadap berbagai perintah yang diberikan. Konsistensi ini penting karena menunjukkan bahwa robot dapat diandalkan dalam berbagai situasi dan berbagai fungsi dapat dijalankan dengan stabil. Fungsi Dribble Masuk dan Tendang Bola: Fungsi dribble masuk dan tendang bola memerlukan waktu respon yang lebih lama (1.5 detik). Hal ini wajar karena fungsi-fungsi ini melibatkan manipulasi objek (bola) dan koordinasi antara motor dribble dan motor penendang. Meskipun memiliki waktu respon lebih lama, kinerja mereka masih sesuai dengan ekspektasi.

Pengoptimalan selanjutnya terkait hasil pengujian ini dapat menjadi dasar untuk pengoptimalan selanjutnya dalam pengembangan sistem kontrol nirkabel. Pengurangan waktu respon untuk beberapa fungsi yang memerlukan reaksi lebih cepat atau peningkatan koordinasi antara motor dribble dan motor penendang dalam fungsi dribble masuk dan tendang bola bisa menjadi area peningkatan potensial.

Secara keseluruhan, hasil pengujian ini memperlihatkan gambaran positif tentang kinerja robot dalam merespon perintah nirkabel. Waktu respon yang cepat dan konsistensi dalam hasil adalah aspek-aspek yang penting dalam pengendalian robot. Analisis ini dapat digunakan sebagai dasar untuk pengembangan dan pengoptimalan sistem pengendalian nirkabel dalam aplikasi robotik lebih lanjut.

Tabel 5. Pengujian perintah dari laptop pengendali secara nirkabel ke **user interface** pada laptop yang diletakkan pada robot dan terkoneksi juga ke mikrokontroler dengan *access point 5.8 Ghz*

No	Tombol di program diklik secara nirkabel dari laptop pengendali	Reaksi gerak robot yang diharapkan	Hasil pengamatan reaksi gerak robot secara keseluruhan	Waktu respon reaksi gerak robot yang diharapkan setelah tombol ditekan oleh laptop kedua via <i>Google Remote Desktop</i>
1	Maju	Bergerak Maju	Robot berhasil bergerak maju	1
2	Mundur	Bergerak Mundur	Robot berhasil bergerak mundur	1
3	Geser Kanan	Bergeser Kanan	Robot berhasil bergeser ke kanan	0.5
4	Geser Kiri	Bergeser Kiri	Robot berhasil bergeser ke kiri	0.5
5	Maju Serong Kanan	Bergerak Maju Serong Kanan	Robot berhasil bergerak maju serong kanan	0.5
6	Maju Serong Kiri	Bergerak Maju Serong Kiri	Robot berhasil bergerak maju serong kiri	0.5
7	Mundur Serong Kanan	Bergerak Mundur Serong Kanan	Robot berhasil bergerak mundur serong kanan	0.5
8	Mundur Serong Kiri	Bergerak Mundur Serong Kiri	Robot berhasil bergerak mundur serong kiri	0.5
9	Berputar ke Kanan	Berputar ke Kanan	Robot berhasil berputar ke kanan	1
10	Berputar ke Kiri	Berputar ke Kiri	Robot berhasil berputar ke kiri	1
11	Break/Stop	Semua motor roda berhenti	Motor dribble telah aktif mengarahkan bola masuk	1

12	Drible masuk	Robot tidak bergerak, motor drible bergerak untuk menarik bola ke arah robot	Motor drible telah aktif mengarahkan bola keluar bersamaan dengan penendang bola aktif mengarah keluar	1.5
13	Tendang bola	Robot tidak bergerak, motor penendang bergerak untuk mendorong bola keluar dari robot	Keempat roda robot berhenti berputar	1.5
			Rerata Waktu Eksekusi (detik)	0.77

Kemudian jika dibandingkan, analisis antara Tabel 3 dan Tabel 5 yang keduanya berisi i pengujian perintah dari laptop pengendali secara nirkabel dan melibatkan pengujian perintah dari laptop pengendali secara lokal. Berikut adalah perbandingan analisis kunci antara bahwa pada Tabel 3 pengujian perintah dari user interface dari komputer lokal (di atas robot) menunjukkan rata-rata waktu respon robot terhadap perintah adalah 0.85 detik. Reaksi gerak robot dalam beberapa fungsi seperti "Drible masuk" dan "Tendang bola" membutuhkan waktu respon yang lebih lama (1.5 detik). Waktu respon rata-rata yang cepat adalah nilai positif dalam pengujian ini. Kemudian Tabel 5 pengujian perintah dari user interface secara nirkabel dengan access point 5 GHz dengan rata-rata waktu respon robot terhadap perintah adalah 0.77 detik. Reaksi gerak robot dalam beberapa fungsi seperti "Drible masuk" dan "Tendang bola" membutuhkan waktu respon yang lebih lama (1.5 detik). Sedangkan waktu respon rata-rata yang lebih cepat (0.77 detik) dibandingkan dengan pengujian sebelumnya (0.85 detik).

Perbandingan waktu respon kedua tabel menunjukkan waktu respon yang lebih cepat secara keseluruhan (0.77 detik) dibandingkan dengan tabel pertama (0.85 detik). Hal ini menunjukkan bahwa pengendalian nirkabel dalam tabel kedua memiliki kinerja yang sedikit lebih responif. Namun, perbedaannya tidak signifikan.

Secara keseluruhan, kedua tabel pengujian menunjukkan kinerja yang baik dalam pengendalian robot, dengan perbedaan waktu respon yang cukup kecil antara pengujian lokal dan pengujian nirkabel. Pengembangan berikutnya perlu mempertimbangkan kecepatan respon yang sesuai dengan kebutuhan aplikasi robotik tertentu dan terus melakukan pengoptimalan jika diperlukan.

Jika dibandingkan dengan penelitian sebelumnya yang memiliki respon gerak robot atas perintah dari remote controller maksimal 0.988 detik [18], maka rerata respon ketika robot pada penelitian ini dioperasikan adalah 0.85 detik dengan wifi access point 2.4 GHz dan 0.77 detik ketika menggunakan wifi access point 5 GHz.

5. Simpulan

Hasil penelitian menunjukkan bahwa pengendalian robot secara nirkabel melalui Google Remote Desktop memiliki kinerja yang baik, dengan rata-rata waktu respon sekitar 0.77 detik. Adapun penelitian serupa [18] memiliki respon gerak robot atas perintah dari *remote controller* maksimal 0.988 detik. Kemudian pada penelitian ini pula, konsistensi dalam hasil dan fungsi-fungsi robot yang berjalan dengan baik juga diamati. Meskipun beberapa fungsi memerlukan waktu respon lebih lama (1.5 detik), hasil ini positif dalam konteks aplikasi robotik. Perbandingan dengan pengendalian lokal menunjukkan sedikit peningkatan dalam waktu respon nirkabel, dan penelitian ini memberikan dasar untuk pengembangan dan pengoptimalan sistem pengendalian nirkabel di masa depan.

Daftar Referensi

- [1] Muliady and G. Arisandy, "Implementasi Sistem Gerak Holonomic Pada Robot Krsbi Beroda 2017 Implementation Of Holonomic Motion In Indonesian Soccer Wheeled Robot Contest 2017 Pada Kontes Robot Indonesia 2017 divisi Kontes Robot Sepak Bola Indonesia," *J. Tek. dan Ilmu Komput.*, vol. 7, no. 25, pp. 9–25, 2017.
- [2] F. Tajti, G. Szayer, B. Kovács, and P. Korondi, "Robot base with holonomic drive," in *IFAC Proceedings Volumes (IFAC-PapersOnline)*, 2014, vol. 19, no. 3, pp. 5715–5720, doi: 10.3182/20140824-6-za-1003.00785.
- [3] B. Rahmani, H. Aprilianto, P. T. Informatika, and S. Banjarbaru, "Model Kendali Multi Kontroler untuk Robot Humanoid dengan 19 Derajat Kebebasan," in *Seminar Nasional*

- KNS&1 STIKOM Bali*, 2015, pp. 9–10.
- [4] B. Rahmani, A. Harjoko, T. K. Priyambodo, and H. Aprilianto, "Early Model of Vision-Based Obstacle Mapping Utilizing Grid-Edge-Depth Map," *Int. J. Eng. Adv. Technol.*, vol. 9, no. 2, pp. 4519–4523, 2019, doi: 10.35940/ijeat.B4550.129219.
- [5] R. N. Handika and B. Rahmani, "Sistem Deteksi Arah Gerak Bola Menggunakan Metode Optical Flow pada Robot Goal Keeper Beroda," *Progresif J. Ilm. Komput.*, vol. 19, no. 2, pp. 623–635, 2023, [Online]. Available: <http://ojs.stmik-banjarbaru.ac.id/index.php/progresif/article/view/1455/764>.
- [6] B. Rahmani, A. Harjoko, and T. K. Priyambodo, "A vision-based real-time obstacle avoidance's rules utilising grid-edge-depth map," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 19, no. 1, pp. 513–525, 2020, doi: 10.11591/ijeecs.v19.i1.pp513-525.
- [7] M. N. Hifzi and B. Rahmani, "Model Kalkulasi Jarak pada Proses Deteksi Bola Menggunakan Metode Color-Block Detection," *Jutisi J. Ilm. Tek. Inform. dan Sist. Inf.*, vol. 12, no. 2, pp. 394–404, 2023, doi: 10.35889/jutisi.v12i2.1230.
- [8] C. Zhu, G. Giorgi, Y. H. Lee, and C. Gunther, "Enhancing accuracy in visual SLAM by tightly coupling sparse ranging measurements between two rovers," *2018 IEEE/ION Position, Locat. Navig. Symp. PLANS 2018 - Proc.*, pp. 440–446, 2018, doi: 10.1109/PLANS.2018.8373412.
- [9] S. Mushtaq, G. Alandjani, S. F. Abbasi, and N. Abosaq, "Hybrid Geo-Location Routing Protocol for Indoor and Outdoor Positioning Applications," *Int. J. Adv. Res. Artif. Intell.*, vol. 10, no. 7, pp. 1–7, 2019.
- [10] P. P. Ray, "A survey on Internet of Things architectures," *J. King Saud Univ. - Comput. Inf. Sci.*, vol. 30, no. 3, pp. 291–319, 2018, doi: 10.1016/j.jksuci.2016.10.003.
- [11] T. Saarikko, U. H. Westergren, and T. Blomquist, "The Internet of Things : Are you ready for what ' s coming?," *Bus. Horiz.*, vol. 60, no. 5, pp. 667–676, 2020, doi: 10.1016/j.bushor.2017.05.010.
- [12] J. S. P. Siy, R. K. C. Chan, and R. G. Baldovino, "Implementation of a real-time appearance-based mapping in a fully autonomous urban search robot," *2018 IEEE 10th Int. Conf. Humanoid, Nanotechnology, Inf. Technol. Commun. Control. Environ. Manag. HNICEM 2018*, pp. 1–4, 2019, doi: 10.1109/HNICEM.2018.8666380.
- [13] K. Pendidikan *et al.*, "Kementerian pendidikan, kebudayaan, riset, dan teknologi," 2023.
- [14] R. A. Fatekha, B. S. B. Dewantara, and H. Oktavianto, "Sistem Deteksi Bola pada Robot Kiper Pemain Sepakbola Beroda," *J. Integr.*, vol. 13, no. 2, pp. 127–134, 2021, doi: 10.30871/ji.v13i2.3133.
- [15] T. Cieslewski, S. Lynen, M. Dymczyk, S. Magnenat, and R. Siegwart, "Map API - Scalable Decentralized Map Building for Robots," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, 2015, pp. 6241–6247.
- [16] A. U. Haque and A. Nejadpak, "Obstacle Avoidance Using Stereo Camera," *Comput. Vis. Pattern Recognit.*, 2017, [Online]. Available: <http://arxiv.org/abs/1705.04114>.
- [17] B. Rahmani, A. Harjoko, and T. K. Priyambodo, "Grid-edge-depth map building employing sad with sobel edge detector," *Int. J. Smart Sens. Intell. Syst.*, vol. 10, no. 3, pp. 551–566, 2017, doi: 10.21307/ijssis-2017-223.
- [18] I. Hafidz, D. Adiputra, B. Montolalu, W. A. Prastyabudi, H. Widyantara, and M. A. Afandi, "Robot Logistik berbasis IoT untuk Memonitoring Pasien dan Pengiriman Logistik Selama Pandemi COVID-19," *J. Nas. Tek. Elektro*, vol. 9, no. 3, pp. 182–188, 2020.
- [19] M. Pushpavalli, L. N. Manikanta, P. Sivagami, P. Saikumar, P. Abirami, and M. V. Reddy, "Smart Home Automation Controlled by Robot using Matlab and Arduino," no. 6, pp. 1181–1184, 2019, doi: 10.35940/ijeat.F1198.0986S319.
- [20] S. Singh and E. Singla, "Realization of task-based designs involving DH parameters: a modular approach," *Intell. Serv. Robot.*, vol. 9, no. 3, pp. 289–296, 2016, doi: 10.1007/s11370-015-0186-x.
- [21] M. Papoutsidakis, "Intelligent Design and Algorithms to Control a Stereoscopic Camera on a Robotic Workspace Z-error Z-Error," *Int. J. Comput. Appl.*, vol. 167, no. 12, pp. 32–35, 2017.
- [22] Pusat Prestasi Nasional, "Buku Pedoman Kontes Robot Indonesia (Kri) Tahun 2023," 2023.