

**KONTROL OTOMASI *ROBOT SOCCER* MENGGUNAKAN
METODE *HOUGH CIRCLE TRANSFORM***

SKRIPSI

Oleh :
FUAIDIL IKHROM
NIM. 210605110077



**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2025**

**KONTROL OTOMASI *ROBOT SOCCER* MENGGUNAKAN
METODE *HOUGH CIRCLE TRANSFORM***

SKRIPSI

Diajukan kepada:
Universitas Islam Negeri Maulana Malik Ibrahim Malang
Untuk memenuhi Salah Satu Persyaratan dalam
Memperoleh Gelar Sarjana Komputer (S.Kom)

Oleh :
FUAIDIL IKHROM
NIM. 210605110077

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2025**

HALAMAN PERSETUJUAN

KONTROL OTOMASI *ROBOT SOCCER* MENGGUNAKAN METODE *HOUGH CIRCLE TRANSFORM*

SKRIPSI

Oleh :
FUAIDIL IKHROM
NIM. 210605110077

Telah Diperiksa dan Disetujui untuk Diuji:
Tanggal: 30 April 2025

Pembimbing I,



Shoffin Nahwa Utama, M.T
NIP. 19860703 202012 1 003

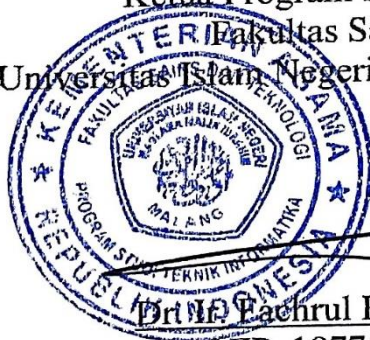
Pembimbing II,



Dr. Cahyo Crysdian, M.Cs
NIP. 19740424 200901 1 008

Mengetahui,

Ketua Program Studi Teknik Informatika
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang



Dr. Faehrul Kurniawan, M.MT., IPU
NIP. 19771020 200912 1 001

HALAMAN PENGESAHAN

KONTROL OTOMASI *ROBOT SOCCER* MENGGUNAKAN METODE *HOUGH CIRCLE TRANSFORM*

SKRIPSI

Oleh :
FUAIDIL IKHROM
NIM. 210605110077

Telah Dipertahankan di Depan Dewan Penguji Skripsi
dan Dinyatakan Diterima Sebagai Salah Satu Persyaratan
Untuk Memperoleh Gelar Sarjana Komputer (S.Kom)
Tanggal: 20 Mei 2025

Susunan Dewan Penguji

Ketua Penguji : Dr. Yunifa Miftachul Arif, M.T
NIP. 19830616 201101 1 004

Anggota Penguji I : Ajib Hanani, M.T
NIP. 19840731 202321 1 013

Anggota Penguji II : Shoffin Nahwa Utama, M.T
NIP. 19860703 202012 1 003

Anggota Penguji III : Dr. Cahyo Crysdian, M.Cs
NIP. 19740424 200901 1 008

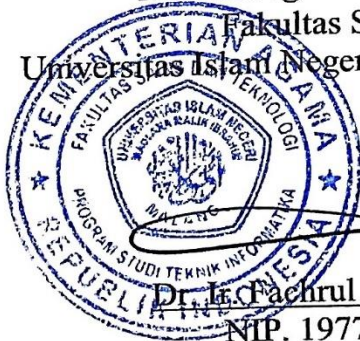
()

()

()

()

Mengetahui dan Mengesahkan,
Ketua Program Studi Teknik Informatika
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang




Dr. Fachrul Kurniawan, M.MT., IPU
NIP. 19771020 200912 1 001

PERNYATAAN KEASLIAN TULISAN

Saya yang bertanda tangan di bawah ini:

Nama : Fuaidil Ikhrom
NIM : 210605110077
Fakultas / Program Studi : Sains dan Teknologi / Teknik Informatika
Judul Skripsi : Kontrol Otomasi *Robot Soccer* Menggunakan
Metode *Hough Circle Transform*

Menyatakan dengan sebenarnya bahwa Skripsi yang saya tulis ini benar-benar merupakan hasil karya saya sendiri, bukan merupakan pengambil alihan data, tulisan, atau pikiran orang lain yang saya akui sebagai hasil tulisan atau pikiran saya sendiri, kecuali dengan mencantumkan sumber cuplikan pada daftar pustaka.

Apabila dikemudian hari terbukti atau dapat dibuktikan skripsi ini merupakan hasil jiplakan, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Malang, 20 Mei 2025

Yang membuat pernyataan,



Fuaidil Ikhrom

NIM. 210605110077

MOTTO

الْيَقِينُ لَا يُزَالُ بِالشَّكِّ

“Keyakinan tidak akan hilang dengan keraguan.”

“If what you love doesn't happen, then love what happens.”

HALAMAN PERSEMBAHAN

Dengan penuh rasa syukur kepada Allah SWT atas segala rahmat dan kemudahan-Nya sehingga skripsi ini dapat terselesaikan dengan baik. Tak lupa, shalawat dan salam selalu tercurah kepada Nabi Muhammad SAW. Karya ini penulis persembahkan dengan hati yang penuh kebahagiaan kepada:

Ibunda dan ayahanda tercinta, Indah Nasirah Hidayati dan Asnada Ary Sukamto yang senantiasa memberikan kasih sayang, doa yang tulus, dukungan baik materi maupun non materi, dan nasehat yang tiada henti.

Teruntuk saudaraku, Hangga Sodiq dan Syuhada

Terima kasih atas saran, semangat, dan dukungan tak henti kepada penulis.

Teman-teman seperjuangan,

Khususnya keluarga besar Teknik Informatika Angkatan 2021 dan KKM 83, terima kasih telah berbagi ilmu, semangat yang menyala, dan pengalaman hidup

KATA PENGANTAR

Assalamu'alaikum Warahmatullahi Wabarakatuh

Alhamdulillah segala puji dan Syukur senantiasa penulis panjatkan pada Allah subhanahu wa ta'ala atas berkat Rahmat, serta hidayah-Nya, sehingga penulis dapat menyelesaikan skripsi ini dengan baik. Sholawat serta salam tetap tercurahkan kepada Nabi Muhammad SAW. Dan semoga kita semua mendapat syafaatnya di hari akhir kelak, Aamiin.

Penulis mengucapkan rasa terima kasih yang begitu besar kepada seluruh pihak yang memberikan dukungan dan motivasi kepada penulis sehingga dapat menyelesaikan skripsi ini. Ucapan terima kasih ini penulis disampaikan kepada:

1. Prof. Dr. M. Zainuddin, M.A., selaku rektor Universitas Islam Negeri Maulana Malik Ibrahim Malang.
2. Prof. Dr. Hj. Sri Harini, M.Si., selaku dekan Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang.
3. Dr. Ir. Fachrul Kurniawan, M.MT., IPU selaku Ketua Program Studi Teknik Informatika Universitas Islam Negeri Maulana Malik Ibrahim Malang.
4. Shoffin Nahwa Utama, M.T., selaku dosen pembimbing 1 serta mentor yang telah memberikan bimbingan, saran, kesabaran dan dukungan yang sangat berarti bagi penulis selama proses penelitian.
5. Dr. Cahyo Crysdian, M.Cs., selaku dosen pembimbing 2, atas arahan dan pencerahan yang sangat membantu dalam menyempurnakan karya ini.

6. Dr. Yunifa Miftachul Arif, M.T., selaku dosen Penguji I dan Bapak Ajib Hanani, M.T., selaku dosen Penguji II yang telah memberikan kritik dan saran yang membangun dalam penyusunan skripsi ini.
7. Seluruh staf dan dosen Program Studi Teknik Informatika yang telah memberikan ilmu, dukungan, dan fasilitas kepada penulis selama masa studi.
8. Kedua orang tua serta saudara-saudari penulis, Indah Nasirah Hidayati, Asnada Ary Sukamto, Hangga Sodiq Asyhari, Syuhada Haga Pratawa, Anisha Ayuningtryas, Khairunisa Dewi Ningrum yang selalu memberikan semangat untuk terus berusaha, dukungan tiada henti, serta doa-doa terbaik sehingga skripsi ini dapat selesai dengan baik. Semoga Allah senantiasa memberikan kesehatan dan lindungan, sehingga dapat selalu berada di setiap perjalanan dan pencapaian penulis.
9. Teman seperjuangan dan sepembimbingan, yang beranggotakan Amirul, Asrul, Rizal, dan khususnya Muhammad Haikal Akbar yang telah membantu banyak dalam proses pengerjaan skripsi ini, semoga selalu diberikan kesehatan.
10. Rekan seperjuangan “Kontrakan Pria Tampan”, yang beranggotakan Rehan, Haikal, Noviansyah, Gianda, Safril, Musa, Khalif, Zufar, Rafi, Al, Fauzil, Amirul, Daffa, dan Dika. Terima kasih telah menemani hari-hari penulis dan terima kasih sudah mau menemani suka duka serta berbagi tawa dan air mata. Sukses untuk kalian semua.

11. Teman-teman seperjuangan “Basecamp Cumlaude” sejak zaman ma’had, yang beranggotakan Charles, Irham, dan Asrul. Terima kasih telah menemani penulis sejak awal kuliah hingga dapat menyelesaikan skripsi ini. Terima kasih atas canda tawa serta pengalaman yang kalian berikan, semoga kalian selalu dalam lindungan Allah dan sehat selalu.
12. Seluruh saudara Teknik Informatika terkhusus Angkatan 2021 “ASTER”, atas segala ilmu, semangat, dan pengalaman berharga yang telah dibagikan. Semoga silaturahmi kita semakin erat dan kita semua dapat meraih impian kita semua.
13. Teman-teman KKM 83 Anagata, terima kasih atas kenangan selama melakukan Kuliah Kerja Mahasiswa di Desa Ngadas, Kabupaten Malang, Jawa Timur. Terima kasih atas pengalaman hidup yang kalian berikan selama masa KKM.
14. Seluruh pihak yang telah terlibat, baik secara langsung maupun tidak langsung dari awal perkuliahan hingga akhir penulisan skripsi ini

Penulis berharap semoga skripsi ini dapat memberikan manfaat untuk kedepannya. Penulis menyadari bahwa skripsi ini masih. Oleh karena itu, menerima masukan yang dapat membantu pengembangan lebih lanjut.

Wassalamu’alaikum warahmatullahi wabarakatuh.

Malang, 20 Mei 2025

Penulis

DAFTAR ISI

HALAMAN PERSETUJUAN	Error! Bookmark not defined.
HALAMAN PENGESAHAN	iv
MOTTO	vi
HALAMAN PERSEMBAHAN	vii
KATA PENGANTAR	viii
DAFTAR ISI	xi
DAFTAR GAMBAR	xiii
DAFTAR TABEL	xiv
ABSTRAK	xv
ABSTRACT	xvi
المخلص	xvii
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Pernyataan Masalah	5
1.3 Tujuan Penelitian	5
1.4 Batasan Masalah	5
1.5 Manfaat Penelitian	5
BAB II STUDI PUSTAKA	7
BAB III DESAIN DAN IMPLEMENTASI	16
3.1 Desain Sistem.....	16
3.2 <i>Preprocessing</i>	17
3.2.1 Konversi <i>Grayscale</i>	18
3.2.2 <i>Gaussian Blur</i>	19
3.2.3 <i>Edge Detection</i>	21
3.3 <i>Hough Circle Transform</i>	24
3.3.1 Konsep <i>Hough Circle Transform</i>	24
3.3.2 Implementasi <i>Hough Circle Transform</i>	28
3.4 Integrasi Sistem Kontrol Robot	32
3.5 Implementasi Sistem.....	36
BAB IV UJI COBA DAN PEMBAHASAN	39
4.1 Skenario Uji	39
4.1.1 Skenario Pengujian <i>Error</i> Deteksi Bola	39
4.1.2 Skenario Pengujian <i>Error</i> Posisi Bola	40
4.1.3 Skenario Pengujian Waktu Respons Robot	42
4.2 Hasil Pengujian	42
4.2.1 Pengujian <i>Error</i> Deteksi Bola.....	43
4.2.2 Pengujian <i>Error</i> Posisi Bola	46
4.2.3 Pengujian Waktu Respons Robot	51
4.3 Pembahasan.....	54
BAB V KESIMPULAN DAN SARAN	61
5.1 Kesimpulan	61
5.2 Saran	61
DAFTAR PUSTAKA	

LAMPIRAN

DAFTAR GAMBAR

Gambar 3. 1 Desain Sistem.....	17
Gambar 3. 2 Hasil <i>Grayscale</i>	19
Gambar 3. 3 <i>Gaussian Blur</i>	21
Gambar 3. 4 Hasil <i>Canny Edge Detection</i>	24
Gambar 3. 5 Ilustrasi Koordinat.....	25
Gambar 3. 6 <i>Flowchart Hough Circle Transform</i>	28
Gambar 3. 7 Hasil Deteksi Lingkaran dan Jarak.....	32
Gambar 3. 8 <i>Flowchart</i> Sistem Kontrol Robot	34
Gambar 3. 9 Realisasi Desain Robot.....	38
Gambar 4. 1 Grafik <i>Error</i> Deteksi	46
Gambar 4. 2 Grafik <i>Error</i> Jarak Bola	49
Gambar 4. 3 Visualisasi Jarak Bola	50
Gambar 4. 4 Grafik Waktu Respons Robot	54

DAFTAR TABEL

Tabel 2. 1 Perbandingan dengan Penelitian Terdahulu.....	13
Tabel 3. 1 Sinkronisasi PWM	35
Tabel 4. 1 Perhitungan <i>Error</i> Deteksi Bola	44
Tabel 4. 2 Perhitungan <i>Error</i> Jarak.....	46
Tabel 4. 3 Perhitungan Waktu <i>Respons</i> Robot.....	51

ABSTRAK

Ikhrom, Fuaidil. 2025. **Kontrol Otomasi Robot Soccer Menggunakan Metode Hough Circle Transform**. Skripsi. Program Studi Teknik Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang. Pembimbing: (I) Shoffin Nahwa Utama, M.T (II) Dr. Cahyo Crysdiyan, M.Cs.

Kata Kunci: *Computer Vision, Deteksi Bola, Hough Circle Transform, Kontrol Otomasi, Robot Soccer.*

Penelitian ini bertujuan untuk mengembangkan sistem kontrol otomasi pada robot soccer yang mampu mendeteksi dan merespons bola secara otomatis menggunakan metode *Hough Circle Transform*. Permasalahan utama yang diangkat adalah bagaimana sistem dapat mendeteksi objek berbentuk lingkaran (bola), menghitung estimasi jarak bola terhadap robot, dan mengambil tindakan pergerakan hingga aksi menendang bola tanpa intervensi manusia. Sistem dikembangkan dengan memanfaatkan *Raspberry Pi* dan *Raspberry Pi Camera* sebagai komponen utama dalam pengambilan dan pemrosesan citra digital yang dilanjutkan dengan tahapan *preprocessing*, deteksi tepi (*Canny Edge Detection*), dan implementasi metode *Hough Circle Transform*. Hasil pengujian menunjukkan bahwa sistem memiliki performa deteksi bola yang sangat baik dengan rata-rata *error* deteksi sebesar 10%. Sistem mampu mengestimasi jarak bola terhadap robot dengan rata-rata *error* sebesar 3,22%. Rata-rata waktu respons sistem terhadap deteksi bola yaitu 1,33 detik yang menunjukkan bahwa sistem cukup responsif dalam merespons stimulus visual secara *real-time*. Sistem juga berhasil mengaktifkan solenoid untuk menendang bola ketika bola berada pada jarak yang sangat dekat (≤ 15 cm), menandakan keberhasilan integrasi antara pengolahan citra dan kontrol aktuator. Secara keseluruhan, penelitian ini menunjukkan bahwa metode *Hough Circle Transform* dapat diimplementasikan untuk sistem kontrol robot berbasis *computer vision*, serta mendemonstrasikan potensi penerapan teknologi otomasi dalam bidang robotika.

ABSTRACT

Ikhrom, Fuaidil. 2025. **Automated Control of Soccer Robots Using the Hough Circle Transform Method**. Thesis. Informatics Engineering Faculty of Science and Technology Universitas Islam Negeri Maulana Malik Ibrahim Malang. Advisors: (I) Shoffin Nahwa Utama, M.T (II) Dr. Cahyo Crysdian, M.Cs.

Key words: Automated Control, Ball Detection, Computer Vision, Hough Circle Transform, Soccer Robot.

This research aims to develop an automation control system for soccer robots that can detect and respond to the ball automatically using the Hough Circle Transform method. The main problem raised is how the system can detect circular objects (balls), calculate the estimated distance of the ball to the robot, and take movement actions up to the action of kicking the ball without human intervention. The system is developed by utilizing Raspberry Pi and Raspberry Pi Camera as the main components in digital image capture and processing followed by preprocessing, edge detection (Canny Edge Detection), and implementation of the Hough Circle Transform method. The test results show that the system has excellent ball detection performance with an average detection error of 10%. The system is able to estimate the distance of the ball to the robot with an average error of 3.22%. The average response time of the system to ball detection is 1.33 seconds which shows that the system is quite responsive in responding to real-time visual stimulus. The system also successfully activates the solenoid to kick the ball when the ball is at a very close distance (≤ 15 cm), signifying the successful integration between image processing and actuator control. Overall, this research shows that the Hough Circle Transform method can be implemented for computer vision-based robot control systems, and demonstrates the potential application of automation technology in robotics.

الملخص

إكرام، فوعيدل. 2025. التحكم الآلي في روبوتات كرة القدم باستخدام طريقة تحويل دائرة هوغ. أطروحة. كلية هندسة المعلوماتية، كلية العلوم والتكنولوجيا، جامعة مولانا مالك إبراهيم الإسلامية الحكومية مالانج. المشرف الأول: شوفين ناهوا أوتاما، الماجستير. المشرف الثاني: كاهيو كريسدیان، الماجستير.

الكلمات الرئيسية: التحكم الآلي، كشف الكرة، الرؤية الحاسوبية، تحويل دائرة هوغ، روبوت كرة القدم

يهدف هذا البحث إلى تطوير نظام تحكم آلي لروبوتات كرة القدم يمكنه اكتشاف الكرات والاستجابة لها تلقائيًا باستخدام طريقة Hough Circle Transform. المشكلة الرئيسية التي يتناولها البحث هي كيفية قيام النظام باكتشاف الأجسام الدائرية (الكرات)، وحساب المسافة التقديرية بين الكرة والروبوت، واتخاذ الإجراء اللازم لكل الكرة دون تدخل بشري. تم تطوير النظام باستخدام Raspberry Pi Camera و Raspberry Pi كمكونات رئيسية لالتقاط ومعالجة الصور الرقمية، تليها المعالجة المسبقة، وكشف الحواف (Canny Edge Detection)، وتنفيذ خوارزمية Hough Circle Transform. تظهر نتائج الاختبار أن النظام يتمتع بأداء ممتاز في كشف الكرة بمتوسط خطأ كشف يبلغ 10%. بالإضافة إلى ذلك، يمكن للنظام تقدير المسافة بين الكرة والروبوت بمتوسط خطأ يبلغ 3.22%. يبلغ متوسط وقت استجابة النظام لاكتشاف الكرة 1.33 ثانية، مما يشير إلى أن النظام يستجيب بشكل كافٍ في الاستجابة الحسية البصرية في الوقت الفعلي. كما نجح النظام في تنشيط الملف اللولبي لكل الكرة عندما كانت الكرة على مسافة قريبة جدًا ($z \geq 15$)، مما يشير إلى التكامل الناجح بين معالجة الصور والتحكم في المشغل. بشكل عام، توضح هذه الدراسة أن طريقة Hough Circle Transform يمكن تطبيقها على أنظمة التحكم في الروبوتات القائمة على الرؤية الحاسوبية، وتظهر إمكانات تطبيقات تكنولوجيا الأتمتة في مجال الروبوتات.

BAB I

PENDAHULUAN

1.1 Latar Belakang

Perkembangan teknologi robotika dalam beberapa tahun terakhir telah mengalami peningkatan secara signifikan. Menurut *International Federation of Robotics*, stok robot operasional di seluruh dunia mencapai rekor baru sekitar 3,9 juta unit (*International Federation of Robotics*, 2024). Di Indonesia, adopsi teknologi robotika juga menunjukkan tren positif dengan diadakannya Kontes Robot Indonesia (KRI). Kontes Robot Indonesia (KRI) adalah kegiatan kompetisi tahunan mahasiswa dalam bidang robotika yang diikuti oleh seluruh mahasiswa perguruan tinggi di Indonesia (*Kontes Robot Indonesia*, 2024). Dalam konteks pengembangan sistem kontrol robot, Arif (2013) berhasil mengimplementasikan sistem kontrol *hexapod* robot MSR-H01 menggunakan mikrokontroler ATmega 128 dan sensor ultrasonik. Robot tersebut mampu beradaptasi pada medan datar, berkarpet, dan bergelombang dengan kecepatan bervariasi, tergantung mode kontrol yang digunakan. Pendekatan ini memiliki keterbatasan, terutama dalam hal respons terhadap objek bergerak dinamis seperti bola pada permainan *robot soccer*.

Dalam konteks pendidikan tinggi, khususnya di Jurusan Teknik Informatika UIN Maliki Malang, pengembangan robotika menjadi salah satu fokus untuk menghasilkan lulusan yang inovatif. Ontaki, sebuah komunitas otomasi dan robotika yang telah berdiri selama satu dekade, telah berperan aktif dalam pengembangan ini dan berpartisipasi dalam berbagai kompetisi robotika tingkat

nasional dan internasional. Namun, terdapat kesenjangan yang signifikan antara keahlian praktis yang dimiliki komunitas dan pengembangan ilmiah di lingkungan akademis. Berdasarkan repositori skripsi di *website thesis* UIN Maliki Malang dalam rentang waktu 5 tahun terakhir hanya terdapat 4 penelitian skripsi yang berfokus pada robotika, merepresentasikan kurang dari 1% dari total skripsi. Hal ini mengindikasikan adanya urgensi untuk menjembatani gap antara praktik dan teori, sekaligus meningkatkan minat mahasiswa terhadap penelitian di bidang robotika.

Salah satu bidang robotika yang menarik untuk diteliti lebih lanjut adalah *robot soccer*, yang menggabungkan unsur olahraga dan teknologi robotika. Kompetisi *robot soccer* seperti Kontes Robot Cerdas Indonesia (KRCI) dan *Technocorner* UGM yang diadakan setiap tahun telah menjadi ajang lomba yang mendorong inovasi di bidang sistem kontrol robot. *Technocorner* merupakan kompetisi tahunan dalam bidang robotika yang diselenggarakan oleh Universitas Gadjah Mada (UGM) untuk tingkat pelajar dan mahasiswa, dengan salah satu cabang perlombaannya adalah *robot soccer*. Dalam perlombaan ini, peserta dihadapkan pada tantangan untuk merancang robot yang dapat bermain sepak bola. Kontrol robot dalam perlombaan *robot soccer* di *Technocorner* saat ini masih menggunakan *remote*. Penggunaan *remote* untuk mengendalikan robot masih menjadi kendala dalam meningkatkan performa robot dalam pertandingan, terutama dalam hal responsivitas. Solusi untuk masalah ini adalah dengan mengintegrasikan *computer vision* supaya robot dapat bergerak secara otomatis tanpa intervensi manusia. Dengan memanfaatkan teknologi *computer vision*, robot

dapat mendeteksi dan mengikuti pergerakan bola yang akan meningkatkan kinerja robot dalam pertandingan.

Teknologi *computer vision*, khususnya *image processing*, memberikan solusi untuk masalah ini. Menurut studi yang dilakukan oleh Arifin (2022), implementasi algoritma *computer vision* yang efisien dapat meningkatkan akurasi deteksi bola hingga 95% dan mengurangi latensi sistem hingga 40% dibandingkan dengan metode konvensional. Implementasi *computer vision* dalam robotika telah terbukti meningkatkan kemampuan robot dalam berbagai aplikasi. Yun dkk. (2013) mendemonstrasikan bahwa *computer vision* memungkinkan robot untuk menginterpretasikan informasi visual dari lingkungan sekitarnya dengan tingkat akurasi yang mendekati kemampuan manusia. Di industri manufaktur, robot dengan kemampuan visual dapat melakukan inspeksi kualitas produk secara otomatis. Sementara itu, dalam interaksi manusia-robot, sistem pengenalan wajah dan *gesture* berbasis *computer vision* telah menunjukkan peningkatan efektivitas komunikasi hingga 60% dibandingkan dengan sistem tanpa kemampuan visual (Mckenna dkk., 2024).

Salah satu metode *computer vision* untuk deteksi bola pada *robot soccer* adalah *Hough Circle Transform*. Teknik ini memiliki kelebihan dalam mengenali lingkaran dengan berbagai ukuran dan posisi, bahkan dalam kondisi *noise* atau oklusi parsial (Wibowo dkk., 2021). Studi yang dilakukan oleh Putra & Puriyanto (2022) menunjukkan bahwa implementasi *Hough Circle Transform* dalam sistem kontrol *robot soccer* dapat meningkatkan akurasi deteksi bola hingga 92% dan mengurangi waktu komputasi sebesar 35% dibandingkan dengan metode *threshold*

warna konvensional. Implementasi *Hough Circle Transform* dalam sistem kontrol *robot soccer* melibatkan beberapa tahapan, mulai dari pengambilan citra oleh kamera *Raspberry Pi*, pengurangan *noise*, deteksi lingkaran yang mewakili bola, hingga pengiriman informasi ke sistem kontrol robot.

Upaya pengembangan teknologi robotika ini tidak hanya bertujuan untuk menyelesaikan permasalahan teknis dalam kontrol otomasi *robot soccer*, tetapi juga mencerminkan keimanan dan keyakinan kepada Allah yang mengatur seluruh makhluk-Nya. Sebagaimana firman Allah dalam Surah At-Tagabun ayat 11:

مَا أَصَابَ مِنْ مُصِيبَةٍ إِلَّا بِإِذْنِ اللَّهِ ۗ وَمَنْ يُؤْمِن بِاللَّهِ يَهْدِ اللَّهُ قَلْبَهُ ۚ وَاللَّهُ بِكُلِّ شَيْءٍ عَلِيمٌ

"Tidak ada sesuatu musibah yang menimpa seseorang kecuali dengan izin Allah; barang siapa beriman kepada Allah, niscaya Dia akan memberi petunjuk kepada hatinya. Dan Allah Maha Mengetahui segala sesuatu" (QS. At-Tagabun: 11).

Menurut tafsir dari Kemenag, ayat ini menjelaskan bahwa Allah tidak hanya menciptakan makhluk, tetapi juga mengatur seluruh kehidupan mereka. Segala sesuatu yang terjadi, baik keberhasilan maupun tantangan, merupakan bagian dari izin-Nya. Barangsiapa beriman kepada Allah dengan teguh, Allah akan memberikan petunjuk kepada hatinya, memantapkan keimanannya, dan membimbingnya untuk terus berusaha dengan istiqamah (Kemenag, 2022b).

Penelitian ini tidak hanya berupaya menyelesaikan permasalahan teknis dalam kontrol otomasi *robot soccer*, tetapi juga bertujuan untuk menjembatani kesenjangan antara keahlian praktis komunitas robotika dan pengembangan ilmiah di lingkungan akademis. Melalui integrasi aspek praktis dan ilmiah, penelitian ini diharapkan dapat memberikan kontribusi signifikan bagi perkembangan komunitas Ontaki, Jurusan Teknik Informatika, dan UIN Maliki Malang secara keseluruhan.

1.2 Pernyataan Masalah

Bagaimana performa *Hough Circle Transform* dalam mendeteksi objek dan jarak bola pada *robot soccer*?

1.3 Tujuan Penelitian

Mengukur performa algoritma *Hough Circle Transform* dalam mendeteksi objek dan jarak bola pada *robot soccer*.

1.4 Batasan Masalah

1. Penelitian dilakukan dalam lingkungan yang terkendali, yaitu di dalam ruangan dengan pencahayaan yang stabil dan tanpa gangguan eksternal.
2. Penelitian dilakukan menggunakan bola tenis berwarna hijau dengan diameter 60 mm.
3. Penelitian dilakukan dalam arena yang berukuran 300 cm x 175 cm dan menggunakan latar belakang berwarna putih.

1.5 Manfaat Penelitian

1. Lembaga penelitian teknologi, seperti LIPI (Lembaga Ilmu Pengetahuan Indonesia) atau BPPT (Badan Pengkajian dan Penerapan Teknologi), dapat memanfaatkan hasil penelitian ini sebagai dasar untuk penelitian lebih lanjut dalam pengembangan teknologi otomasi berbasis pengolahan citra.
2. Komunitas robotika di perguruan tinggi yang mengikuti kompetisi seperti *RoboCup* atau Kontes Robot Sepak Bola Indonesia (KRSBI) dapat memanfaatkan hasil penelitian ini untuk meningkatkan performa robot

mereka, terutama dalam hal deteksi bola dan pengambilan keputusan otomatis.

3. Perusahaan yang berfokus pada pengembangan sistem kendali otomatis, seperti PT. *Infineon Technologies* Batam, dapat mengaplikasikan hasil penelitian dalam sistem inspeksi berbasis robot terutama yang membutuhkan deteksi objek berbentuk lingkaran, seperti inspeksi produk dalam proses manufaktur yang menggunakan sensor visual.
4. Perusahaan yang menyediakan perangkat robotika untuk pendidikan, seperti *Makeblock* Indonesia, dapat menerapkan hasil penelitian ini pada produk-produk robot mereka seperti robot-robot yang digunakan untuk pelatihan coding atau kompetisi robotika di tingkat sekolah dan perguruan tinggi.
5. Instansi yang mengadakan kompetisi robotika, seperti *RoboCup*, dapat menggunakan hasil penelitian ini untuk meningkatkan standar dan kualitas kompetisi, serta menginspirasi inovasi baru dalam teknologi *robot soccer*.

BAB II

STUDI PUSTAKA

Winarno dkk. (2020) melakukan penelitian tentang deteksi bola untuk robot sepak bola KRSBI menggunakan *PeleeNet* pada kamera *omnidirectional*. Penelitian mereka menggabungkan kamera *omnidirectional* yang memberikan pandangan 360 derajat dengan kemampuan *deep learning PeleeNet* untuk deteksi objek. Mereka mengatasi masalah distorsi gambar dari kamera *omnidirectional* melalui proses kalibrasi, kemudian melatih ulang *PeleeNet* menggunakan dataset *PASCAL VOC* yang dimodifikasi. Hasil eksperimen menunjukkan *PeleeNet* memiliki akurasi deteksi bola sebesar 83,2%, lebih tinggi dibandingkan model SSD (81,1%) dan *MobileNet-SSD* (80,1%). Para peneliti menyimpulkan bahwa *PeleeNet* menawarkan keseimbangan yang baik antara efisiensi memori, kecepatan, dan akurasi sehingga berpotensi untuk diimplementasikan pada platform *mobile* robot sepak bola KRSBI. Perbedaan dengan penelitian yang akan dilakukan adalah metode dan kamera yang digunakan, yaitu menggunakan metode *Hough Circle Transform* dan *Raspberry Pi Camera*.

Penelitian oleh Risfendra dkk. (2020) membahas tentang sistem pergerakan dan deteksi pada robot sepak bola beroda menggunakan *image processing* dengan penerapan *multivision*. Dalam penelitian tersebut, mereka menggunakan dua kamera yaitu kamera utama untuk deteksi objek di depan robot (120°) dan kamera kedua dengan lensa omni untuk deteksi 360° di sekitar robot. Metode *color filtering HSV* diterapkan untuk deteksi bola, dengan hasil jarak deteksi kamera utama

mencapai 12 meter dan radius deteksi kamera omni hingga 4 meter. Peneliti menggunakan konfigurasi 3 roda omni yang memungkinkan pergerakan *holonomic* untuk sistem pergerakannya. Hasil penelitian menunjukkan bahwa penerapan *multivision* memberikan efektivitas pergerakan robot dengan lebih banyak kondisi pergerakan, sementara konfigurasi 3 roda omni menciptakan sistem pergerakan yang dinamis. Perbedaan dengan penelitian yang akan dilakukan adalah metode dan kamera yang digunakan. Penelitian yang akan dilakukan menggunakan metode *Hough Circle Transform* dan hanya satu kamera yaitu *Raspberry Pi Camera*.

Satriyo dkk. (2021) melakukan penelitian tentang sistem kontrol robot sepak bola beroda. Dalam penelitiannya, mereka mengembangkan sistem kontrol robot sepak bola beroda menggunakan metode *Finite State Machine* (FSM). Robot dirancang dengan tiga roda *omni-directional* dan menggunakan sensor CMUCam5 untuk mendeteksi bola dan gawang berdasarkan warna. Sistem FSM yang dikembangkan terdiri dari beberapa *state* utama seperti mencari bola, meluruskan posisi dengan bola, mencari gawang, dan menendang. Pengujian dilakukan dalam berbagai skenario dengan variasi posisi bola di lapangan. Hasil penelitian menunjukkan bahwa robot mampu menyelesaikan misi dengan tingkat keberhasilan 86% dan waktu rata-rata 29,24 detik. Robot dapat mendeteksi objek pada jarak 20-60 cm dan sudut -40° hingga 40° dalam kondisi pencahayaan 113-1213 lux. Perbedaan dengan penelitian yang akan dilakukan adalah penggunaan metode. Penelitian yang akan dilakukan menggunakan metode *Hough Circle Transform*.

Kemudian, penelitian yang dilakukan oleh Rizal dkk. (2020) membahas tentang implementasi sistem deteksi lembar jawaban komputer menggunakan

metode *Canny Edge Detection* dan *Hough Circle Transform*. Penelitian ini bertujuan untuk meningkatkan akurasi dalam mendeteksi jawaban pada lembar jawab komputer. Proses dimulai dengan *preprocessing* gambar, termasuk mengubah ukuran dan mengkonversi ke *grayscale*. Selanjutnya, metode *Canny* digunakan untuk mendeteksi tepi, diikuti oleh *Hough Circle Transform* untuk mendeteksi lingkaran jawaban. Dalam peningkatan akurasi, peneliti menerapkan teknik penentuan jarak antar jawaban berdasarkan koordinat. Sistem ini diimplementasikan menggunakan *library OpenCV* dan *Raspberry Pi*. Hasil pengujian menunjukkan tingkat akurasi mencapai 95,75% dalam mendeteksi jawaban. Peneliti menyimpulkan bahwa kombinasi *Canny Edge Detection* dan *Hough Circle Transform* merupakan metode yang efektif untuk deteksi jawaban pada lembar jawab komputer yang hakikatnya berupa lingkaran, meskipun faktor pencahayaan dan kualitas gambar dapat mempengaruhi hasil deteksi. Perbedaan dengan penelitian yang akan dilakukan adalah obyek deteksi. Obyek pada penelitian yang akan dilakukan adalah bola pada *robot soccer*.

Penelitian lain oleh Bahri (2020) membahas tentang penggunaan metode pusat dan *Circular Hough Transformation* (CHT) untuk mendeteksi lingkaran dalam sebuah citra, baik lingkaran tunggal maupun yang tumpang tindih. Metode pusat dan CHT menggunakan array akumulator dua dimensi untuk mendeteksi pusat dan jari-jari lingkaran. Prosesnya melibatkan pra-pemrosesan citra seperti input citra, deteksi objek, ambang batas tepi, dan skala abu-abu, yang kemudian diikuti dengan penerapan metode pusat untuk CHT. Implementasi menggunakan Matlab R2020b menunjukkan bahwa sistem yang dibangun dapat mendeteksi

semua lingkaran dalam citra dengan akurasi 100% dalam kondisi intensitas cahaya 0,93, ambang batas 0,33, serta jari-jari lingkaran antara 16px dan 110px. Perbedaan dengan penelitian yang akan dilakukan adalah obyek deteksi. Obyek pada penelitian yang akan dilakukan adalah bola pada *robot soccer*.

Penelitian yang dilakukan oleh Asri dkk. (2022) menggunakan *Circle Hough Transform* (CHT) sebagai metode utama untuk deteksi lingkaran roda kendaraan. CHT adalah pengembangan dari *Hough Transform* yang dikhususkan untuk mendeteksi bentuk lingkaran dalam citra digital. Metode ini memiliki kelebihan dalam ketahanannya terhadap *noise* dan kemampuannya untuk mendeteksi lingkaran bahkan ketika sebagian dari lingkaran tersebut terhalang. Dalam penelitiannya, mereka melakukan optimasi parameter CHT, termasuk *gradient threshold*, interval radius (*radrange*), radius filter untuk lokal maksima, dan radius jamak (*multirad*). Mereka menemukan bahwa nilai optimal untuk *gradient threshold* adalah 14, minimum *radrange* 27, maksimum *radrange* 100, *radius to local maxima* 30, dan *multirad* 0,98. Dengan parameter optimal ini, CHT berhasil mendeteksi semua lingkaran roda dari 225 citra kendaraan dengan akurasi 100%. Penelitian ini menunjukkan efektivitas CHT dalam deteksi lingkaran pada roda kendaraan. Perbedaan dengan penelitian yang akan dilakukan adalah obyek deteksi. Obyek pada penelitian yang akan dilakukan adalah bola pada *robot soccer*.

Putra & Puriyanto (2022) mengembangkan sistem deteksi dan pelacakan bola dengan menggunakan metode *Hough Circle Transform* dan *color filtering* pada robot sepak bola beroda (KRSBI-B) yang dilengkapi dengan kamera *omnidirectional*. Fokusnya adalah mengatasi kekurangan pada robot sebelumnya,

yang mengalami kesulitan mendeteksi bola karena perubahan jarak pandang dan intensitas cahaya. Penelitian ini mengimplementasikan metode *Hough Circle Transform* dan filter warna untuk mendeteksi bola, bahkan ketika sebagian bola tertutup oleh robot lawan. Uji coba dilakukan dengan berbagai intensitas cahaya dan sudut pandang 360 derajat. Hasil menunjukkan tingkat akurasi dan presisi deteksi hingga 100% dengan rata-rata error sebesar 0,14%. Sistem juga mampu mendeteksi bola hingga 75% ketika tertutup sebagian oleh robot, dengan kualitas citra yang baik ditunjukkan oleh nilai PSNR rata-rata 39,6%. Perbedaan dengan penelitian yang akan dilakukan adalah kamera dan hanya menggunakan satu metode. Kamera yang digunakan yaitu *Raspberry Pi Camera* dan metode yang digunakan yaitu *Hough Circle Transform*.

Pradana & Irmawati (2020) mengembangkan sistem pendeteksian bola pada robot penjaga gawang dengan menerapkan metode *Hough Circle* menggunakan *EmguCV*. Penelitian ini fokus pada pendeteksian bola berwarna oranye yang digunakan dalam kompetisi Kontes Robot Sepak Bola Indonesia (KRSBI). Dalam penelitian tersebut, metode *Hough Circle* digunakan untuk mendeteksi kontur lingkaran bola setelah melalui serangkaian proses pengolahan citra, seperti konversi warna ke format HSV, filtering gambar, dan deteksi tepi menggunakan *Canny Edge Detection*. Hasil penelitian menunjukkan akurasi pendeteksian bola sebesar 80%, serta robot berhasil menghadang bola dengan tingkat keberhasilan sebesar 90%. Penelitian ini menunjukkan bahwa metode *Hough Circle Transform* mampu mendeteksi bola secara efektif meskipun terdapat tantangan teknis pada aspek mekanis yang perlu disempurnakan untuk meningkatkan performa robot penjaga

gawang. Perbedaan dengan penelitian yang akan dilakukan adalah subyek robot dan kontroler. Subyek robot yaitu *robot soccer* beroda dan kontrolernya menggunakan *Raspberry Pi*.

Hariyadi dkk. (2023) mengembangkan sistem kendali arah pada robot *Unmanned Ground Vehicle* (UGV) berbasis komunikasi nirkabel *LoRa E32 433MHz* dengan metode *Proportional Integral Derivative (PID)*. Penelitian ini berfokus pada kemampuan robot untuk menghadapi arah tertentu secara presisi dengan memanfaatkan sensor kompas HMC5883L sebagai masukan umpan balik. Sistem ini dikendalikan melalui perangkat Android yang terhubung ke *Arduino Nano* menggunakan komunikasi serial, kemudian dipancarkan melalui modul LoRa ke penerima yang terdapat pada robot. Hasil pengujian menunjukkan bahwa penggunaan PID mampu mengurangi nilai error arah gerak robot dengan persentase kesalahan terkecil sebesar 0,57% hingga 12,51% untuk kombinasi parameter $K_p = 10$, $K_i = 0.2$, dan $K_d = 1.5$. Selain itu, dibandingkan dengan metode tanpa PID yang menghasilkan error hingga 49.22%, penerapan PID terbukti jauh lebih efektif dalam menstabilkan gerak dan orientasi robot. Penelitian ini menekankan pentingnya pemilihan parameter PID yang optimal dan penggunaan LoRa sebagai solusi komunikasi jarak jauh berdaya rendah. Perbedaan dengan penelitian yang akan dilakukan terletak pada sistem kendali dan jenis robot. Penelitian ini menggunakan robot tank dengan pengendali *Arduino Nano* sedangkan penelitian yang direncanakan akan memanfaatkan robot dengan konfigurasi serta algoritma kontrol dan sensor yang berbeda.

Isrofi dkk. (2021) merancang sebuah *prototype* robot pemotong rumput otomatis yang dikendalikan secara nirkabel menggunakan modul ESP32-CAM berbasis *Internet of Things (IoT)*. Penelitian ini bertujuan untuk mengatasi keterbatasan sistem kendali pada penelitian sebelumnya yang hanya mampu menjangkau jarak 7,2 meter serta belum mampu mengontrol kecepatan putaran mata pisau pemotong secara efisien. Dalam sistem ini, ESP32-CAM berperan sebagai mikrokontroler utama yang dilengkapi dengan kamera OV2640 untuk memantau area pemotongan secara *real-time*. Koneksi antara pengontrol dan robot dilakukan melalui jaringan *Wi-Fi* dengan akses melalui *IP address* yang dapat dibuka melalui *web browser* di laptop atau *smartphone*. Komponen utama pemotongan adalah motor *brushless* yang dikendalikan oleh modul ESC30A, memungkinkan pengaturan kecepatan putaran sesuai kebutuhan serta menghemat daya baterai hingga 0,16V per menit pada kecepatan maksimal. Hasil pengujian menunjukkan bahwa robot dapat beroperasi secara optimal dalam radius kendali hingga 50 meter di ruang terbuka. Selain itu, sistem juga diuji pada berbagai jenis medan seperti rumput berbatu, berpasir, dan berair, dengan hasil memuaskan kecuali pada medan basah karena keterbatasan daya tahan air. Perbedaan dengan penelitian yang akan dilakukan terletak pada karakteristik medan, metode kendali, dan penerapan algoritma adaptif.

Tabel 2. 1 Perbandingan dengan Penelitian Terdahulu

No.	Nama Peneliti	Judul Penelitian	Metode yang digunakan	Perbedaan
1.	Winarno dkk. (2020)	<i>Ball detection for KRSBI soccer robot using PeleeNet on omnidirectional camera</i>	<i>PeleeNet</i>	- Menggunakan model <i>peleenet</i> deep learning

No.	Nama Peneliti	Judul Penelitian	Metode yang digunakan	Perbedaan
				- Menggunakan kamera <i>omnidirectional</i>
2.	Risfendra dkk. (2020)	Sistem Pergerakan dan Deteksi Pada Robot Sepak Bola Beroda Berbasis <i>Image Processing</i> dengan Penerapan <i>Multivision</i>	<i>Hue Saturation Value (HSV)</i>	- Menggunakan dua kamera untuk deteksi objek - Menggunakan metode HSV
3.	Satriyo dkk. (2021)	Sistem Kontrol Robot Sepak Bola Beroda menggunakan <i>Finite State Machine (FSM)</i>	<i>Finite State Machine (FSM)</i>	- Perbedaan penggunaan metode, yaitu FSM.
4.	Rizal dkk. (2020)	<i>Canny Edge and Hough Circle Transformation for Detecting Computer Answer Sheets</i>	<i>Canny Edge Detection dan Hough Circle Transformation</i>	- Objek penelitian terkait berbeda yaitu lembar jawab komputer, sedangkan objek yang digunakan peneliti adalah bola dalam <i>robot soccer</i> .
5.	Bahri (2020)	Metode Pusat dan <i>Circular Hough Transformation</i> untuk Mendeteksi Lingkaran pada Sebuah Citra	<i>Circular Hough Transformation</i>	- Objek penelitian terkait adalah objek yang berbentuk lingkaran tanpa spesifikasi khusus, sedangkan objek yang digunakan peneliti adalah bola dalam <i>robot soccer</i> .
6.	Asri dkk. (2022)	Deteksi Roda Kendaraan Dengan <i>Circle Hough Transform (CHT)</i> dan <i>Support Vector Machine (SVM)</i>	<i>Circle Hough Transform dan Support Vector Machine (SVM)</i>	- Objek penelitian terkait berbeda yaitu roda pada kendaraan, sedangkan objek yang digunakan peneliti adalah bola dalam <i>robot soccer</i> .

No.	Nama Peneliti	Judul Penelitian	Metode yang digunakan	Perbedaan
7.	Putra & Puriyanto (2022)	Sistem Deteksi dan Pelacakan Bola dengan Metode <i>Hough circle Transform</i> Menggunakan Kamera <i>Omnidirectional</i> pada Robot Sepak Bola Beroda	<i>Hough Circle Transform dan Colour Filtering</i>	- Metode yang digunakan pada penelitian terkait ada dua sedangkan yang akan digunakan oleh peneliti hanya satu. - Kamera yang digunakan
8.	Pradana & Irmawati (2020)	Pendeteksi Bola Pada Robot Penjaga Gawang Menggunakan Metode <i>Hough Circle</i>	<i>Hough Circle</i>	- Menggunakan kontroler ATmega. - Subyek penelitian hanya robot penjaga gawang.
9.	Hariyadi dkk. (2023)	<i>433Mhz based Robot using PID (Proportional Integral Derivative) for Precise Facing Direction</i>	<i>Proportional Integral Derivative (PID)</i>	- Menggunakan PID - Menggunakan robot yang berbeda.
10.	Isrofi dkk. (2021)	Rancang Bangun Robot Pemotong Rumput Otomatis Menggunakan <i>Wireless</i> Kontroler Modul Esp32-Cam Berbasis <i>Internet Of Things (Iot)</i>	-	- Robot pemotong rumput. - Kontrol manual.
11.	Usulan Penelitian	Kontrol Otomasi Robot Soccer Menggunakan Metode <i>Hough Circle Transform</i>	<i>Hough Circle Transform</i>	- Menggunakan satu metode saja yaitu <i>Hough Circle Transform</i> - Menggunakan <i>Raspberry Pi</i> sebagai kontrolernya. - Menggunakan <i>Raspberry Pi Camera</i> untuk menangkap citra.

BAB III

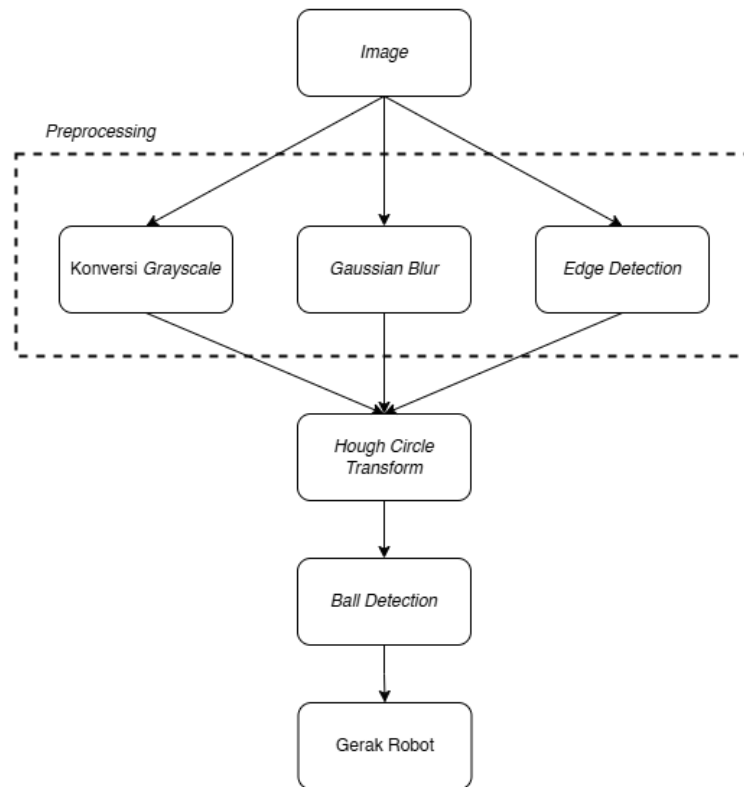
DESAIN DAN IMPLEMENTASI

3.1 Desain Sistem

Proses dimulai dengan pengambilan citra oleh *Raspberry Pi Camera* dalam format R, G, B dimana citra yang diambil berupa *frame* video yang menampilkan kondisi arena. Setelah citra diperoleh, sistem melakukan tahap *preprocessing* untuk meningkatkan kualitas data sebelum dianalisis lebih lanjut. Langkah pertama dalam *preprocessing* adalah konversi citra berwarna atau R, G, B ke *grayscale* supaya hanya intensitas cahaya yang diproses (kontras), bukan warna. Setelah itu, *Gaussian Blur* diterapkan untuk mengurangi *noise* dan membuat tepi objek lebih jelas. Selanjutnya adalah pendeteksian tepi menggunakan *Canny Edge Detection*, yang berfungsi untuk menyoroti batas objek dalam citra. Algoritma ini menyaring tepi yang paling signifikan berdasarkan perubahan intensitas cahaya sehingga hanya bentuk utama yang digunakan untuk analisis lebih lanjut.

Setelah tepi objek terdeteksi, sistem menerapkan *Hough Circle Transform* untuk mengenali bola berdasarkan bentuk lingkarannya. Metode ini bekerja dengan mencari pusat lingkaran dan radiusnya dalam ruang parameter dengan akumulasi suara (*voting*), dimana semakin banyak suara yang terakumulasi di suatu titik, semakin besar kemungkinan titik tersebut merupakan pusat bola. Jika bola berhasil dideteksi, sistem menghitung posisi relatifnya terhadap robot untuk menentukan arah dan kecepatan gerakan yang tepat. Kontrol pergerakan dilakukan dengan mengatur kecepatan motor dengan *Pulse Width Modulation* (PWM) melalui driver

L298N sehingga robot dapat menyesuaikan kecepatannya tergantung pada jarak bola.



Gambar 3. 1 Desain Sistem

3.2 *Preprocessing*

Preprocessing merupakan tahap awal dalam pemrosesan citra yang bertujuan untuk meningkatkan kualitas gambar sebelum dilakukan analisis lebih lanjut. *Preprocessing* pada penelitian ini melalui tiga tahap utama, yaitu konversi *grayscale*, penerapan *Gaussian Blur*, dan deteksi tepi menggunakan metode *Edge Detection*.

3.2.1 Konversi *Grayscale*

Proses ini mengubah citra berwarna menjadi citra berskala abu-abu supaya lebih mudah diproses dalam tahap analisis selanjutnya. Pada citra berwarna, setiap piksel memiliki tiga kanal warna utama yaitu merah (*Red*), hijau (*Green*), dan biru (*Blue*), yang sering disebut sebagai model warna R, G, B . Setiap kanal menyimpan informasi warna dalam rentang intensitas 0 hingga 255. Informasi warna ini tidak selalu diperlukan dalam proses deteksi bentuk sehingga citra dikonversi menjadi *grayscale* supaya sistem hanya mempertimbangkan intensitas cahaya tanpa memperhitungkan warna. Konversi *grayscale* dilakukan dengan cara menghitung nilai intensitas baru untuk setiap piksel berdasarkan kombinasi bobot dari ketiga kanal warna sesuai dengan Persamaan 3.1.

$$I = (0.299 \times R) + (0.587 \times G) + (0.114 \times B) \quad (3.1)$$

Keterangan:

I : intensitas piksel *grayscale*

R, G, B : nilai intensitas untuk masing-masing kanal merah, hijau, dan biru

Bobot dalam persamaan ini didasarkan pada standar Rec. 601. Ini adalah standar yang digunakan untuk pengkodean warna dalam video analog dan digital serta nilai-nilai ini didasarkan pada persepsi manusia terhadap warna yang berbeda (Setiawan & Faisal, 2020). Pembobotan ini mengacu pada sensitivitas mata manusia terhadap warna-warna tertentu. Kanal hijau memiliki bobot terbesar karena mata manusia lebih peka terhadap warna hijau dibandingkan dengan merah dan biru.

Pseudocode 3.1 Konversi Grayscale

START

1. Input frame citra berwarna yang diambil oleh kamera

```

2. Lakukan konversi citra berwarna ke citra grayscale dengan langkah
   berikut:
   - Ambil nilai intensitas untuk setiap piksel berdasarkan kombinasi bobot
     dari saluran warna (merah, hijau, biru)
   - Hasilkan citra dengan satu kanal (grayscale)
3. Hasil citra satu kanal (grayscale) digunakan untuk proses berikutnya
END

```

Pada *Pseudocode* 3.1, proses konversi memanfaatkan fungsi transformasi yang dirancang untuk menghasilkan nilai intensitas piksel berdasarkan formula yang umum digunakan dalam pengolahan citra digital sesuai dengan Persamaan 3.1. Dengan konversi ini, sistem dapat mengolah citra dengan lebih efisien karena jumlah informasi yang harus diproses berkurang menjadi satu kanal saja sehingga mempercepat pemrosesan tanpa kehilangan detail penting dalam citra. Hasil akhirnya dapat dilihat pada Gambar 3.2.



Gambar 3. 2 Hasil *Grayscale*

3.2.2 *Gaussian Blur*

Setelah konversi *grayscale*, tahap berikutnya adalah penerapan *Gaussian Blur* untuk mengurangi *noise* yang dapat mengganggu proses deteksi objek. *Noise* dalam citra dapat berasal dari pencahayaan yang tidak merata, tekstur latar

belakang, atau artefak kamera. *Noise* ini dapat menyebabkan kesalahan dalam mendeteksi tepi objek, terutama dalam metode yang bergantung pada perbedaan intensitas seperti *Hough Circle Transform*. Penerapan *Gaussian Blur* dilakukan menggunakan filter *Gaussian* yang menghitung rata-rata bobot piksel berdasarkan distribusi *Gaussian*. Rumus fungsi *Gaussian* dapat dilihat pada Persamaan 3.2.

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \quad (3.2)$$

Keterangan:

$G(x, y)$: nilai bobot Gaussian pada koordinat piksel (x, y)

σ : standar deviasi distribusi *Gaussian*

e : basis dari logaritma natural (≈ 2.718)

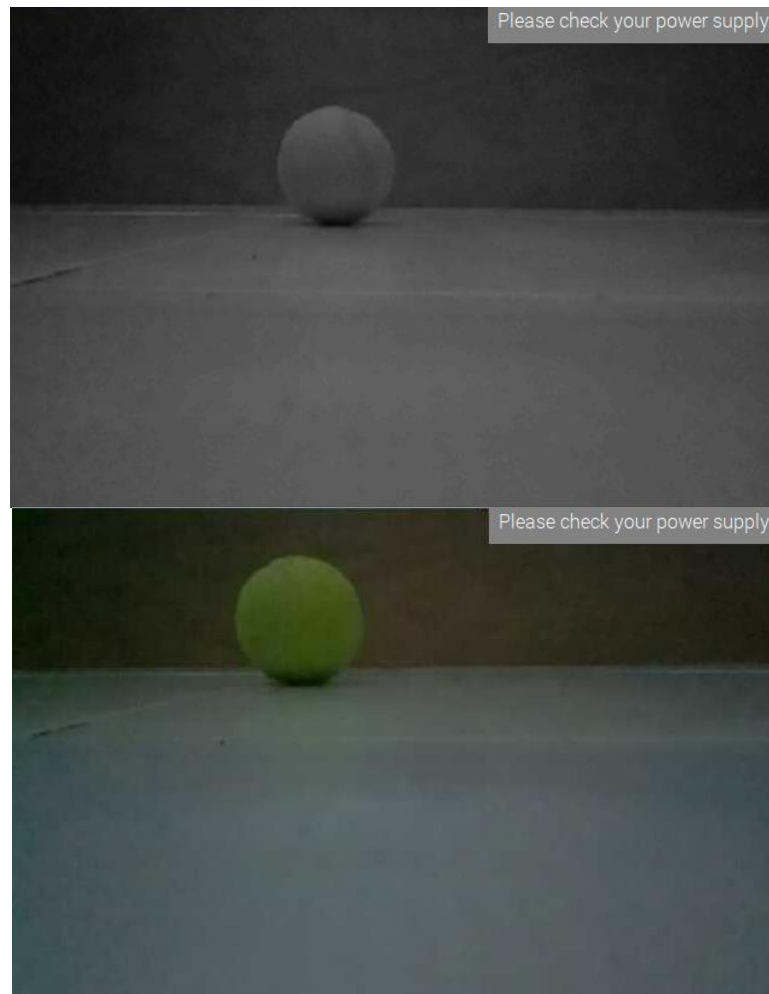
Pseudocode 3.2 Gaussian Blur

```

START
1. Citra grayscale sebagai input
2. Tentukan ukuran kernel untuk Gaussian blur
3. Terapkan Gaussian blur pada citra dengan langkah berikut:
   - Hapus noise dengan pengaburan halus menggunakan filter Gaussian
   - Pastikan pengaburan tidak menghilangkan fitur penting pada citra
4. Hasil citra yang telah diblur digunakan untuk proses selanjutnya
END

```

Pada *Pseudocode 3.2*, citra *grayscale* digunakan sebagai input karena deteksi objek lebih akurat dilakukan pada citra yang memiliki satu kanal warna. Ukuran kernel yang umum digunakan dalam *Gaussian Blur* adalah 5×5 atau 7×7 piksel, tergantung pada tingkat penghalusan yang diinginkan. Semakin besar nilai σ dan ukuran kernel, semakin besar pula efek pengaburan yang diterapkan pada citra. Dengan menerapkan *Gaussian Blur*, citra menjadi lebih bersih sehingga deteksi tepi dapat dilakukan dengan lebih akurat. Hasilnya dapat dilihat pada Gambar 3.3.



Gambar 3. 3 *Gaussian Blur*

3.2.3 *Edge Detection*

Setelah citra dihaluskan menggunakan *Gaussian Blur*, tahap berikutnya adalah deteksi tepi (*Edge Detection*) yang bertujuan untuk menyoroti batas objek dalam citra. Deteksi tepi sangat penting dalam sistem ini karena algoritma *Hough Circle Transform* bergantung pada tepi objek untuk mengenali bentuk lingkaran. Salah satu algoritma yang digunakan dalam deteksi tepi adalah *Canny Edge Detection* yang dikenal memiliki tingkat akurasi tinggi dalam menemukan perubahan intensitas yang tajam (Perangin-angin & Harianja, 2020). Algoritma

Canny Edge Detection terdiri dari beberapa tahapan utama seperti pada *Pseudocode*

3.3.

Pseudocode 3.3 Canny Edge Detection

```

START
1. Input citra hasil Gaussian blur
2. Hitung gradien menggunakan operator Sobel
3. Lakukan Non-Maximum Suppression untuk menipiskan tepi:
   - Jika  $G > T_{high}$ , piksel dianggap sebagai tepi
   - Jika  $G < T_{low}$ , piksel bukan tepi
   - Jika  $T_{low} < G < T_{high}$ , piksel diperiksa keterhubungannya dengan piksel tepi lain
4. Simpan citra hasil deteksi tepi
END

```

Pseudocode 3.3 menjelaskan alur algoritma *Canny* dalam menemukan tepi objek dalam citra. Langkah pertama yaitu input citra hasil dari *Gaussian Blur*. Selanjutnya, perhitungan gradien intensitas menggunakan operator Sobel untuk menemukan perubahan intensitas cahaya yang signifikan dalam citra. Gradien ini dihitung dalam dua arah, yaitu arah horizontal (G_x) dan arah vertikal (G_y). Kemudian menghitung gradien total yang menunjukkan seberapa besar perubahan intensitas terjadi pada suatu piksel dalam citra dengan Persamaan 3.3.

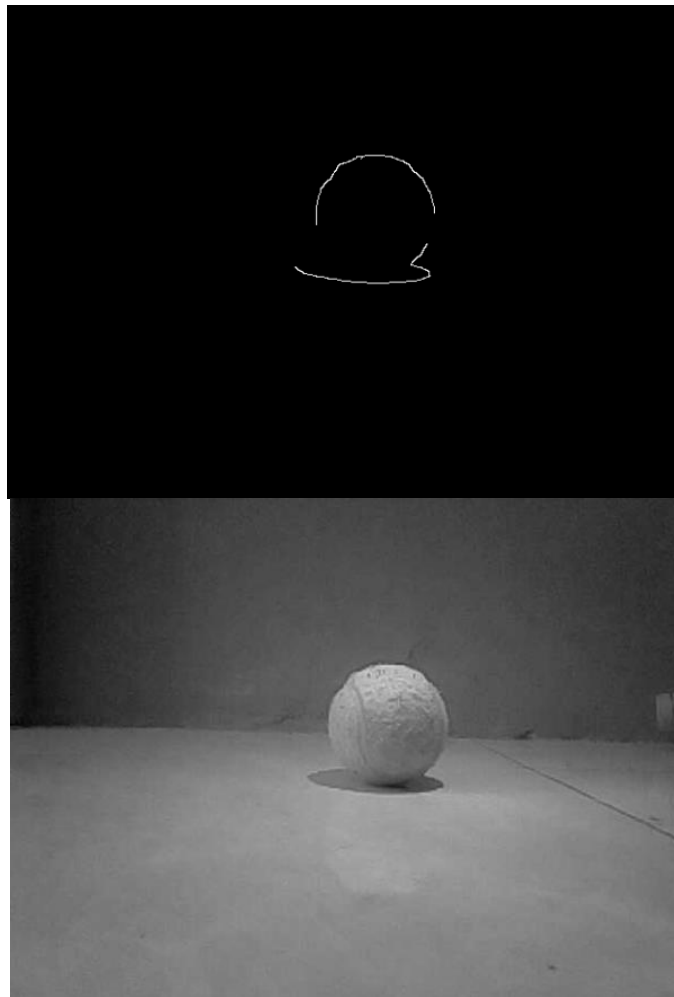
$$G = \sqrt{G_x^2 + G_y^2} \quad (3.3)$$

Nilai G merupakan hasil akhir dari gradien yang merepresentasikan perubahan intensitas total di setiap piksel. Nilai gradien ini menentukan seberapa tajam perubahan intensitas cahaya dalam citra. Sistem kemudian menentukan arah gradien menggunakan Persamaan 3.4.

$$\theta = \tan^{-1} \left(\frac{G_y}{G_x} \right) \quad (3.4)$$

Persamaan 3.4 menghitung arah perubahan intensitas dalam satuan derajat atau radian. Arah gradien mengacu pada orientasi tepi yang terbentuk dalam citra dan digunakan untuk menyaring tepi yang benar-benar signifikan. Dalam praktiknya, arah gradien dikelompokkan ke dalam empat kategori utama, yaitu 0° (horizontal), 45° (diagonal naik), 90° (vertikal), dan 135° (diagonal turun). Kategorisasi ini membantu dalam proses penyaringan tepi melalui algoritma *Non-Maximum Suppression* dimana hanya piksel dengan nilai gradien tertinggi dalam arah tertentu yang dipertahankan, sementara piksel lain yang berada di sekitarnya akan dikurangi menjadi nol agar tidak mengganggu analisis selanjutnya.

Setelah penyaringan tepi dilakukan, tahap berikutnya adalah *double thresholding* yang digunakan untuk menentukan apakah suatu piksel merupakan tepi kuat, tepi lemah, atau bukan bagian dari tepi sama sekali. Piksel dengan intensitas lebih tinggi dari ambang batas atas (T_{high}) dikategorikan sebagai tepi kuat dan langsung diterima sedangkan piksel dengan intensitas antara ambang batas atas (T_{high}) dan bawah (T_{low}) dikategorikan sebagai tepi lemah. Tepi lemah hanya akan dipertahankan jika terhubung dengan tepi kuat, sementara piksel dengan intensitas lebih rendah dari ambang batas bawah (T_{low}) akan dihapus. Langkah terakhir adalah *Hysteresis Thresholding* dimana tepi lemah yang tidak memiliki hubungan dengan tepi kuat akan dieliminasi agar hanya kontur yang valid yang dipertahankan. Dengan menerapkan *edge detection* menggunakan metode *Canny*, citra yang dihasilkan hanya menampilkan tepi-tepi signifikan yang merupakan batas objek utama. Hasil deteksi tepi *Canny* dapat dilihat pada Gambar 3.4.



Gambar 3. 4 Hasil *Canny Edge Detection*

3.3 *Hough Circle Transform*

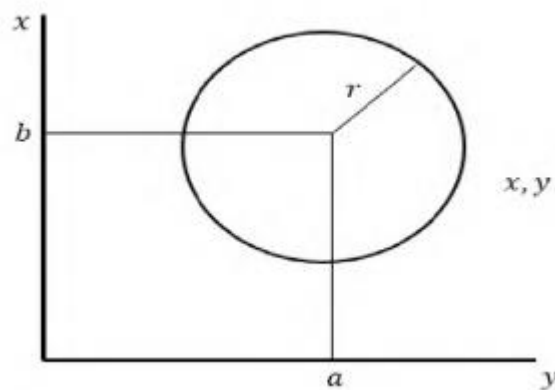
3.3.1 *Konsep Hough Circle Transform*

Hough Circle Transform merupakan metode dalam pengolahan citra digital yang digunakan untuk mendeteksi objek berbentuk lingkaran (S. Rahman dkk., 2020). Metode ini merupakan pengembangan dari *Hough Transform* yang awalnya digunakan untuk mendeteksi garis lurus. Dalam penelitian ini, metode tersebut diterapkan untuk mendeteksi bola. Implementasi metode ini menggunakan *library*

opencv dengan bahasa pemrograman *python*. Langkah-langkah *Hough Circle Transform* dapat dilihat pada Gambar 3.6.

Metode ini bekerja dengan prinsip dasar bahwa setiap lingkaran dalam ruang citra dapat direpresentasikan dalam ruang parameter. Dalam sistem koordinat Kartesius, persamaan umum dari suatu lingkaran dinyatakan seperti pada Persamaan 3.5 berikut dan diilustrasikan pada Gambar 3.5.

$$(x - a)^2 + (y - b)^2 = r^2 \quad (3.5)$$



Gambar 3. 5 Ilustrasi Koordinat

Koordinat (x, y) merupakan koordinat titik pada lingkaran, (a, b) adalah koordinat pusat lingkaran yang dicari, dan r adalah radius lingkaran yang akan dideteksi. Proses deteksi dilakukan dengan mencari kombinasi (a, b, r) yang paling sesuai dengan fitur lingkaran yang terdapat dalam citra. Metode ini mengumpulkan suara (*voting*) dalam suatu ruang parameter yang disebut *hough space*, dimana setiap kemungkinan pusat dan radius lingkaran diberikan suara berdasarkan titik-titik tepi yang ditemukan dalam citra.

Tahap awal dalam metode ini adalah input citra yang telah melalui tahap *preprocessing*. Setelah itu, inisialisasi parameter yang akan digunakan dalam proses deteksi. Beberapa parameter penting yang harus diatur meliputi resolusi akumulator, ambang batas deteksi, serta rentang ukuran lingkaran yang akan diidentifikasi. Parameter resolusi akumulator (dp) digunakan untuk mengontrol skala transformasi antara ruang citra dan ruang parameter, dimana nilai yang lebih rendah akan memberikan akurasi lebih tinggi namun dengan konsekuensi peningkatan waktu komputasi. Jarak minimum antar pusat lingkaran ($minDist$) digunakan untuk menghindari deteksi ganda pada objek yang sama. Parameter lainnya adalah $param1$ yang digunakan sebagai *threshold* pertama dalam algoritma deteksi tepi *Canny*, serta $param2$ yang berfungsi sebagai ambang batas untuk validasi jumlah suara dalam akumulator. Kemudian, menentukan batas minimum dan maksimum radius ($minRadius$ & $maxRadius$) untuk memastikan bahwa hanya lingkaran dengan ukuran tertentu yang akan dipertimbangkan dalam proses deteksi.

Setelah inisialisasi parameter, masuk ke tahap inti dari metode ini yaitu *hough space voting*. Pada tahap ini, setiap titik tepi dalam citra yang telah terdeteksi akan dihitung sebagai kandidat pusat lingkaran untuk berbagai kemungkinan radius r . Setiap nilai r , sistem akan menghitung koordinat pusat lingkaran menggunakan Persamaan 3.6 dan Persamaan 3.7.

$$a = x - r \cos \theta \quad (3.6)$$

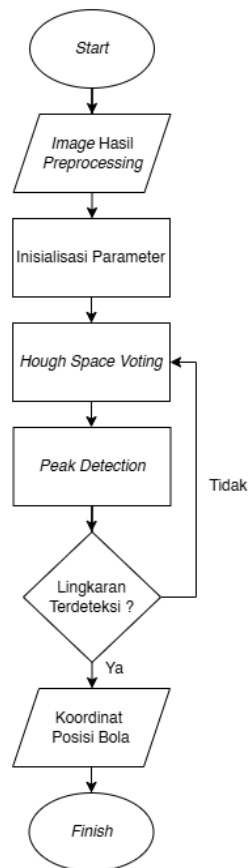
$$b = y - r \sin \theta \quad (3.7)$$

Dimana θ bervariasi dari 0 hingga 2π sehingga semua kemungkinan pusat lingkaran dapat dipertimbangkan. Proses ini menghasilkan banyak kandidat pusat lingkaran yang masing-masing mendapatkan suara dalam *hough space*, yaitu sebuah ruang parameter tiga dimensi yang menyimpan kemungkinan koordinat pusat (a, b) dan radius r . Semakin banyak suara yang diterima oleh suatu koordinat pusat dalam akumulator, semakin besar kemungkinan bahwa titik tersebut benar-benar merupakan pusat dari suatu lingkaran.

Setelah semua titik tepi memberikan suara dalam akumulator, langkah selanjutnya adalah *peak detection*, yaitu proses mengidentifikasi puncak *voting* tertinggi dalam *hough space*. Puncak *voting* tertinggi menunjukkan koordinat pusat lingkaran yang paling mungkin sesuai dengan objek berbentuk lingkaran dalam citra. Jika jumlah suara yang terkumpul dalam suatu koordinat pusat lingkaran melebihi ambang batas yang telah ditentukan oleh parameter *param2*, maka sistem akan mengonfirmasi bahwa objek tersebut adalah lingkaran yang valid. Jika jumlah suara tidak mencukupi maka sistem akan mengulang kembali proses transformasi untuk mencari kandidat lain yang lebih sesuai.

Tahap berikutnya adalah pengecekan keberhasilan deteksi, yang dilakukan dengan mengevaluasi apakah lingkaran telah berhasil dideteksi berdasarkan ambang batas yang telah ditentukan. Jika sistem berhasil mendeteksi lingkaran, maka koordinat pusat (x, y) dan radius r dari lingkaran akan disimpan sebagai hasil deteksi. Jika tidak, sistem akan kembali ke tahap *hough space voting* untuk mengulang proses pencarian dengan kemungkinan parameter yang berbeda. Jika sistem berhasil menemukan lingkaran yang valid, maka hasil deteksi akan

digunakan untuk menentukan koordinat posisi bola yang menjadi informasi utama bagi sistem kontrol robot. Informasi ini memungkinkan robot untuk mengetahui posisi bola dalam lapangan sehingga dapat mengambil keputusan untuk bergerak sesuai yang telah diprogramkan. Setelah posisi bola berhasil diperoleh, proses deteksi dianggap selesai dan sistem dapat melanjutkan ke tahap berikutnya dalam pengambilan keputusan robot.



Gambar 3. 6 Flowchart Hough Circle Transform

3.3.2 Implementasi *Hough Circle Transform*

Langkah-langkah implementasi metode ini dapat dilihat pada *Pseudocode* 3.4 berikut.

Pseudocode 3.4 Deteksi Lingkaran

```

START
1. Input berupa citra yang telah diblur dan citra asli
2. Lakukan deteksi lingkaran menggunakan metode Hough Circle Transform
   dengan mengatur parameter berikut:
   - dp: 1,2
   - minDist: 50
   - param1: 50
   - param2: 30
   - minRadius: 10
   - maxRadius: 100
3. Jika lingkaran terdeteksi:
   - Tentukan koordinat pusat lingkaran (x, y) dan radius (r)
   - Gambarkan lingkaran di citra hasil dengan warna hijau
   - Tandai pusat lingkaran dengan warna oranye
4. Jika lingkaran tidak terdeteksi Kembali ke langkah pertama dengan input
   frame berikutnya
5. Hasilnya citra asli dengan lingkaran yang ditandai
END

```

Pada *Pseudocode 3.4*, deteksi lingkaran dilakukan dengan fungsi dari *library OpenCV* menggunakan metode *Hough Gradient*. Metode ini memanfaatkan perubahan intensitas (gradien) pada tepi objek untuk mendeteksi pusat dan radius lingkaran. Proses deteksi terdiri dari dua tahap utama yaitu pendeteksian tepi dan akumulasi suara di ruang parameter. Pendeteksian tepi menggunakan *Canny Edge Detection* untuk menemukan piksel yang berpotensi menjadi bagian dari lingkaran. Piksel tepi ini menjadi input untuk akumulasi suara. Setelah tepi terdeteksi, algoritma mengakumulasi suara (*votes*) pada kemungkinan pusat dan radius lingkaran di ruang parameter. Puncak suara dalam akumulator diidentifikasi sebagai pusat lingkaran.

Pada *Pseudocode 3.4*, terdapat beberapa parameter penting yang memengaruhi hasil deteksi lingkaran. Parameter pertama, *dp*, adalah faktor resolusi akumulator relatif terhadap citra asli. Nilai ini digunakan untuk mengurangi ukuran

akumulator tanpa kehilangan akurasi sehingga proses deteksi dapat dilakukan lebih cepat. Dalam sistem ini, nilai dp diatur sebesar 1,2.

Parameter kedua adalah $minDist$, yaitu jarak minimum antar pusat lingkaran yang terdeteksi. Nilai ini mencegah deteksi ganda pada objek yang sama dengan memastikan bahwa dua lingkaran yang terdeteksi memiliki jarak tertentu. Nilai $minDist$ diatur sebesar 50 piksel.

Selanjutnya, terdapat parameter $param1$, yang merupakan ambang batas pertama untuk metode *Canny Edge Detection*. Algoritma ini mendeteksi tepi objek dalam citra dan nilai $param1$ menentukan sensitivitas terhadap perubahan intensitas pada tepi. Dalam implementasi ini, nilai $param1$ ditetapkan sebesar 50 yang dianggap cukup untuk mendeteksi tepi dengan presisi tanpa menghasilkan terlalu banyak noise.

Parameter lain, $param2$, yaitu ambang batas untuk akumulator pada tahap deteksi lingkaran. Nilai ini memengaruhi sensitivitas algoritma terhadap lingkaran yang potensial. Semakin kecil nilai $param2$, semakin sensitif algoritma dalam mendeteksi lingkaran. Hal ini dapat meningkatkan risiko mendeteksi lingkaran palsu. Nilai $param2$ diatur sebesar 30 untuk memastikan deteksi yang stabil dan akurat. Terakhir, parameter $minRadius$ dan $maxRadius$ menentukan ukuran lingkaran minimum dan maksimum yang akan dideteksi. Dalam sistem ini, nilai $minRadius$ diatur sebesar 10 piksel sedangkan $maxRadius$ sebesar 100 piksel. Pengaturan rentang ini dirancang berdasarkan perkiraan ukuran bola dalam citra digital sehingga algoritma dapat secara efektif mendeteksi objek dengan ukuran yang relevan.

Selain mendeteksi lingkaran, sistem ini juga dapat menghitung jarak antara objek dan kamera menggunakan konsep proyeksi perspektif. Perhitungan ini didasarkan pada hubungan antara ukuran objek dalam citra dan panjang fokus kamera, yang dinyatakan dalam rumus pada Persamaan 3.8.

$$Z = \frac{F \times W}{P} \quad (3.8)$$

Dimana Z adalah jarak bola ke kamera, F merupakan panjang fokus kamera dalam satuan piksel, W adalah diameter sebenarnya dari bola, dan P adalah diameter proyeksi bola dalam citra. Dengan menggunakan metode ini, sistem dapat secara akurat menentukan jarak objek berdasarkan ukuran yang terdeteksi dalam citra digital.

Pseudocode 3.5 Perhitungan Jarak

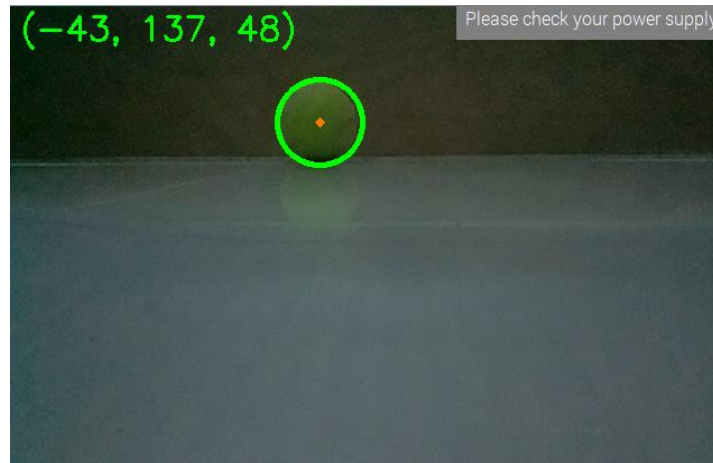
```

START
1. Input citra asli, citra yang telah diblur, koordinat pusat citra
   (center_x, center_y), panjang fokus kamera dalam piksel (F), diameter
   objek sebenarnya (W)
2. Lakukan deteksi lingkaran pada citra
3. Jika lingkaran terdeteksi:
   a. Tentukan koordinat pusat lingkaran (x, y) dan radius (r)
   b. Hitung posisi relatif lingkaran terhadap pusat citra
   c. Hitung diameter lingkaran dalam citra (P)
   d. Hitung jarak objek (z) dari kamera
   e. Anotasi citra hasil dengan informasi posisi relatif dan jarak
4. Hasilnya citra asli dengan lingkaran yang ditandai dan informasi
   koordinat lingkaran serta jarak objek dari kamera
END

```

Pada *Pseudocode 3.5*, langkah setelah lingkaran terdeteksi adalah menghitung posisi relatif lingkaran terhadap pusat citra kamera yang dilakukan dengan mengurangi koordinat pusat lingkaran dengan koordinat tengah citra. Radius lingkaran yang terdeteksi dikonversi menjadi diameter (P) dengan mengalikan nilai radius sebanyak dua kali. Berdasarkan informasi diameter ini,

jarak objek ke kamera dihitung menggunakan rumus pada Persamaan 3.8. Hasil tampilan deteksi lingkaran dan perhitungan jarak dapat dilihat pada Gambar 3.6.



Gambar 3. 7 Hasil Deteksi Lingkaran dan Jarak

3.4 Integrasi Sistem Kontrol Robot

Sistem kontrol robot pada penelitian ini memanfaatkan informasi posisi horizontal bola (sumbu-x) dan estimasi jarak bola (z) terhadap kamera yang diperoleh dari proses deteksi menggunakan metode *Hough Circle Transform*. Informasi tersebut menjadi dasar dalam pengambilan keputusan gerak robot yang dibagi menjadi tiga tahap utama yaitu pencarian bola, penyesuaian arah gerak berdasarkan posisi horizontal bola, dan pengaturan kecepatan serta aksi robot berdasarkan jarak bola terhadap robot.

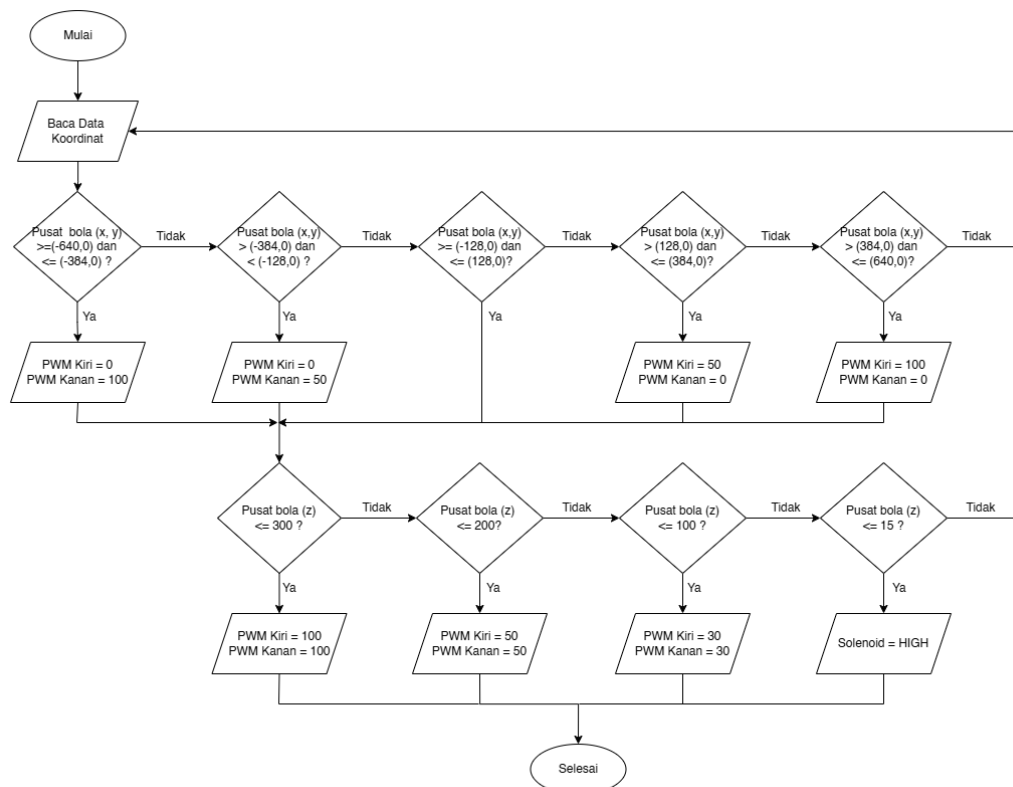
Tahap pertama, ketika bola tidak terdeteksi dalam *frame* kamera, robot secara otomatis akan masuk ke dalam mode pencarian. Dalam mode ini, robot dikendalikan untuk melakukan rotasi ke arah kiri secara perlahan sampai bola kembali terdeteksi dalam jangkauan kamera. Mekanisme ini memastikan bahwa sistem tetap aktif dan adaptif dalam kondisi kehilangan visual terhadap bola.

Tahap kedua, saat bola telah terdeteksi, posisi horizontal bola pada *frame* (nilai sumbu-x) akan dievaluasi dan dibagi menjadi lima zona yaitu zona kiri jauh, kiri dekat, tengah, kanan dekat, dan kanan jauh. Zona-zona ini digunakan untuk menentukan arah gerak robot. Jika bola berada di zona kiri jauh (-640 hingga -384 px) atau kanan jauh (384 hingga 640 px), robot akan berputar secara cepat ke arah tersebut. Jika bola berada di zona kiri dekat (-384 hingga -128 px) atau kanan dekat (128 hingga 384), robot hanya akan melakukan penyesuaian arah atau berputar secara perlahan ke arah tersebut. Apabila bola berada di zona tengah, robot diarahkan untuk maju lurus ke depan. Apabila bola terdeteksi berada di zona tengah (-128 hingga 128 px), robot diarahkan untuk maju lurus lalu melanjutkan evaluasi terhadap jarak bola.

Tahap ketiga, parameter jarak (z) digunakan untuk menentukan kecepatan gerak maju robot. Zona jarak dibagi menjadi tiga kategori yaitu zona jauh (200–300 cm), zona sedang (100–200 cm), dan zona dekat (≤ 100 cm). Ketika bola semakin dekat, maka kecepatan maju robot semakin meningkat dimana kecepatan dikendalikan melalui sinyal *Pulse Width Modulation* (PWM) yang dikirim ke motor dc melalui driver L298N. PWM dengan *duty cycle* lebih tinggi akan menghasilkan kecepatan motor yang lebih besar. Dalam implementasi ini, zona jarak menjadi landasan untuk menentukan nilai PWM dasar. Sistem ini menerapkan sinkronisasi PWM antara dua motor untuk memastikan pergerakan yang stabil. Hal ini dilakukan karena dua motor dc yang identik secara spesifikasi, sering kali menghasilkan kecepatan yang sedikit berbeda akibat perbedaan karakteristik fisik seperti torsi dan beban. Oleh karena itu, perlu dilakukan kalibrasi sinkronisasi nilai

PWM secara manual melalui uji coba untuk mendapatkan kompensasi nilai PWM yang sesuai antara kedua motor. Sinkronisasi ini sangat penting untuk mencegah gerakan melenceng yang tidak diinginkan khususnya saat robot diperintahkan untuk bergerak lurus menuju bola.

Pada saat bola berada sangat dekat dengan robot ($z \leq 15$ cm) dan terletak di zona tengah, sistem akan mengaktifkan aktuator solenoid yang bertugas untuk menendang bola. Aktivasi solenoid dikendalikan oleh pin *GPIO Raspberry Pi* dengan sinyal logika *HIGH* selama periode singkat untuk mengeksekusi aksi. Jika bola tidak berada pada zona dekat ($z \leq 15$ cm), pin *GPIO* dikembalikan ke logika *LOW* untuk mencegah pemanasan berlebih pada solenoid.



Gambar 3. 8 Flowchart Sistem Kontrol Robot

Sistem ini menggunakan kontrol PWM (*Pulse Width Modulation*) untuk mengatur kecepatan dan arah motor dc melalui *Driver* L298N. PWM adalah teknik yang digunakan untuk mengontrol kekuatan atau intensitas sinyal dengan mengubah durasi pulsa dalam suatu periode waktu. Cara kerja PWM dengan mengubah durasi sinyal aktif (*ON*) dan tidak aktif (*OFF*) dalam satu siklus untuk mengontrol tingkat daya atau intensitas dari perangkat yang terhubung. Rasio antara waktu sinyal aktif dan total waktu siklus disebut "*duty cycle*" dan dinyatakan dalam persentase. PWM mengontrol daya yang diberikan ke motor dengan mengubah *duty cycle* sinyal. *Duty cycle* dapat dihitung dengan Persamaan 3.9.

$$Duty\ Cycle = \frac{T_{on}}{T_{on} + T_{off}} \cdot 100\% \quad (3.9)$$

Dimana T_{on} adalah durasi waktu sinyal aktif dan T_{off} adalah durasi waktu sinyal tidak aktif. *Duty cycle* menyatakan berapa lama sinyal berada dalam keadaan aktif dibandingkan dengan total waktu siklus. Kecepatan motor berbanding lurus dengan *duty cycle* PWM.

Tabel 3. 1 Sinkronisasi PWM

Percobaan ke-	PWM Motor Kiri	PWM Motor Kanan	Hasil Gerakan Robot
1	100	100	Maju Lurus
2	100	90	Maju Lurus
3	90	100	Maju Lurus
4	100	80	Deviasi Kanan
5	80	100	Deviasi Kiri
6	90	90	Maju Lurus
7	90	80	Maju Lurus
8	80	90	Maju Lurus
9	70	90	Deviasi Kiri
10	90	70	Deviasi Kanan
11	60	60	Maju Lurus

12	60	50	Maju Lurus
13	50	60	Maju Lurus
14	60	40	Deviasi Kanan
15	40	60	Deviasi Kiri
16	30	30	Deviasi Kanan
17	30	25	Maju Lurus
18	20	20	Maju Lurus
19	15	15	Maju Lurus

Tabel 3.1 menunjukkan kalibrasi nilai *Pulse Width Modulation* (PWM) pada motor kiri dan kanan yang dilakukan untuk mencapai pergerakan maju lurus yang pada robot. Berdasarkan 19 percobaan, terlihat bahwa nilai PWM yang simetris sering kali menghasilkan pergerakan lurus. Namun, terdapat kasus dimana nilai simetris justru menyebabkan deviasi lateral, seperti pada PWM 30-30 yang menghasilkan "Deviasi Kanan". Hal ini mengindikasikan pentingnya penyesuaian non-simetris untuk mengkompensasi perbedaan karakteristik motor atau pengaruh eksternal pada kecepatan rendah, sebagaimana terlihat pada koreksi dari 30-30 menjadi 30-25 yang mengembalikan pergerakan menjadi "Maju Lurus". Hasil ini juga menunjukkan bahwa perbedaan signifikan pada nilai PWM antara kedua motor, seperti 100-80 atau 80-100, menyebabkan deviasi ke arah motor dengan PWM lebih rendah, sementara perbedaan PWM yang moderat (seperti 100-90 atau 90-80) masih dapat mempertahankan lintasan lurus.

3.5 Implementasi Sistem

Implementasi sistem dalam penelitian ini melibatkan realisasi dari desain konseptual ke dalam bentuk fisik dan perangkat lunak yang terintegrasi secara menyeluruh. Sistem dikembangkan pada platform *Raspberry Pi* sebagai *unit* pemrosesan utama, yang menjalankan algoritma *computer vision* dan pengambilan

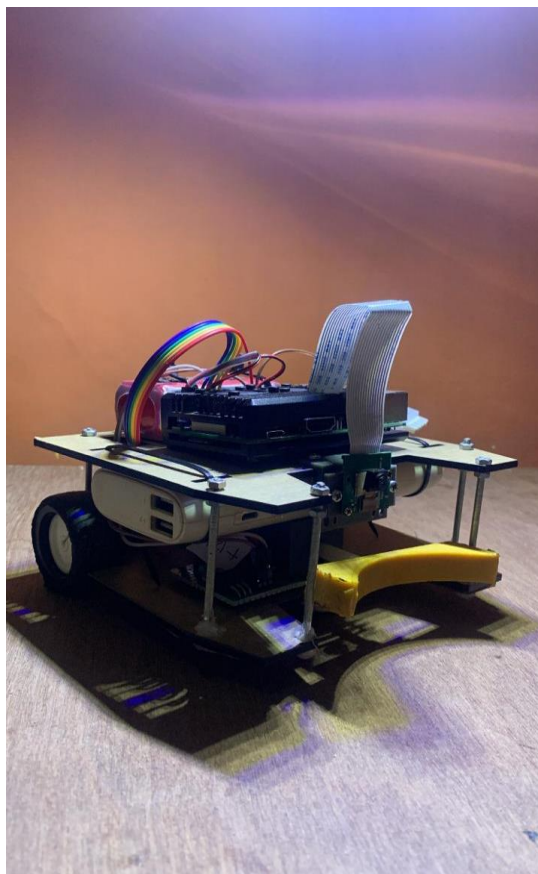
keputusan secara *real-time*. Komponen utama sistem meliputi kamera *Raspberry Pi* sebagai sensor visual, dua buah motor DC sebagai aktuator penggerak utama, solenoid sebagai aktuator penendang bola, serta driver motor L298N untuk mengatur arah dan kecepatan pergerakan motor berdasarkan sinyal *Pulse Width Modulation* (PWM). Realisasi desain robot dapat dilihat pada Gambar 3.9.

Pada perangkat keras, rangka robot dibuat menggunakan bahan akrilik berukuran $145 \text{ mm} \times 100 \text{ mm}$ yang dirancang untuk memuat rangkaian elektronik. Kamera *Raspberry Pi* dipasang pada bagian depan robot untuk menangkap citra arena permainan secara terus-menerus. Citra yang diperoleh kemudian diolah menggunakan metode *Hough Circle Transform* guna mendeteksi bola berdasarkan bentuk lingkarannya. Hasil deteksi berupa koordinat pusat dan radius bola dikonversi menjadi data koordinat posisi relatif serta estimasi jarak bola terhadap robot menggunakan persamaan proyeksi perspektif. Data koordinat tersebut menjadi dasar dalam pengambilan keputusan gerak.

Motor DC dikendalikan menggunakan driver L298N yang menerima sinyal PWM dari *Raspberry Pi* untuk menentukan kecepatan dan arah gerak robot. Konfigurasi PWM kanan dan kiri berdasarkan posisi horizontal bola terhadap kamera. Penyesuaian nilai PWM dilakukan agar robot dapat berbelok secara halus maupun cepat, tergantung dari seberapa jauh posisi bola dari pusat pandang kamera. Selain itu, kecepatan maju ditentukan berdasarkan jarak bola. Semakin dekat bola terhadap robot, maka nilai PWM yang diberikan semakin besar.

Sistem ini juga dilengkapi dengan mekanisme pencarian bola dan aksi menendang bola. Ketika sistem tidak dapat mendeteksi objek bola karena bola

berada di luar frame, robot akan masuk ke mode pencarian. Pada mode ini, motor dc dikonfigurasi agar robot berputar ke arah kanan secara perlahan. Rotasi ini dilakukan secara terus-menerus hingga sistem kembali mendeteksi bola dalam frame kamera. Ketika bola berada sangat dekat dengan robot ($z \leq 15$ cm), sistem akan mengeksekusi aksi menendang menggunakan solenoid. Solenoid dikendalikan melalui *output GPIO Raspberry Pi* yang dikonfigurasi untuk memberikan tegangan tinggi (*HIGH*) selama periode singkat untuk mendorong bola secara cepat ke depan. Setelah aksi selesai, pin *GPIO* dikembalikan ke kondisi rendah (*LOW*) untuk menghindari aktivitas berlebih.



Gambar 3. 9 Realisasi Desain Robot

BAB IV

UJI COBA DAN PEMBAHASAN

Bab ini membahas skenario uji coba yang dirancang untuk mengevaluasi performa sistem dalam mendeteksi dan mengikuti arah gerak bola. Uji coba dilakukan untuk mengukur performa metode *Hough Circle Transform* dalam mendeteksi bola serta mengontrol pergerakan robot. Analisa hasil pengujian dilakukan untuk menentukan keandalan sistem, dilengkapi dengan pembahasan mengenai kelebihan, kendala, dan potensi perbaikan yang dapat dilakukan.

4.1 Skenario Uji

Pengujian dilakukan dalam lingkungan yang terkendali dengan beberapa skenario yang mencakup pengukuran akurasi deteksi bola, perhitungan jarak bola terhadap robot, serta waktu *respons* robot terhadap pergerakan bola. Setiap skenario dirancang untuk menguji keandalan sistem dalam berbagai kondisi, termasuk perubahan posisi bola dan pencahayaan yang stabil. Skenario uji dapat dilihat pada Tabel 4.1.

4.1.1 Skenario Pengujian *Error* Deteksi Bola

Pengujian *error* deteksi bola dilakukan dengan meletakkan bola pada beberapa jarak tetap yang telah ditentukan dari robot. Tujuan dari pengujian ini adalah untuk mengukur *error* deteksi bola, dimana representasi dari lingkaran, pada metode *Hough Circle Transform*. Perhitungan dilakukan berdasarkan selisih data

aktual dengan peramalan dibagi dengan data aktual kemudian diubah menjadi nilai absolut, seperti pada Persamaan 4.1.

$$MAPE (\%) = \frac{1}{n} \sum \left| \left(\frac{A_t - F_t}{A_t} \right) \right| \cdot 100\% \quad (4.1)$$

Keterangan:

n : jumlah data
 A_t : nilai aktual
 F_t : nilai perkiraan

Nilai akurasi *Mean Absolute Percentage Error* (MAPE) merupakan perhitungan kesalahan suatu prediksi untuk mendapatkan nilai rata – rata persentase *error* mutlak (Maricar, 2019). Rumus ini menghitung persentase *error* deteksi yang terjadi pada saat pengujian. Selain itu, jika bola tidak terdeteksi, robot akan masuk ke mode pencarian bola dimana robot akan berputar ke arah kanan hingga bola kembali terdeteksi oleh sistem. Pencarian bola dilakukan untuk mengatasi situasi dimana bola hilang dari *frame* kamera, seperti ketika bola bergerak keluar dari jangkauan penglihatan kamera. Pengujian ini mengukur seberapa cepat sistem dapat beralih ke mode pencarian bola ketika deteksi gagal dan kembali melanjutkan tugasnya setelah bola ditemukan. Seluruh percobaan akan mencatat waktu yang dibutuhkan robot untuk menemukan bola kembali serta mengukur *error* deteksi yang terjadi saat bola terdeteksi.

4.1.2 Skenario Pengujian *Error* Posisi Bola

Pada tahap ini, pengujian dilakukan dengan mengukur jarak antara objek bola dengan kamera. Tujuan dari pengujian ini adalah untuk mengukur selisih antara deteksi posisi atau jarak bola yang sebenarnya dengan jarak bola yang terdeteksi oleh sistem. Perhitungan dilakukan berdasarkan selisih data aktual (jarak

sebenarnya) dengan peramalan (jarak terdeteksi) dibagi dengan data aktual kemudian diubah menjadi nilai absolut, seperti pada Persamaan 4.1 yang sudah dijelaskan di atas. Deteksi posisi bola pada sistem dilakukan menggunakan Persamaan 4.2 berikut.

$$Z (cm) = \frac{C}{P} \quad (4.2)$$

Keterangan:

Z : jarak objek ke kamera
 P : diameter proyeksi objek dalam piksel ($2 \times radius$)
 C : Constanta

Dimana Z merupakan jarak objek ke kamera, P adalah diameter proyeksi objek dalam piksel ($2 \times radius$), dan C adalah *constant*. *Constanta* diperoleh dari proses kalibrasi kamera. Konstanta ini mewakili hubungan antara ukuran sebenarnya objek dengan proyeksinya pada citra kamera dalam satuan piksel. Perhitungan konstanta dilakukan menggunakan rumus Persamaan 4.3.

$$C = W . F \quad (4.3)$$

Keterangan:

W : diameter bola sebenarnya
 F : *focal length* kamera

Dimana W merupakan diameter objek bola yang sebenarnya dan F adalah *focal length* kamera dalam satuan piksel. *Focal length* kamera dalam satuan piksel didapatkan dari rumus pada Persamaan 4.4.

$$F = \frac{focal\ length\ (mm)}{panjang\ sensor} \times resolusi\ horizontal \quad (4.4)$$

Focal length Raspberry Pi Camera dalam satuan milimeter adalah 3,60 mm dan panjang sensornya yaitu 3,76 mm. Resolusi horizontal yang digunakan yaitu 640 px. Nilai dalam *focal length* dan panjang sensor diperoleh dari dokumentasi resmi *Raspberry Pi* (*Raspberry Pi Documentation*, 2024).

Pada jarak yang sangat dekat ($z \leq 15$ cm), robot diharapkan dapat melakukan aksi menendang bola menggunakan solenoid, yang diaktifkan hanya setelah jarak bola dipastikan sangat dekat. Ketepatan dalam pengukuran jarak bola sangat penting karena aksi menendang bola hanya dilakukan saat jarak bola benar-benar mencapai posisi yang telah ditentukan. Pengujian ini juga melibatkan pengukuran *error* jarak dan apakah robot dapat melakukan aksi menendang dengan tepat saat bola berada di depan robot.

4.1.3 Skenario Pengujian Waktu Respons Robot

Waktu respons robot diukur dengan menghitung selisih waktu pada saat bola terdeteksi dan saat robot mulai bergerak. Rumus waktu respons yang digunakan seperti pada Persamaan 4.5 (E. S. Rahman dkk., 2023). Pengujian ini bertujuan untuk mengukur seberapa cepat respons robot atas pergerakan bola.

$$\Delta t (s) = t_{max} - t_{min} \quad (4.5)$$

Keterangan:

Δt : selisih waktu
 t_{max} : waktu tertinggi
 t_{min} : waktu terendah

4.2 Hasil Pengujian

Pengujian sistem dilakukan untuk mengevaluasi performa *robot soccer* dalam mendeteksi dan mengikuti bola secara otomatis. Pengujian dilakukan dengan

menghitung akurasi deteksi bola, kesalahan posisi deteksi atau jarak bola terhadap robot, dan waktu *respons* robot terhadap pergerakan bola. Tujuan pengujian ini adalah untuk mengukur kinerja algoritma *Hough Circle Transform* dalam mendeteksi bola dan seberapa cepat robot merespons deteksi tersebut. Pengujian dilakukan berdasarkan perbandingan antara nilai aktual dengan hasil pengukuran yang dihasilkan oleh sistem, serta menggunakan persentase kesalahan untuk mengukur deviasi hasil. Hasil pengujian ini digunakan untuk menilai kemampuan robot dalam mendeteksi bola dan mengikuti pergerakannya secara otomatis tanpa intervensi manusia. Sistem dianggap berhasil dalam mengontrol pergerakan robot jika nilai kesalahan berada dalam batas toleransi yang dapat diterima. Pengujian ini penting untuk memastikan bahwa metode *Hough Circle Transform* dapat diimplementasikan dengan baik dalam *robot soccer* untuk mendukung kinerja dalam kompetisi maupun skenario nyata lainnya.

4.2.1 Pengujian *Error* Deteksi Bola

Pengujian ini dilakukan untuk mengevaluasi performa sistem dalam mendeteksi bola menggunakan algoritma *Hough Circle Transform* pada berbagai jarak pengamatan. Tujuan utama dari pengujian ini adalah untuk mengukur persentase kesalahan deteksi bola yang berbentuk lingkaran pada citra digital, serta mengidentifikasi sejauh mana sistem mampu mempertahankan konsistensi deteksi dalam kondisi yang berbeda. Pengujian dilakukan sebanyak lima kali pada setiap jarak yang telah ditentukan, yaitu 50 cm, 100 cm, 150 cm, 200 cm, 250 cm, dan 300 cm sehingga total percobaan sebanyak 30 kali. Perhitungan *error* deteksi mengacu pada rumus *Mean Absolute Percentage Error* (MAPE) seperti yang dijelaskan pada

Persamaan 4.1. Dalam pengujian ini, keberhasilan deteksi dicatat setiap kali sistem mampu menandai keberadaan bola dalam frame citra. Sebaliknya, apabila bola tidak berhasil terdeteksi dalam citra meskipun secara fisik bola berada di posisi yang sesuai, maka kondisi tersebut diklasifikasikan sebagai kegagalan deteksi.

Tabel 4. 1 Perhitungan *Error* Deteksi Bola

No.	Jarak (cm)	Total Percobaan	Deteksi Berhasil	Error (%)
1.	50	5	5	0
2.	100	5	5	0
3.	150	5	5	0
4.	200	5	5	0
5.	250	5	4	20
6.	300	5	3	40
Rata - Rata				10

Berdasarkan Tabel 4.1, hasil pengujian berada pada tingkat optimal untuk jarak antara 50 cm hingga 200 cm dengan persentase *error* sebesar 0%. Hal ini menunjukkan bahwa sistem mampu mendeteksi bola secara konsisten pada rentang jarak menengah. Namun, pada jarak 250 cm, terjadi penurunan performa dengan *error* sebesar 20%, dan pada jarak 300 cm *error* meningkat hingga 40%. Rata-rata *error* keseluruhan dari seluruh pengujian masih berada pada angka yang relatif rendah, yaitu 10%. Berikut salah satu contoh perhitungan *error* pada jarak 300 cm.

$$Error = \left| \frac{Total\ Percobaan - Deteksi\ Berhasil}{Total\ Percobaan} \right| \times 100\%$$

$$Error = \left| \frac{5 - 3}{5} \right| \times 100\%$$

$$Error = 40\%$$

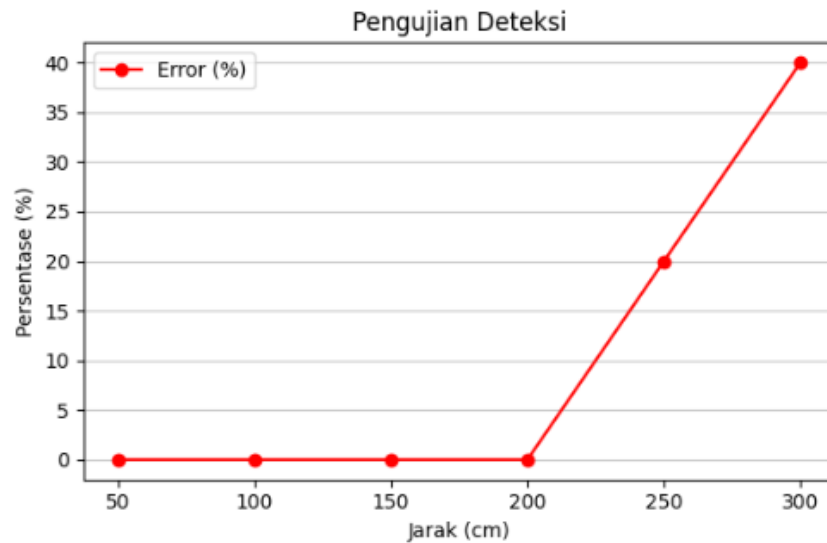
Hasil persentase yang diperoleh dari perhitungan *error* menggunakan data yang terakhir, pada jarak 300 cm, adalah 40%. Setelah itu, dilakukan perhitungan rata-rata menggunakan rumus pada Persamaan 4.1.

$$Rata - Rata Error = \frac{1}{n} \sum \left| \left(\frac{Total Percobaan - Deteksi Berhasil}{Total Percobaan} \right) \right| \times 100\%$$

$$Rata - Rata Error = \frac{60}{6}$$

$$Rata - Rata Error = 10\%$$

Selain mengukur keberhasilan deteksi dalam kondisi optimal, sistem juga diuji terhadap kondisi ketika bola berada di luar jangkauan kamera karena bola bergerak ke luar frame. Dalam kondisi tersebut, sistem akan secara otomatis mengaktifkan mode pencarian bola, dimana robot akan berputar ke arah kiri secara perlahan untuk mencari kembali keberadaan bola hingga berhasil terdeteksi. Grafik hasil pengujian dapat dilihat pada Gambar 4.1 yang menggambarkan tingkat *error* deteksi pada masing-masing jarak pengujian. Grafik tersebut menunjukkan tren peningkatan *error* seiring bertambahnya jarak antara robot dan bola yang mengindikasikan keterbatasan algoritma dalam mendeteksi objek dengan ukuran proyeksi yang semakin kecil.

Gambar 4. 1 Grafik *Error* Deteksi

4.2.2 Pengujian *Error* Posisi Bola

Pengujian posisi atau jarak bola terhadap robot dilakukan untuk menilai kinerja sistem dalam mengukur jarak bola berdasarkan deteksi proyeksi lingkaran pada citra digital. Sistem memanfaatkan algoritma Hough Circle Transform untuk mendeteksi lingkaran bola dan menghitung jarak objek berdasarkan ukuran diameter bola yang terbentuk dalam citra. Perhitungan dilakukan menggunakan rumus proyeksi perspektif sebagaimana dijelaskan pada Persamaan 4.2 hingga 4.4. Dalam pengujian ini, bola diletakkan pada enam jarak berbeda dari robot yakni 50 cm, 100 cm, 150 cm, 200 cm, 250 cm, dan 300 cm. Pada masing-masing jarak dilakukan lima kali percobaan dan menghasilkan total 30 data pengujian.

Tabel 4. 2 Perhitungan *Error* Jarak

No.	Jarak Sebenarnya (cm)	Jarak Terdeteksi (cm)	Error (%)
1.	50	48	4
2.		51	2
3.		53	6
4.		48	4
5.		52	4
6.	100	98	2
7.		103	3

No.	Jarak Sebenarnya (cm)	Jarak Terdeteksi (cm)	Error (%)
8.		97	3
9.		99	1
10.		102	2
11.	150	153	2
12.		148	1,3
13.		144	4
14.		152	1,3
15.		146	2,6
16.	200	204	2
17.		206	3
18.		197	1,5
19.		195	2,5
20.		196	3
21.	250	258	3,2
22.		243	2,8
23.		241	3,6
24.		256	2,4
25.		255	2
26.	300	287	4,3
27.		277	7,6
28.		285	5
29.		281	6,3
30.		286	4,6
Rata – Rata			3,22

Hasil pengujian pada Tabel 4.2 yang menunjukkan nilai jarak sebenarnya, jarak yang terdeteksi oleh sistem, dan persentase error untuk setiap percobaan. Berdasarkan hasil tersebut, dapat dihitung nilai rata-rata *error* untuk setiap jarak, yang kemudian digunakan sebagai representasi performa sistem dalam grafik. Rata-rata *error* tertinggi terjadi pada jarak 300 cm dengan persentase 4,3%, sedangkan error terkecil dicapai pada jarak 100 cm dengan nilai sebesar 2%. Berikut salah satu contoh perhitungan *error* pada percobaan ke-15 yaitu pada jarak 150 cm.

$$Error = \left| \frac{Jarak\ Sebenarnya - Jarak\ Terdeteksi}{Jarak\ Sebenarnya} \right| \times 100\%$$

$$Error = \left| \frac{150 - 146}{150} \right| \times 100\%$$

$$Error = 2,6\%$$

Hasil persentase yang diperoleh dari perhitungan *error* menggunakan data pada jarak 150 cm adalah 2,6%. Setelah dilakukan perhitungan persentase *error* pada setiap jarak, kemudian dilakukan perhitungan rata-rata *error* keseluruhan menggunakan rumus pada Persamaan 4.1.

$$Rata - Rata Error = \frac{1}{n} \sum \left| \left(\frac{Jarak Sebenarnya - Jarak Terdeteksi}{Jarak Sebenarnya} \right) \right| \times 100\%$$

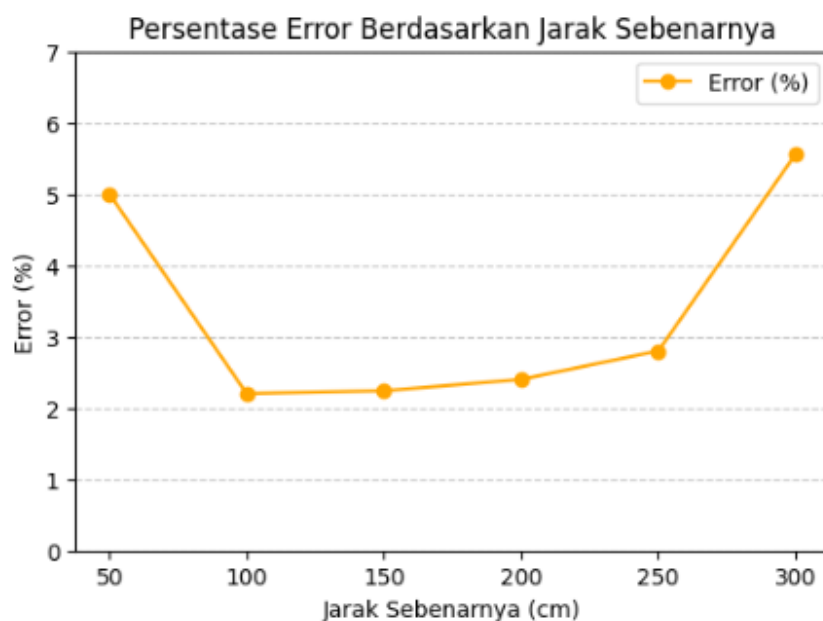
$$Rata - Rata Error = \frac{96,7}{30}$$

$$Rata - Rata Error = 3,22\%$$

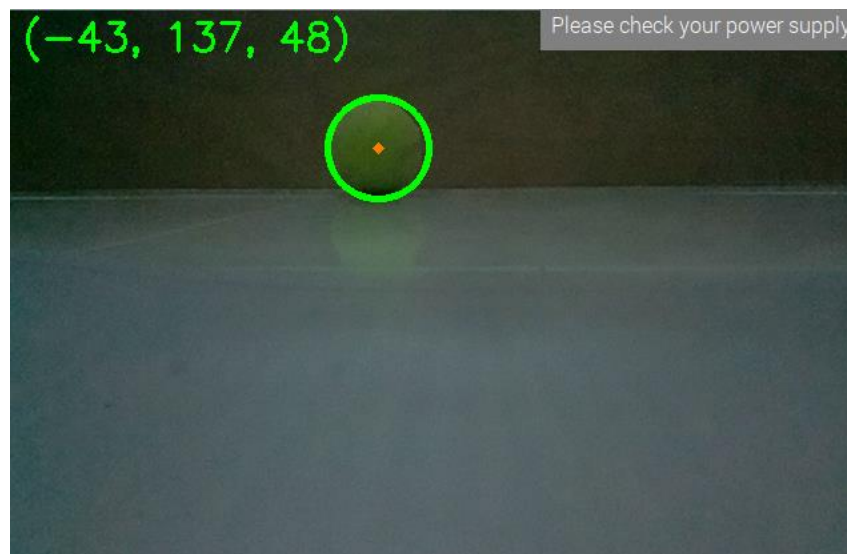
Secara keseluruhan, rata-rata *error* pada seluruh pengujian adalah 3,22%. Hal ini menunjukkan bahwa sistem memiliki performa yang baik dalam mendeteksi jarak bola. *Error* yang kecil ini menunjukkan keandalan algoritma *Hough Circle Transform* dalam mendeteksi proyeksi bola pada citra digital. Grafik hasil pengukuran jarak dapat dilihat pada Gambar 4.2 dengan rata-rata *error* berdasarkan setiap jarak uji yang terdiri dari enam titik pengamatan utama. Grafik ini memudahkan dalam mengidentifikasi tren performa pengukuran seiring bertambahnya jarak antara bola dan kamera. Terlihat bahwa sistem cenderung memiliki performa yang lebih baik pada jarak menengah (100 cm hingga 200 cm), dan sedikit menurun pada jarak yang sangat dekat maupun sangat jauh, disebabkan oleh distorsi perspektif dan keterbatasan resolusi proyeksi.

Selain itu, pada kondisi jarak sangat dekat (≤ 15 cm), sistem dirancang untuk melakukan aksi menendang bola menggunakan aktuator *solenoid*. Aktivasi solenoid dilakukan berdasarkan hasil pengukuran jarak bola. Jika sistem mendeteksi bahwa bola berada dalam ambang jarak pendek, maka *GPIO Raspberry*

Pi akan diaktifkan untuk memberikan sinyal *HIGH* ke *solenoid* yang mendorong bola ke depan secara mekanis. Keberhasilan sistem dalam mengaktifkan solenoid sangat bergantung pada ketepatan pengukuran jarak sehingga pengujian ini tidak hanya mengevaluasi performa estimasi jarak, tetapi juga validasi terhadap respon aktuator berdasarkan jarak terukur. Dalam seluruh pengujian pada zona jarak dekat, solenoid berhasil diaktifkan secara tepat waktu dan sesuai dengan logika sistem, menunjukkan bahwa fungsi ini telah terintegrasi dengan baik dalam skema kontrol robot secara menyeluruh.



Gambar 4. 2 Grafik *Error* Jarak Bola



Gambar 4. 3 Visualisasi Jarak Bola

Gambar 4.3 menunjukkan visualisasi jarak yang terdeteksi oleh sistem dalam proses pengukuran jarak bola terhadap robot. Dalam pengujian ini, visualisasi digunakan untuk menggambarkan hubungan antara posisi bola yang terdeteksi dalam citra dan jarak sebenarnya dari bola ke kamera. Visualisasi ini juga menggambarkan perbedaan antara nilai jarak yang terdeteksi dan jarak aktual bola yang diukur secara fisik. Pada gambar tersebut, bola yang terdeteksi oleh sistem diwakili oleh lingkaran yang digambar pada citra hasil deteksi. Sistem mengukur posisi bola berdasarkan proyeksi diameter bola dalam citra dan menghitung jaraknya menggunakan rumus proyeksi perspektif yang dijelaskan pada Persamaan 4.3 dan 4.4. Nilai jarak yang terdeteksi diwakili dengan anotasi yang menunjukkan seberapa jauh bola dari kamera, yang dikalkulasikan berdasarkan ukuran diameter bola dalam citra dan panjang fokus kamera. Visualisasi jarak yang terdeteksi ini memberikan pemahaman yang lebih jelas tentang kemampuan sistem dalam memetakan objek fisik ke dalam ruang digital, serta memberikan gambaran bahwa

sistem dapat memonitor pergerakan bola dan membuat keputusan berbasis jarak dengan performa yang cukup baik.

4.2.3 Pengujian Waktu Respons Robot

Pengujian waktu respons robot bertujuan untuk mengukur seberapa cepat sistem dapat merespons keberadaan bola setelah objek terdeteksi. Respons yang cepat dan konsisten merupakan salah satu indikator penting dari keberhasilan implementasi algoritma deteksi bola dan sistem kontrol robot secara keseluruhan. Pengukuran waktu respon dilakukan dengan menghitung selisih waktu saat bola pertama kali terdeteksi oleh kamera dan waktu saat robot mulai bergerak. Selisih keduanya dihitung sebagai durasi waktu respons sistem. Perhitungan ini merujuk pada rumus dalam Persamaan 4.5. Pengujian dilakukan sebanyak 30 kali pada enam variasi jarak, yakni 50 cm, 100 cm, 150 cm, 200 cm, 250 cm, dan 300 cm dengan masing-masing lima kali pengamatan. Hasil dari pengujian ini disajikan pada Tabel 4.3 yang menunjukkan data waktu deteksi, waktu awal gerak, dan durasi waktu respons untuk setiap percobaan.

Tabel 4. 3 Perhitungan Waktu *Respons* Robot

Percobaan ke-	Jarak (cm)	Awal Gerak (s)	Terdeteksi (s)	Waktu (s)
1	50	1,1	0,4	0,7
2		1,7	0,5	1,2
3		1,5	0,7	0,8
4		1,2	0,6	0,6
5		2,1	1,2	0,9
6	100	2,0	0,7	1,3
7		2,5	1,5	1,0
8		1,7	1,2	0,5
9		1,6	0,9	0,7
10		3,0	1,7	1,3
11	150	1,4	0,8	0,6
12		1,9	1,0	0,9
13		1,6	0,7	0,9
14		1,3	0,5	0,8
15		2,0	1,3	0,7

16	200	2,2	1,4	0,8
17		1,8	1,1	0,7
18		1,7	1,0	0,7
19		2,1	1,3	0,8
20		1,9	1,2	0,7
21	250	3,4	0,8	2,6
22		3,0	1,0	2,0
23		3,1	0,9	2,2
24		3,3	1,2	2,1
25		3,5	1,1	2,4
26	300	3,8	1,2	2,6
27		3,6	1,4	2,2
28		3,9	1,3	2,6
29		3,7	1,1	2,6
30		3,5	1,5	2,0
Rata – Rata				1,33

Berdasarkan Tabel 4.4, hasil pengujian menunjukkan bahwa waktu respons robot terhadap bola berada dalam rentang antara 0,5 hingga 1,3 detik. Pada jarak yang lebih dekat seperti 50 cm dan 100 cm, sistem menunjukkan waktu respons yang relatif lebih cepat yang mengindikasikan bahwa ukuran proyeksi bola yang lebih besar dalam citra dapat mempermudah proses deteksi dan pemrosesan keputusan gerak. Sementara itu, pada jarak yang lebih jauh seperti 250 cm dan 300 cm, waktu respons cenderung sedikit meningkat yang disebabkan oleh waktu komputasi tambahan yang dibutuhkan untuk mendeteksi objek dengan ukuran proyeksi yang lebih kecil dan kurang kontras. Berikut merupakan salah satu contoh perhitungan waktu *respons* robot terhadap gerak bola.

$$\text{Waktu Respons} = t_{\text{awal gerak}} - t_{\text{terdeteksi}}$$

$$\text{Waktu Respons} = 1,1 - 0,4$$

$$\text{Waktu Respons} = 0,7 \text{ s}$$

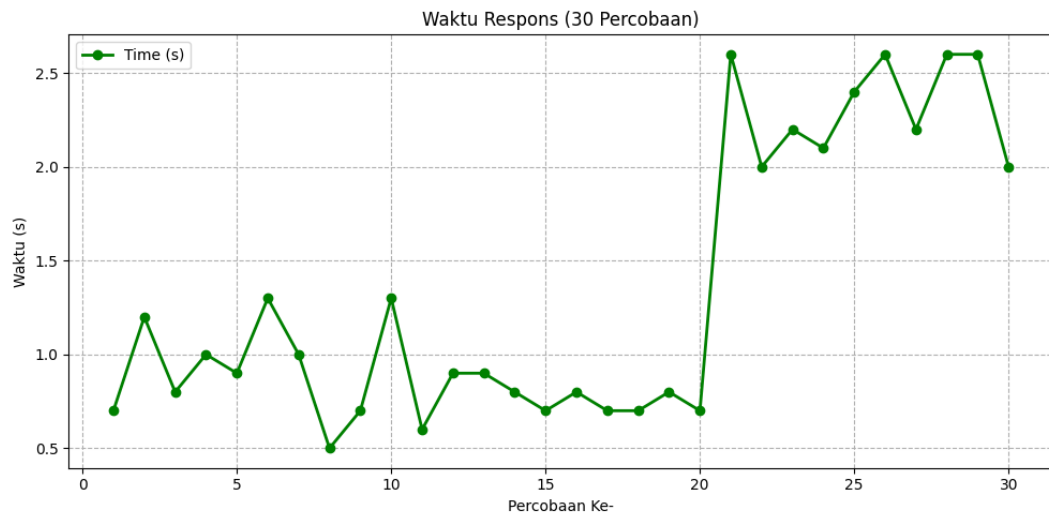
Hasil yang diperoleh dari perhitungan waktu *respons* menggunakan data pada pertama adalah 0,7 detik. Setelah dilakukan perhitungan pada setiap percobaan, kemudian dilakukan perhitungan rata-rata waktu *respons* (Ariyani & La Djamudi, 2023).

$$\text{Rata – Rata Waktu Respons} = \frac{\text{Total Waktu Respons}}{\text{Jumlah Percobaan}}$$

$$\text{Rata – Rata Waktu Respons} = \frac{39,9}{30}$$

$$\text{Rata – Rata Waktu Respons} = 1,33 \text{ s}$$

Rata-rata waktu *respons* sebesar 1,33 detik menunjukkan bahwa robot memiliki kemampuan yang baik dalam merespons posisi atau gerak bola dengan waktu yang relatif singkat. Grafik hasil pengujian ditampilkan pada Gambar 4.4. Grafik ini memperlihatkan perbandingan rata-rata waktu *respons* pada masing-masing jarak bola. Terlihat bahwa waktu *respons* mengalami sedikit peningkatan pada jarak 250 cm dan 300 cm dibandingkan dengan jarak sebelumnya. Hal ini disebabkan oleh kompleksitas pemrosesan visual pada objek dengan proyeksi yang lebih kecil serta potensi penundaan akibat noise visual yang lebih tinggi pada jarak yang jauh.



Gambar 4. 4 Grafik Waktu Respons Robot

4.3 Pembahasan

Hasil pengujian yang telah dilakukan pada sistem *robot soccer* menunjukkan bahwa algoritma *Hough Circle Transform* memiliki performa yang cukup baik dalam mendeteksi bola, mengestimasi jarak, dan merespons keberadaan bola melalui pergerakan serta aksi mekanis. Setiap pengujian yang telah dilakukan memberikan kontribusi terhadap penilaian efektivitas integrasi algoritma pengolahan citra dengan sistem kendali robot berbasis *Raspberry Pi*.

Hasil pengujian *error* deteksi bola menunjukkan performa yang tinggi dalam mendeteksi bola pada jarak dengan rata rata *error* 10%. Hal ini menandakan bahwa metode *Hough Circle Transform* mampu mengidentifikasi objek berbentuk lingkaran seperti bola dengan akurasi tinggi secara optimal. Sistem dapat mengenali pola tepi bola dengan lebih jelas pada jarak dekat hingga menengah sehingga proses deteksi berjalan lebih akurat dan stabil. *Error* deteksi cenderung meningkat pada jarak jauh yang disebabkan oleh ukuran bola dalam citra yang semakin kecil serta pengaruh pencahayaan dan keterbatasan resolusi kamera sehingga lingkaran atau

bola lebih sulit untuk dikenali oleh sistem. Ketika bola tidak berada dalam jangkauan kamera, sistem berhasil beralih ke mode pencarian dengan melakukan pergerakan rotasi supaya dapat mendeteksi bola kembali. Hal ini menunjukkan bahwa sistem memiliki mekanisme pemulihan deteksi yang adaptif.

Hasil pengujian *error* posisi bola terhadap robot menunjukkan kemampuan yang cukup baik dalam mengestimasi jarak bola dari robot. Rata-rata *error* dari seluruh pengujian adalah 3,22% yang menunjukkan bahwa estimasi jarak tergolong cukup akurat. Estimasi jarak digunakan untuk menentukan nilai PWM dasar yang akan mengatur kecepatan robot dalam mendekati bola. Grafik pada Gambar 4.2 menunjukkan kecenderungan peningkatan *error* secara bertahap seiring bertambahnya jarak. Hal ini disebabkan oleh berkurangnya akurasi proyeksi perspektif pada objek yang lebih jauh dari kamera yang mengakibatkan estimasi jarak menjadi kurang presisi. Meski demikian, nilai *error* yang diperoleh masih dalam batas toleransi untuk sistem berbasis pengolahan citra tanpa dukungan sensor jarak tambahan. Keandalan sistem dalam pengukuran jarak juga berpengaruh terhadap keberhasilan aktivasi solenoid untuk aksi menendang bola. Ketika bola berada dalam jarak yang sangat dekat (≤ 15 cm), sistem secara otomatis mengaktifkan solenoid berdasarkan hasil estimasi jarak yang dilakukan. Keberhasilan ini menunjukkan bahwa koordinasi antara unit visual (kamera dan algoritma deteksi) dengan unit aktuator (solenoid) telah berjalan dengan sinkron yang menjadikan robot tidak hanya mampu mendekati bola tetapi juga melakukan aksi fisik terhadap objek secara mandiri. Integrasi antara pengukuran jarak dan

logika aktivasi aktuator menjadi bukti bahwa robot mampu melakukan aksi secara *real-time*.

Hasil pengujian waktu respons robot terhadap bola menunjukkan bahwa sistem memiliki waktu respons yang relatif cepat. Rata-rata waktu respons dari seluruh pengujian adalah 1,33 detik. Grafik pada Gambar 4.3 menunjukkan bahwa semakin jauh posisi bola dari robot, semakin tinggi waktu yang dibutuhkan oleh sistem untuk memproses informasi dan menggerakkan motor. Hal ini disebabkan oleh semakin kecil ukuran proyeksi bola pada jarak jauh yang membutuhkan proses verifikasi visual yang lebih kompleks. Meski demikian, rata-rata waktu respons membuktikan bahwa sistem ini layak diterapkan untuk robot yang membutuhkan reaksi cepat.

Hasil penelitian ini dapat dibandingkan dengan studi oleh Putra & Puriyanto (2022) yang mengimplementasikan metode *Hough Circle Transform* pada robot sepak bola beroda dengan kamera *omnidirectional*. Sistem tersebut mencapai akurasi 100% dengan *error* koordinat rata-rata 0,14%, lebih rendah dibandingkan *error* 3,22% dalam penelitian ini. Keunggulan tersebut terkait dengan penggunaan kamera *omnidirectional* dan implementasi *colour filtering*. Meski demikian, waktu *respons* sistem dalam penelitian ini yang mencapai 1,33 detik menunjukkan performa yang memadai untuk aplikasi *real-time* dalam pertandingan *robot soccer*.

Secara keseluruhan, hasil pengujian tersebut saling melengkapi dalam menggambarkan performa sistem secara komprehensif. Deteksi visual yang akurat pada jarak optimal, pengukuran jarak yang mendekati nilai aktual, serta respons gerak dan aksi fisik yang sinkron menunjukkan bahwa integrasi antara perangkat

lunak (algoritma deteksi dan kontrol) dengan perangkat keras (kamera, motor, dan solenoid) telah berjalan secara efektif. Meskipun masih terdapat beberapa keterbatasan dalam deteksi pada jarak jauh dan potensi kehilangan objek dari bidang pandang kamera, sistem telah dirancang dengan fitur seperti mode pencarian otomatis dan pengambilan keputusan berbasis zona. Implementasi algoritma *Hough Circle Transform* dalam sistem *robot soccer* telah memberikan hasil yang optimal dalam mendukung skenario permainan, baik untuk kompetisi maupun aplikasi nyata lainnya.

Penelitian ini mengintegrasikan nilai-nilai Islam ke dalam proses pengembangan teknologi. Ayat Al-Qur'an sebagai landasan reflektif dalam memahami keteraturan sistem serta batasan logika yang berlaku dalam rancangan otomatisasi robot. Integrasi teknologi ini dengan nilai-nilai Islam terlihat dari konsep keteraturan yang juga tercermin dalam sunnatullah, yaitu hukum-hukum alam yang telah ditetapkan oleh Allah sebagaimana dijelaskan dalam Surah Yasin ayat 40:

لَا الشَّمْسُ يَنْبَغِي لَهَا أَنْ تُدْرِكَ الْقَمَرَ وَلَا اللَّيْلُ سَابِقُ النَّهَارِ ۚ وَكُلٌّ فِي فَلَكٍ يَسْبَحُونَ

"Tidaklah mungkin bagi matahari mengejar bulan, dan malam pun tidak mendahului siang. Masing-masing beredar pada garis edarnya" (QS. Yasin: 40).

Tafsir ayat ini menegaskan bahwa Allah telah menciptakan dan mengatur alam semesta dengan hukum-hukumnya sehingga semua benda langit berjalan dalam keteraturan yang sempurna sesuai garis edar masing-masing (Kemenag, 2022c). Hal ini dapat dianalogikan dalam sistem *robot soccer* yang memiliki batasan logika gerak dan ambang pengambilan keputusan. Sistem yang dirancang untuk

menendang bola hanya jika bola berada dalam jarak tertentu dan posisi visual berada di tengah, berarti ada batasan logis yang tidak boleh dilanggar oleh sistem agar tidak menghasilkan kesalahan aksi. Sistem ini harus menunggu urutan kondisi yang sesuai sebelum melakukan tindakan, sama seperti matahari dan bulan yang tidak saling mendahului. Hal ini menekankan pentingnya *order*, *sequence*, dan *batas-batas logis* dalam pengambilan keputusan sebagaimana nilai yang terkandung dalam ayat tersebut.

Kemudian, integrasi teknologi dengan nilai-nilai Islam juga diperkuat melalui pemaknaan terhadap Surah As-Sajdah ayat 5:

يُدَبِّرُ الْأَمْرَ مِنَ السَّمَاءِ إِلَى الْأَرْضِ ثُمَّ يَعْرُجُ إِلَيْهِ فِي يَوْمٍ كَانَ مِقْدَارُهُ أَلْفَ سَنَةٍ مِّمَّا تَعُدُّونَ

"Dia mengatur urusan dari langit ke bumi, kemudian (urusan) itu naik kepada-Nya dalam satu hari yang kadarnya (lamanya) adalah seribu tahun menurut perhitunganmu" (QS. As-Sajdah: 5).

Tafsir ayat ini menyatakan bahwa Allah adalah pengatur segala urusan di alam semesta, dari langit hingga bumi, dan tidak ada sesuatu pun yang menyimpang dari kehendaknya. Segala yang terjadi di alam raya mengikuti aturan dan ketetapan Allah (Kemenag, 2022a). Dalam sistem kontrol robot ini, prinsip *tadbīr al-amr* tercermin dalam alur kerja sistem kontrol. Mulai dari input citra, analisis tepi, deteksi lingkaran, estimasi jarak, hingga pengambilan keputusan yang semuanya dijalankan secara bertahap dan berjenjang. Pengaturan urusan dalam ayat ini mencerminkan pentingnya struktur dan urutan, dimana semua proses dalam sistem robot memiliki alur yang logis dan tidak tumpang tindih. Apabila salah satu tahapan tidak berjalan sebagaimana mestinya, maka keseluruhan sistem akan terganggu.

Oleh karena itu, keteraturan dan kendali berjenjang merupakan bentuk teknis dari nilai-nilai yang disebutkan dalam ayat ini.

Selanjutnya, integrasi teknologi dengan nilai-nilai Islam yang relevan dengan topik penelitian ini terkandung dalam Surah Az-Zariyat ayat 49:

وَمِنْ كُلِّ شَيْءٍ خَلَقْنَا زَوْجَيْنِ لَعَلَّكُمْ تَذَكَّرُونَ

“Dan segala sesuatu Kami ciptakan berpasang-pasangan agar kamu mengingat (kebesaran Allah)” (QS. Az-Zariyat: 49).

Menurut penafsiran Jalalain, makna dari konsep berpasang-pasangan dalam ayat ini yaitu mencakup pasangan-pasangan ciptaan seperti laki-laki dan perempuan, langit dan bumi, matahari dan bulan, dataran tinggi dan dataran rendah, musim panas dan musim dingin, rasa manis dan rasa masam, serta gelap dan terang (Tafsirq, 2025). Tafsir ini menekankan bahwa dalam setiap aspek ciptaan Allah terkandung prinsip keteraturan, keseimbangan, dan keharmonisan yang menjadi landasan struktur semesta.

Dalam penelitian ini, berbagai bentuk pasangan dapat diamati secara nyata seperti antara kondisi sinkron dan tidak sinkron pada pergerakan motor. Ketika PWM kedua motor tidak seimbang, robot akan bergerak menyimpang yang menunjukkan bahwa harmoni antar pasangan (nilai PWM) sangat menentukan kestabilan sistem. Kemudian dalam proses deteksi objek bola, sistem dihadapkan pada pasangan kondisi seperti terdeteksi dan tidak terdeteksi, pusat (*center*) dan tepi (*edge*), serta jarak dekat dan jauh. Ketika bola berada di pusat bidang pandang kamera, sistem dapat mengambil keputusan lebih cepat dan akurat. Namun, ketika bola berada di tepi atau tidak terdeteksi maka sistem memerlukan upaya pencarian

kembali. Hal ini menunjukkan bahwa kondisi ekstrem suatu pasangan dapat berdampak langsung terhadap performa sistem. Pasangan antara *error* dan akurasi juga menjadi penentu keberhasilan sistem. *Error* yang semakin kecil menunjukkan semakin besar ketepatan dalam pengambilan keputusan yang menunjukkan bahwa sistem yang seimbang mampu bekerja secara optimal. Waktu respon yang cepat juga menjadi lawan dari keterlambatan yang keduanya harus dikelola agar sistem tetap stabil. Prinsip dualitas ini mengajarkan pentingnya menyelaraskan setiap elemen dalam sistem, baik dalam aspek teknis maupun spiritual, sebagaimana Allah menciptakan segala sesuatu secara berpasang-pasangan.

Integrasi teknologi dalam penelitian ini menunjukkan bahwa sains tidak hanya bermanfaat untuk kepentingan duniawi tetapi juga sebagai bentuk pengakuan atas keagungan Allah yang telah menganugerahkan manusia akal untuk menciptakan inovasi. Penelitian ini merupakan upaya menunaikan amanah sebagai khalifah di muka bumi, sebagaimana Allah memerintahkan manusia untuk memanfaatkan ilmu pengetahuan demi kemaslahatan umat. Penelitian ini menghubungkan nilai-nilai Islam untuk menciptakan keselarasan antara sains, teknologi, dan iman sehingga hasilnya tidak hanya menjadi solusi teknis, tetapi juga untuk mendekatkan diri kepada Allah SWT.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Penelitian ini telah berhasil mengembangkan sistem kontrol otomasi *robot soccer* menggunakan metode *Hough Circle Transform* sebagai deteksi bola dan pengambilan keputusan. Sistem yang dirancang mampu menjalankan fungsi utama yaitu mendeteksi bola secara *real-time*, menghitung jarak antara bola dan kamera, mengatur arah serta kecepatan gerak motor, dan mengaktifkan solenoid untuk menendang bola apabila bola berada pada jarak yang sangat dekat. Hasil pengujian menunjukkan bahwa sistem memiliki performa deteksi bola yang sangat baik dengan rata-rata *error* sebesar 10%. Estimasi jarak menunjukkan rata-rata *error* sebesar 3,22% yang berarti bahwa sistem cukup presisi dalam memperkirakan posisi bola. Waktu respons sistem terhadap deteksi bola menunjukkan performa yang cukup baik dengan rata-rata *error* sebesar 1,33 detik, menandakan kemampuan sistem dalam merespons pergerakan objek secara *real-time*. Sistem juga berhasil mengaktifkan solenoid berdasarkan estimasi jarak pada saat bola berada dalam jarak ≤ 15 cm. Secara keseluruhan, hasil ini membuktikan bahwa integrasi antara algoritma pengolahan citra, kontrol logika, dan aktuasi fisik dalam sistem berjalan dengan sangat baik.

5.2 Saran

Berdasarkan hasil penelitian, terdapat beberapa saran yang dapat dipertimbangkan untuk pengembangan lebih lanjut:

1. Peningkatan Ketahanan Sistem Terhadap Variasi Lingkungan

Sistem yang dikembangkan dalam penelitian ini diuji dalam lingkungan terkendali dengan pencahayaan yang stabil. Oleh karena itu, untuk meningkatkan tingkat generalisasi dan adaptabilitas sistem dalam skenario dunia nyata, disarankan agar sistem diuji pada lingkungan yang lebih kompleks termasuk perubahan pencahayaan, keberadaan bayangan, serta latar belakang yang bervariasi.

2. Estimasi Jarak yang Lebih Presisi

Penambahan sensor jarak eksternal seperti ultrasonic, infrared, atau LiDAR dalam sistem yang kemudian diintegrasikan dengan data visual melalui pendekatan sensor fusion. Hal ini dapat memperkuat ketepatan estimasi dan memperluas daya jelajah robot dalam berbagai medan.

3. Integrasi Metode Deteksi Tambahan

Meskipun metode *Hough Circle Transform* terbukti efektif dalam mendeteksi objek berbentuk lingkaran, performa sistem masih dapat ditingkatkan melalui integrasi metode deteksi citra tambahan. Salah satu pendekatan yang direkomendasikan adalah kombinasi dengan metode *color filtering* berbasis ruang warna HSV, yang memungkinkan sistem mendeteksi bola tidak hanya berdasarkan bentuk namun juga warna spesifik. Di samping itu, penggunaan pendekatan berbasis *machine learning* seperti *Convolutional Neural Network (CNN)* dapat memperkuat kapabilitas

sistem dalam menghadapi objek dengan kemiripan bentuk atau kondisi citra dalam *noise* tinggi.

DAFTAR PUSTAKA

- Arif, Y. M. (2013). Sistem Kontrol Hexapod robot MSR-H01 Menggunakan Mikrokontroler ATmega 128. *MATICS*. <https://doi.org/10.18860/mat.v0i0.2422>
- Arifin, S. (2022). Sistem Pendeteksi Bola Menggunakan Open CV pada Mobile Robot. *Jurnal Inovasi Teknologi dan Edukasi Teknik*, 2(3), 131–137. <https://doi.org/10.17977/um068v2i32022p131-137>
- Ariyani, L., & La Djamudi, N. (2023). Meningkatkan Keterampilan Membaca melalui Metode Suku Kata pada Mata Pelajaran Bahasa Indonesia Siswa Kelas III Sekolah Dasar. *Prosa: Jurnal Penelitian Pendidikan Guru Sekolah Dasar*, 1(1), 139–146.
- Asri, S. D., Ramayanti, D., Putra, A. D., & Utami, Y. T. (2022). DETEKSI RODA KENDARAAN DENGAN CIRCLE HOUGH TRANSFORM (CHT) DAN SUPPORT VECTOR MACHINE (SVM). *Jurnal Teknoinfo*, 16(2), 427. <https://doi.org/10.33365/jti.v16i2.1952>
- Bahri, Z. (2020). Metode Pusat dan Circular Hough Transformation untuk Mendeteksi Lingkaran pada Sebuah Citra. *Digital Zone: Jurnal Teknologi Informasi dan Komunikasi*, 11(2), 301–310. <https://doi.org/10.31849/digitalzone.v11i2.5086>
- Hariyadi, M. A., Fadila, J. N., & Sifaulloh, H. (2023). 433Mhz based Robot Using Proportional Integral Derivative (PID) for Precise Facing Direction. *International Federation of Robotics*. (2024, Februari 15). <https://ifr.org/ifr-press-releases/news/top-5-robot-trends-2024>
- Isrofi, A., Utama, S. N., & Putra, O. V. (2021). RANCANG BANGUN ROBOT PEMOTONG RUMPUT OTOMATIS MENGGUNAKAN WIRELESS KONTROLER MODUL ESP32-CAM BERBASIS INTERNET of THINGS (IoT). *Jurnal Teknoinfo*, 15(1), 45. <https://doi.org/10.33365/jti.v15i1.675>
- Kemenag. (2022a). *Tafsir As-Sajdah ayat 5*. <https://quran.kemenag.go.id/quran/per-ayat/surah/32?from=5&to=30>
- Kemenag. (2022b). *Tafsir At-Tagabun 11*. <https://quran.kemenag.go.id/quran/per-ayat/surah/64?from=11&to=18>
- Kemenag. (2022c). *Tafsir Yasin ayat 40*. <https://quran.kemenag.go.id/quran/per-ayat/surah/36?from=40&to=83>
- Kontes Robot Indonesia. (2024). <https://kontesrobotindonesia.id/kri-2024.html>

- Maricar, M. A. (2019). Analisa perbandingan nilai akurasi moving average dan exponential smoothing untuk sistem peramalan pendapatan pada perusahaan xyz. *Jurnal Sistem dan Informatika (JSI)*, 13(2), 36–45.
- Mckenna, P. E., Ahmad, M. I., Maisva, T., Nasset, B., Lohan, K., & Hastie, H. (2024). A Meta-Analysis of Vulnerability and Trust in Human–Robot Interaction. *ACM Transactions on Human-Robot Interaction*, 13(3), 1–25. <https://doi.org/10.1145/3658897>
- Perangin-angin, R., & Harianja, E. J. G. (2020). Comparison Detection Edge Lines Algoritma Canny dan Sobel. *Jurnal TIMES*, 8(2), 35–42. <https://doi.org/10.51351/jtm.8.2.2019616>
- Pradana, A. W., & Irmawati, D. (2020). Pendeteksi Bola Pada Robot Penjaga Gawang Menggunakan Metode Hough Circle. *Elinvo (Electronics, Informatics, and Vocational Education)*, 5(1), 21–31.
- Putra, R. M., & Puriyanto, R. D. (2022). Sistem Deteksi dan Pelacakan Bola dengan Metode Hough circle Transform Menggunakan Kamera Omnidirectional pada Robot Sepak Bola Beroda. *Buletin Ilmiah Sarjana Teknik Elektro*, 3(3), 176–184. <https://doi.org/10.12928/biste.v3i3.4786>
- Rahman, E. S., Mapeasse, M. Y., & Hasrul, H. (2023). Studi Pengujian Keserempakan Pemutus Tenaga (PMT) 150 Kv Menggunakan Breaker Analyzer Di Gardu Induk. *Jurnal Media Elektrik*, 20(2), 119–127.
- Rahman, S., Ramli, M., Arnia, F., Muharar, R., Luthfi, M., & Sundari, S. (2020). Analysis and Comparison of Hough Transform Algorithms and Feature Detection to Find Available Parking Spaces. *Journal of Physics: Conference Series*, 1566(1), 012092. <https://doi.org/10.1088/1742-6596/1566/1/012092>
- Raspberry Pi Documentation. (2024). <https://www.raspberrypi.com/documentation/accessories/camera.html>
- Risfendra, R., Akbar, A. A., & Firdaus, F. (2020). Sistem Pergerakan dan Deteksi Pada Robot Sepak Bola Beroda Berbasis Image Processing dengan Penerapan Multivision. *INVOTEK: Jurnal Inovasi Vokasional dan Teknologi*, 20(3), 31–42. <https://doi.org/10.24036/invotek.v20i3.830>
- Rizal, M. F., Sarno, R., & Sabilla, S. I. (2020). Canny Edge and Hough Circle Transformation for Detecting Computer Answer Sheets. *2020 International Seminar on Application for Technology of Information and Communication (iSemantic)*, 346–352. <https://doi.org/10.1109/iSemantic50169.2020.9234208>
- Satriyo, M. B., Anam, K., & Negara, M. A. P. (2021). Sistem Kontrol Robot Sepak Bola Beroda menggunakan Finite State Machine (FSM). *ELKOMIKA: Jurnal Teknik Energi Elektrik, Teknik Telekomunikasi, & Teknik Elektronika*, 9(2), 344. <https://doi.org/10.26760/elkomika.v9i2.344>

- Tafsirq. (2025). *Tafsir Az-Zariyat* 49. <https://tafsirq.com/51-az-zariyat/ayat-49#tafsir-jalalayn>
- Wibowo, B. C., Nugraha, F., & Utomo, A. P. (2021). Uji Deteksi Objek Bentuk Bola Dengan Menerapkan Metode Circular Hough Transform. *Jurnal Informatika Upgris*, 7(1). <https://doi.org/10.26877/jiu.v7i1.8309>
- Winarno, Agoes, A. S., Agustin, E. I., & Arifianto, D. (2020). *Ball detection for KRSBI soccer robot using PeleeNet on omnidirectional camera*. 040012. <https://doi.org/10.1063/5.0036172>
- Yun, C., Ahn, J., & Kim, Y.-H. (2013). An implementation of computer vision technique for an edutainment robot with a visual programming language. *2013 10th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI)*, 131–133. <https://doi.org/10.1109/URAI.2013.6677495>

LAMPIRAN

Lampiran 1

Dokumentasi Kegiatan



