

**ANALISIS SISTEM KENDALI KURSI RODA BERBASIS SINYAL  
ELECTROOCULOGRAPHY DENGAN METODE RANDOM FOREST**

**TUGAS AKHIR**

Karya ilmiah sebagai salah satu syarat untuk menyelesaikan jenjang strata satu  
(S-1) di Jurusan Teknik Elektro, Fakultas Teknik, Universitas Andalas

Oleh

**Gilang Ramadhan**

**2110951005**

Dosen Pembimbing 1

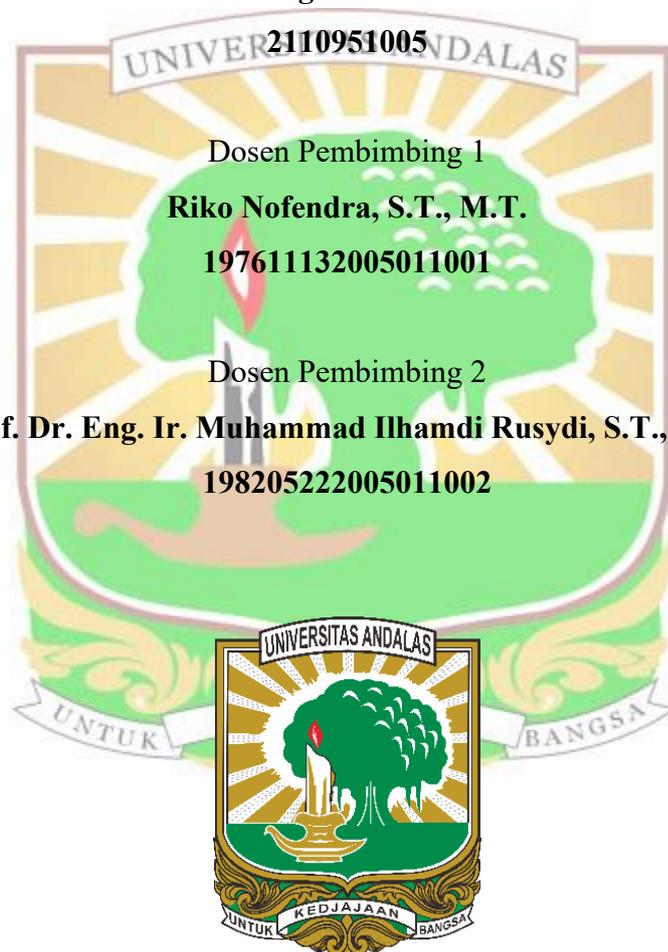
**Riko Nofendra, S.T., M.T.**

**197611132005011001**

Dosen Pembimbing 2

**Prof. Dr. Eng. Ir. Muhammad Ilhamdi Rusydi, S.T., M.T.**

**198205222005011002**



**Program Studi Sarjana Teknik Elektro  
Fakultas Teknik  
Universitas Andalas  
2025**

# LEMBAR PENGESAHAN

## ANALISIS SISTEM KENDALI KURSI RODA BERBASIS SINYAL ELECTROOCULOGRAPHY DENGAN METODE RANDOM FOREST

Oleh  
**Gilang Ramadhan**  
2110951005

Departemen Teknik Elektro, Fakultas Teknik  
Universitas Andalas  
Disetujui Pada Tanggal : 19 Agustus 2025

Pembimbing (Utama),



**Riko Nofendra, S.T., M.T**  
NIP.197611132005011001

Pembimbing Pendamping 1,

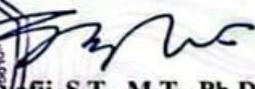


**Prof. Dr.-Eng. Ir. Muhammad Ilhandi Rusydi, S.T., M.T**  
NIP. 198205222005011002

Mengetahui

Ketua Departemen Teknik Elektro



  
**Prof. Syafii, S.T., M.T., Ph.D.**  
NIP. 197405051998021001

## HALAMAN PENGHARGAAN

Alhamdulillah segala puji serta syukur kepada Allah SWT yang telah melimpahkan rahmat, nikmat, kesehatan, kemudahan, dan karunia-Nya sehingga penulis dapat menyelesaikan Tugas Akhir ini dengan rasa bangga dan syukur yang sangat besar. Dalam penulisan Tugas Akhir ini, penulis telah banyak menerima bantuan dari berbagai pihak yang telah memberikan kontribusi seperti saran, kritikan, motivasi, dan dukungan baik secara moral maupun material yang telah penulis terima. Maka, pada kesempatan ini, penulis ingin mengucapkan ucapan terima kasih yang sedalam-dalamnya kepada:

1. Allah SWT dengan segala rahmat dan karunia dari-Nya yang telah memberikan kesempatan, kekuatan dan kemudahan bagi penulis dalam menyelesaikan tugas akhir ini.
2. Ibu dan ayah tercinta, Rita Merli, Amd.Keb. dan Almarhum Yulhanri Lubis, engkau adalah pilar kekuatan dan kelembutan dalam hidup. Terima kasih, ibu dan ayah, atas cinta dan dukungan tanpa batas yang telah kau berikan sehingga penulis mampu menyelesaikan studinya.
3. Teruntuk kedua abang penulis, Andri Chardo Tanusa, A.Md.T. dan Agung A. Maulana, S.I.Kom. Penulis berterima kasih atas dukungan yang kalian berikan selama ini.
4. Kepada Bapak Prof. Dr.Eng. Ir. Muhammad Ilhamdi Rusydi, S.T., dan Bapak Riko Nofendra, M.T. Sebagai dosen pembimbing selama penulisan tugas akhir yang selalu tidak hentinya memberikan saran, kritikan, bimbingan dan membantu penulis dalam mengembangkan pemikiran akademis yang akan terus membersamai penulis meskipun sudah tidak berada pada lingkungan kampus lagi.
5. Kepada Bapak Dr. Ir. Darmawan, M.Sc. dan Bapak Ir. Hanalde Andre, S.T., M.T. Sebagai dosen penguji saat melaksanakan tugas akhir yang terus senantiasa memberikan saran dan masukan dalam menyelesaikan tugas akhir ini.
6. Rekan – rekan satu bimbingan tugas akhir dan teman – teman seperjuangan yang banyak berkontribusi dalam menyelesaikan tugas akhir. Walizaqhi Ridho Allbis, Kevin Imam Satria, Aqmal Fajar Putra, Kevin Imam Satria, Anugrah Illahi, Fhahrezy Thomas, dan Nabila Shifa Deyo. Terimakasih telah menjadi teman berbagi pikiran dan pendapat, membantu dan membersamai penulis dari awal penulisan tugas akhir hingga selesai. Kebersamaan dalam proses ini membuat penulis menjadi tidak patah semangat dalam menyelesaikan tugas akhir ini.

## LEMBAR PERNYATAAN KEASLIAN

Saya yang bertanda tangan dibawah ini :

Nama : Gilang Ramadhan  
NIM : 2110951005  
Program Studi : Teknik Elektro  
Perguruan Tinggi : Universitas Andalas

Dengan ini menyatakan bahwa tugas akhir yang berjudul “ **Analisis Sistem Kendali Kursi Roda Berbasis Sinyal Electrooculography Dengan Metode Random Forest**” adalah asli (orisinil) atau tidak plagiat (menjiplak) dan belum pernah diterbitkan/dipublikasikan dimanapun dan dalam bentuk apapun.

Demikianlah surat pernyataan ini saya buat dengan sebenar-benarnya tanpa ada paksaan dari pihak manapun juga. Apabila dikemudian hari ternyata saya memberikan keterangan palsu dan atau ada pihak lain yang mengklaim bahwa tugas akhir yang telah saya buat adalah hasil karya milik seseorang atau badan tertentu. Saya bersedia diproses baik secara pidana maupun perdata dan kelulusan saya dari Universitas Andalas dicabut/dibatalkan.

Padang, 19 Agustus 2025

Penulis



Gilang Ramadhan

NIM. 2110951005

## RIWAYAT HIDUP



**Gilang Ramadhan**, lahir di Paraman Ampalu, Kabupaten Pasaman Barat, pada tanggal 26 November 2002. Merupakan anak ketiga dari tiga bersaudara, dilahirkan dari pasangan Bapak Yulhanri Lubis, A.Ma. dan Ibu Rita Merli, A.Md. Keb. Penulis menyelesaikan pendidikan sekolah dasar di SDN 06 Gunung Tuleh dan SDN 09 Pasaman, kemudian melanjutkan ke jenjang menengah pertama di SMPN 1 Pasaman, dan jenjang menengah atas di SMAN 1 Pasaman. Setelah lulus pada tahun 2021, penulis melanjutkan pendidikan sebagai mahasiswa Program Studi S1 Teknik Elektro, Fakultas Teknik, Universitas Andalas. Selama masa perkuliahan, penulis melaksanakan program magang di PT Semen Padang, tepatnya di Indarung V pada unit pemeliharaan listrik dan instrumentasi selama 1 bulan. Di sana, penulis menyusun laporan akhir dengan judul “Analisis Penyebab Perbedaan Nilai Pressure Damper 5J1T10M1 pada Electrostatic Precipitator Kiln Indarung V PT Semen Padang dengan Metode Fishbone Diagram”. Penulis juga mengikuti Program Kreativitas Mahasiswa (PKM) hingga tahap pendanaan DIKTI pada tahun 2023 dengan penelitian berjudul “Pemodelan Matematika Trayektori Drone Untuk Pemetaan Cepat Area Terdampak Abu Erupsi Gunung Marapi sebagai Upaya Penanggulangan bencana”. Ditahun 2024, Penulis mengikuti mata kuliah proyek desain Teknik elektro dengan membuat alat inovasi skiring dini penyakit retinopati hipertensi berbasis deep learning yaitu dengan metode convolutional neural network (CNN) dan terintegrasi Internet of Things. Penulis juga mengikuti program MBKM Studi Independen di Indobot Academy, dengan fokus pada pembelajaran Internet of Things (IoT) pada tahun 2024. Pada akhir program, penulis mengerjakan proyek akhir berupa alat monitoring suhu dan kelembaban untuk kandang maggot Black Soldier Fly (BSF) yang diimplementasikan di Lampung. Penulis juga memperoleh juara 1 dalam lomba capstone project tingkat nasional yang diadakan di Universitas Sriwijaya dengan judul alat “ HYPERTOPHY : Alat Skrining Retinopati Hipertensi dengan Menggunakan Convolutional Neural Network dan terintegrasi Internet of Things. Penulis juga merupakan anggota dari Himpunan Mahasiswa Teknik Elektro (HMTE) Universitas Andalas. Dalam bidang organisasi dan kepanitiaan, penulis pernah menjadi anggota pengurus BEM KM FT Unand selama dua periode 2022 dan 2023 serta anggota pengurus HMTE FT Unand periode 2023.

Judul	<b>ANALISIS SISTEM KENDALI KURSI RODA BERBASIS SINYAL ELECTROOCULOGRAPHY DENGAN METODE RANDOM FOREST</b>	Gilang Ramadhan
Program Studi	Teknik Elektro	2110951005
Fakultas Teknik Universitas Andalas		
<b>ABSTRAK</b>		
<p>Penyandang disabilitas adalah setiap orang yang mengalami keterbatasan fisik, intelektual, mental, atau sensorik dalam jangka waktu lama. Berdasarkan survei BAPPENAS tahun 2021, jenis penyandang disabilitas berjalan sebagai salah satu jumlah penyandang terbanyak di Indonesia. Pentingnya alat bantu berjalan bagi penyandang disabilitas, membuat banyak peneliti mengembangkan alat bantu berupa kursi roda pintar. Salah satu pengembangannya, yaitu dengan memanfaatkan EOG (<i>electrooculography</i>) sebagai pengendali kursi roda, karena tidak terkendala dengan cacat tubuh. Tantangan utama dalam sistem kendali kursi roda berbasis EOG adalah membedakan sinyal yang memiliki polaritas serupa, seperti lirik atas, kedip sadar, dan kedip tidak sadar, serta mengabaikan sinyal non-perintah akibat gerakan refleks atau sinyal balik. Penelitian ini merancang sistem pengendalian kursi roda berbasis EOG dengan memanfaatkan fitur puncak sinyal dan metode klasifikasi Random Forest, serta menerapkan mekanisme locking time untuk meningkatkan akurasi navigasi. Hasil klasifikasi tiga jenis sinyal EOG dengan polaritas yang sama menggunakan Random Forest menghasilkan akurasi sebesar 98,9%. Pengujian performa pergerakan kursi roda mencapai akurasi 98,8%, dan pengujian terhadap lintasan menunjukkan waktu tempuh rata-rata satu putaran selama 79,78 detik, menandakan respons sistem yang baik terhadap perintah navigasi. Selain itu, pengujian locking time menunjukkan bahwa durasi 1,5 detik hingga 3 detik secara efektif mencegah deteksi sinyal tidak valid tanpa mengurangi responsivitas sistem. Dengan demikian, sistem yang dikembangkan mampu meningkatkan ketepatan klasifikasi sinyal serta keandalan navigasi kursi roda.</p> <p><b>Kata Kunci:</b> akurasi, <i>electrooculography</i>, kursi roda, <i>locking time</i>, <i>random forest</i>,</p>		

<i>Title</i>	<i>ANALYSIS OF A WHEELCHAIR CONTROL SYSTEM BASED ON ELECTROOCULOGR PHY SIGNALS USING THE RANDOM FOREST METHOD</i>	Gilang Ramadhan
Mayor	<i>Electrical Engineering</i>	2110951005

*Engineering Faculty Andalas University*

**ABSTRACT**

*Persons with disabilities are individuals who experience long-term physical, intellectual, mental, or sensory limitations. According to a 2021 survey by BAPPENAS, mobility impairments rank among the most common types of disabilities in Indonesia. The importance of mobility aids for persons with disabilities has driven many researchers to develop assistive technologies such as smart wheelchairs. One such development utilizes Electrooculography (EOG) as a wheelchair control system, as it is not hindered by physical impairments. The main challenge in EOG-based wheelchair control systems lies in distinguishing signals with similar polarity—such as upward gaze, voluntary blink, and involuntary blink—while ignoring non-command signals caused by reflex movements or signal rebounds. This study designs a wheelchair control system based on EOG by leveraging signal peak features and the Random Forest classification method, and applies a locking time mechanism to improve navigation accuracy. Classification of three EOG signal types with similar polarity using Random Forest achieved an accuracy of 98.9%. Performance testing of wheelchair movement reached 98.8% accuracy, and path navigation tests showed an average lap time of 79.78 seconds, indicating good system responsiveness to navigation commands. Furthermore, locking time testing demonstrated that durations between 1.5 to 3 seconds effectively prevented the detection of invalid signals without reducing system responsiveness. Thus, the developed system is capable of improving both signal classification accuracy and the reliability of wheelchair navigation.*

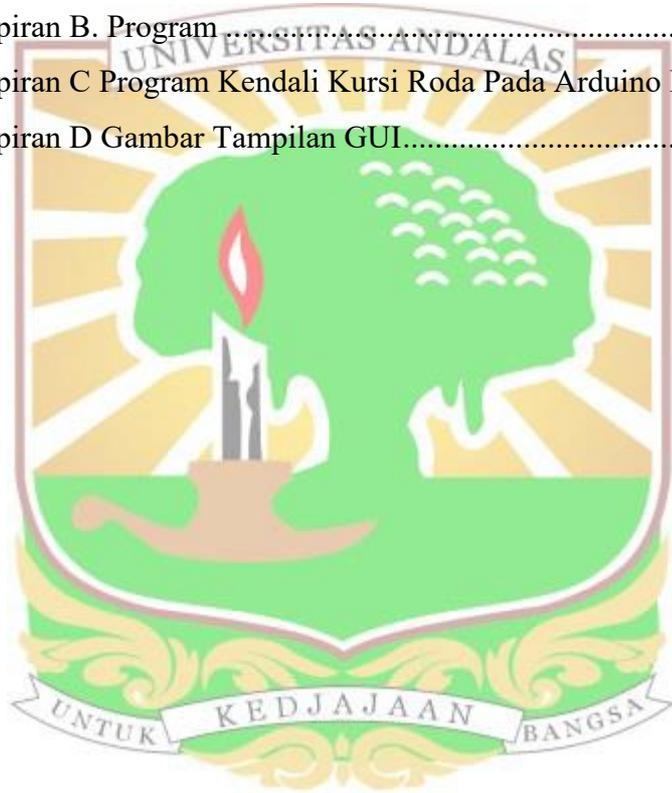
**Keywords:** *accuracy, electrooculography, wheelchair, locking time, random forest.*

## DAFTAR ISI

LEMBAR PENGESAHAN .....	ii
HALAMAN PENGHARGAAN .....	iii
LEMBAR PERNYATAAN KEASLIAN .....	iv
RIWAYAT HIDUP .....	v
ABSTRAK .....	vi
<i>ABSTRACT</i> .....	vii
DAFTAR ISI .....	viii
PRAKATA .....	xi
DAFTAR GAMBAR .....	xii
DAFTAR TABEL .....	xiii
DAFTAR SIMBOL .....	xiv
DAFTAR LAMPIRAN .....	xv
DAFTAR ISTILAH .....	xvi
DAFTAR SINGKATAN .....	xvii
BAB I PENDAHULUAN .....	1
1.1 Latar Belakang .....	1
1.2 Rumusan Masalah .....	3
1.3 Tujuan Penelitian .....	3
1.4 Batasan Masalah .....	3
1.5 Manfaat Penelitian .....	4
1.6 Sistematika Penulisan .....	4
BAB II TINJAUAN PUSTAKA .....	6
2.1 Kursi Roda .....	6
2.1.1 Kursi Roda Konvensional .....	6
2.1.2 Kursi Roda Elektrik .....	6
2.2 <i>Electrooculography</i> (EOG) .....	7
2.2.1 Elektroda .....	7
2.2.2 Rangkaian EOG .....	8
2.2.3 Bentuk Sinyal EOG .....	10
2.2.4 Fitur Sinyal EOG .....	12

2.3	Rangkaian Kursi Roda .....	14
2.4	NI USB 6008 .....	14
2.5	Random Forest Classifier .....	15
2.6	Confusion Matrix.....	17
2.7	Visual Studio Code .....	18
BAB III METODOLOGI PENELITIAN.....		19
3.1	Jenis dan Tahapan Penelitian .....	19
3.2	Perancangan Sistem .....	20
3.2.1	Perancangan Perangkat Keras .....	23
3.2.2	Perancangan Perangkat Lunak.....	24
3.3	Rancangan Pengujian .....	26
3.4	Rancangan Operasi Kursi Roda .....	28
3.4	Jadwal Penelitian.....	29
3.5	Analisis Sistem.....	30
BAB IV HASIL DAN PEMBAHASAN .....		31
4.1	Penetapan Nilai Threshold.....	31
4.2	Pengambilan Data Sinyal Pergerakan Mata.....	32
4.2.1	Data Sinyal Lirik Kanan .....	33
4.2.2	Data Sinyal Lirik Kiri.....	34
4.2.3	Data Sinyal Lirik Bawah .....	35
4.2.4	Data Sinyal Lirik Atas .....	36
4.2.5	Data Sinyal Kedip Sadar .....	37
4.2.6	Data Sinyal Kedip Tidak Sadar .....	37
4.2.7	Analisis Fitur Sinyal Pergerakan Mata.....	38
4.3	Hasil Perancangan Model <i>Random Forest</i> .....	39
4.3.1	Pengujian Akurasi Model.....	41
4.3.2	Pengujian Waktu Komputasi Model <i>Random Forest</i> .....	44
4.3.3	Pemilihan Parameter Model <i>Random Forest</i> .....	45
4.4	Pengujian Durasi Locking Time .....	46
4.5	Implementasi Kursi Roda .....	47
4.6	Pengujian Kursi Roda .....	50
4.6.1	Pengujian Ketepatan Klasifikasi <i>Random Forest</i> .....	50

4.6.2 Pengujian Ketepatan Gerakan Kursi Roda .....	52
4.6.3 Pengujian Kursi Roda menggunakan Lintasan .....	53
4.7 Perbandingan dengan Penelitian Sebelumnya .....	55
BAB V KESIMPULAN DAN SARAN .....	59
5.1 Kesimpulan .....	59
5.2 Saran .....	60
DAFTAR PUSTAKA .....	61
LAMPIRAN .....	66
Lampiran A. Data Uji dan Data Latih.....	66
Lampiran B. Program .....	67
Lampiran C Program Kendali Kursi Roda Pada Arduino IDE.....	76
Lampiran D Gambar Tampilan GUI.....	79



## PRAKATA

Puji syukur kepada Allah SWT yang telah memberikan rahmat dan karunia-Nya sehingga penulis dapat menyelesaikan laporan tugas akhir dengan judul “Analisis Sistem Kendali Kursi Roda Berbasis Sinyal Electrooculography Dengan Metode Random Forest”. Tugas akhir ini disusun sebagai salah satu syarat untuk menyelesaikan jenjang pendidikan Strata 1 (S1) di Departemen Teknik Elektro, Fakultas Teknik, Universitas Andalas

Dalam proses penyusunan tugas akhir ini, penulis memperoleh banyak bantuan serta dukungan dari berbagai pihak. Oleh sebab itu, dengan penuh rasa hormat penulis menyampaikan ucapan terima kasih kepada:

1. Bapak Prof. Syafii, S.T., M.T., Ph.D. selaku Ketua Departemen Teknik Elektro, Fakultas Teknik, Universitas Andalas.
2. Bapak Primas Emeraldi, M.T. selaku Kepala Program Studi S1 Departemen Teknik Elektro, Fakultas Teknik, Universitas Andalas.
3. Bapak Prof. Dr. Eng. Ir. Muhammad Ilhamdi Rusydi, S.T., M.T. dan Bapak Riko Nofendra, S.T., M. sebagai dosen pembimbing selama penulisan tugas akhir yang selalu memberikan saran, bimbingan, kritik dan bantuan dalam penyusunan tugas akhir ini.
4. Bapak Ir. Darmawan, S.T., M.Sc. dan Bapak Ir. Hanalde Andre, S.T., M.T. selaku dosen penguji yang senantiasa memberikan saran dan masukan selama penyelesaian tugas akhir ini.
5. Seluruh pihak yang telah membantu penulis dalam menyelesaikan tugas akhir ini, yang tidak dapat disebutkan satu persatu.

Penulis menyadari bahwa dalam penulisan tugas akhir ini masih memiliki kekurangan baik dari cara penyajian maupun penulisan. Oleh karena itu, penulis mengharapkan saran dan masukan dari para pembaca demi penyempurnaan tugas akhir ini melalui [2110951005\\_gilang@student.unand.ac.id](mailto:2110951005_gilang@student.unand.ac.id). Semoga tugas akhir ini bisa berguna bagi pembaca dan bagi penulis sendiri.

Padang, 19 Agustus 2025



Gilang Ramadhan

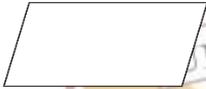
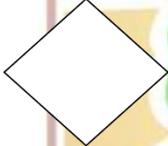
## DAFTAR GAMBAR

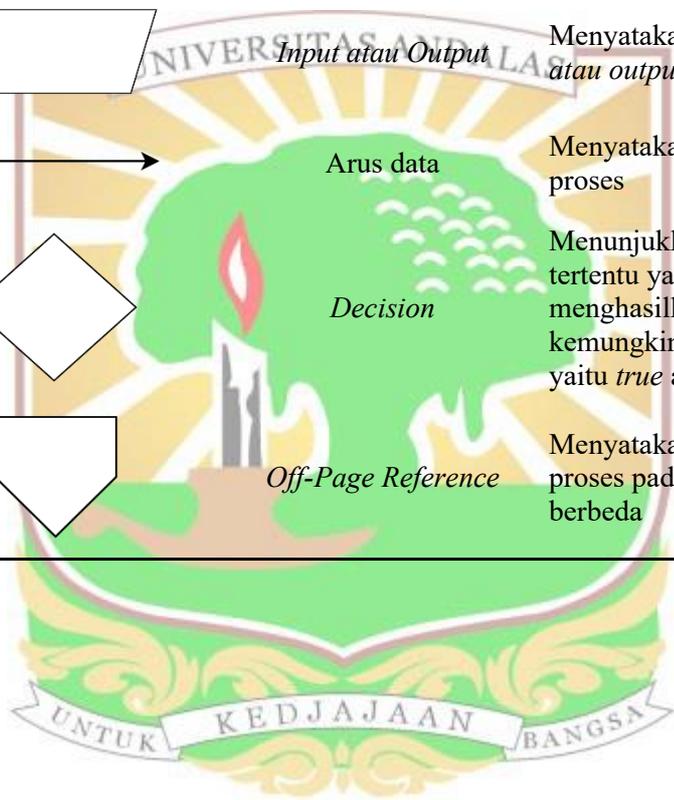
Gambar 2. 1 Kursi Roda Konvensional .....	6
Gambar 2. 2 Kursi Roda Elektrik.....	7
Gambar 2. 3 Polaritas kornea dan retina mata[26].....	7
Gambar 2. 4 Elektroda <i>Surface</i> [34] .....	8
Gambar 2. 5 Posisi Elektroda.....	8
Gambar 2. 6 Rangkaian EOG .....	9
Gambar 2. 7 Sinyal Lirikan Mata ke Kanan .....	10
Gambar 2. 8 Sinyal Lirikan Mata ke Kiri .....	10
Gambar 2. 9 Sinyal Lirikan Mata ke Atas .....	11
Gambar 2. 10 Sinyal Lirikan Mata ke Bawah.....	11
Gambar 2. 11 Sinyal Berkedip Secara Sadar .....	12
Gambar 2. 12 Sinyal Berkedip Secara Tidak Sadar.....	12
Gambar 2. 13 Grafik Fitur Sinyal[17].....	14
Gambar 2. 14 Rangkaian Kursi Roda .....	14
Gambar 2. 15 NI USB-6008 [46].....	15
Gambar 2. 16 Strukur Skema kerja <i>Random Forest</i> .....	17
Gambar 2. 17 Tampilan Interface Visual Studio Code.....	17
Gambar 3. 1 Diagram Alir Tahapan Penelitian .....	19
Gambar 3. 2 Diagram Alir Tahapan Perancangan Sistem .....	21
Gambar 3. 3 Rancangan Perangkat Keras.....	23
Gambar 3. 4 Diagram Alir Sistem Perangkat Lunak .....	24
Gambar 3. 5 Diagram Alir Proses <i>Random Forest</i> .....	25
Gambar 3. 6 Diagram Alir Proses Kerja <i>Locking Time</i> .....	26
Gambar 3. 7 Pergerakan Kursi Roda.....	29
Gambar 4. 1 Grafik Sinyal Diam .....	32
Gambar 4. 2 Posisi Peletakan Elektroda Pengambilan Data Sinyal EOG .....	33
Gambar 4. 3 Sinyal Lirik Kanan.....	34
Gambar 4. 4 Sinyal Lirik Kiri .....	35
Gambar 4. 5 Sinyal Lirik Bawah .....	36
Gambar 4. 6 Sinyal Lirik Atas .....	36
Gambar 4. 7 Sinyal Lirik Sadar .....	37
Gambar 4. 8 Sinyal Lirik Tidak Sadar .....	38
Gambar 4. 9 Pengimplementasian Perangkat Keras Kursi Roda.....	48
Gambar 4. 10 Tampilan GUI .....	49
Gambar 4. 11 Lintasan Pengujian Kursi Roda.....	54

## DAFTAR TABEL

Tabel 3. 1 Confusion Matrix .....	27
Tabel 3. 2 Jadwal Penelitian.....	29
Tabel 4. 1 Analisis Nilai <i>Threshold</i> .....	32
Tabel 4. 2 Fitur Polaritas Berdasarkan Jenis Sinyal Pergerakan Mata .....	38
Tabel 4. 3 Fitur Puncak Sinyal Berdasarkan Jenis Sinyal Pergerakan Mata .....	39
Tabel 4. 4 Data Kelas .....	40
Tabel 4. 5 Pembagian Data Latih dan Data Uji .....	40
Tabel 4. 6 Parameter Pengujian <i>Random Forest</i> .....	40
Tabel 4. 7 Confusion Matrix .....	41
Tabel 4. 8 Jumlah Pohon 25 dengan Kedalaman Maksimal 3 .....	42
Tabel 4. 9 Jumlah Pohon 25 dengan Kedalaman Maksimal 5 .....	42
Tabel 4. 10 Jumlah Pohon 25 dengan Kedalaman Maksimal 7 .....	42
Tabel 4. 11 Jumlah Pohon 25 dengan Kedalaman Maksimal 10 .....	42
Tabel 4. 12 Jumlah Pohon 50 dengan Kedalaman Maksimal 3 .....	43
Tabel 4. 13 Jumlah Pohon 50 dengan Kedalaman Maksimal 5 .....	43
Tabel 4. 14 Jumlah Pohon 50 dengan Kedalaman Maksimal 7 .....	43
Tabel 4. 15 Jumlah Pohon 50 dengan Kedalaman Maksimal 10 .....	43
Tabel 4. 16 Jumlah Pohon 100 dengan Kedalaman Maksimal 3 .....	44
Tabel 4. 17 Jumlah Pohon 100 dengan Kedalaman Maksimal 5 .....	44
Tabel 4. 18 Jumlah Pohon 100 dengan Kedalaman Maksimal 7 .....	44
Tabel 4. 19 Jumlah Pohon 100 dengan Kedalaman Maksimal 10 .....	44
Tabel 4. 20 Waktu Komputasi <i>Random Forest</i> .....	45
Tabel 4. 21 Akurasi dan Waktu Komputasi Model <i>Random Forest</i> .....	46
Tabel 4. 22 Durasi <i>Locking Time</i> .....	47
Tabel 4. 23 Hubungan Sinyal dan Output Kursi Roda.....	48
Tabel 4. 24 Pengujian Klasifikasi Metode <i>Random Forest</i> .....	50
Tabel 4. 25 Hasil Pengujian Ketepatan Gerakan Kursi Roda .....	52
Tabel 4. 26 Hasil Pengujian Performa Kursi Roda terhadap Lintasan .....	54
Tabel 4. 27 Perbandingan terhadap penelitian sebelumnya .....	56

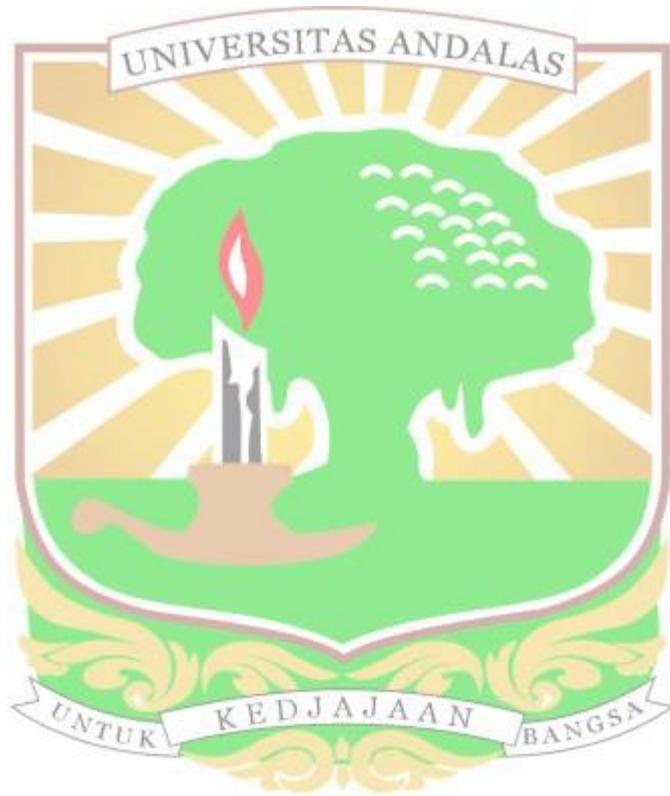
## DAFTAR SIMBOL

No	Simbol	Nama	Keterangan
1		Terminal	Menyatakan awal atau akhir suatu program
2		<i>Preparation</i>	Menyatakan penyediaan tempat penyimpanan suatu pengolahan untuk memberikan nilai awal
3		Proses	Menyatakan Proses yang dilakukan
4		<i>Input atau Output</i>	Menyatakan proses <i>input</i> atau <i>output</i>
5		Arus data	Menyatakan arus suatu proses
6		<i>Decision</i>	Menunjukkan kondisi tertentu yang akan menghasilkan dua kemungkinan jawaban, yaitu <i>true</i> atau <i>false</i>
7		<i>Off-Page Reference</i>	Menyatakan penghubung proses pada halaman yang berbeda



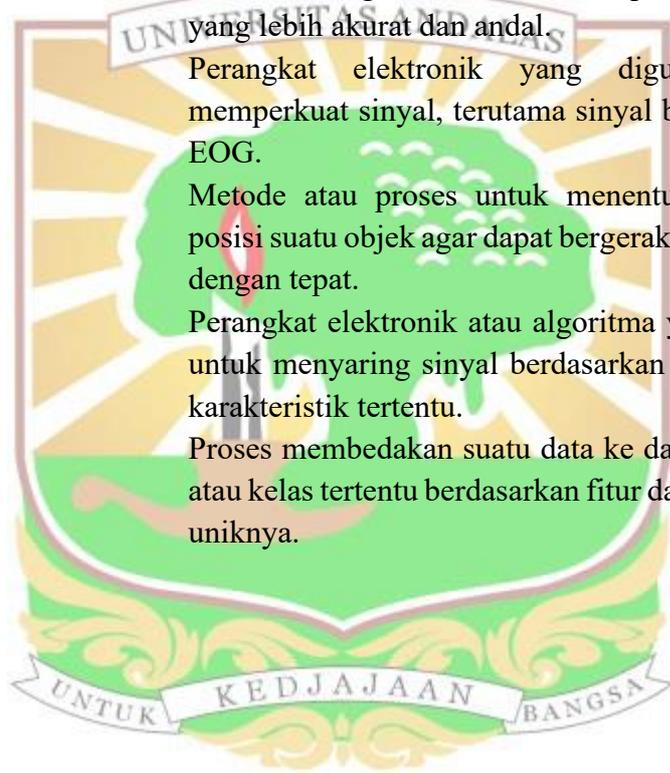
## DAFTAR LAMPIRAN

Lampiran A. Data Uji dan Data Latih.....	66
Lampiran B. Program.....	67
Lampiran C. Program Kendali Kursi Roda Pada Arduino IDE.....	76
Lampiran D. Gambar Tampilan GUI.....	79



## DAFTAR ISTILAH

Disabilitas	Suatu keadaan yang mengakibatkan seseorang memiliki keterbatasan dalam menjalankan aktivitas.
Fitur	Karakteristik dari suatu data yang memiliki arti unik dan dapat digunakan dalam proses analisis atau pembelajaran mesin.
Electrooculography (EOG)	Sinyal listrik yang muncul akibat perbedaan potensial antara kornea dan retina mata, digunakan untuk mendeteksi gerakan mata.
<i>Ensemble Learning</i>	Metode machine learning yang menggabungkan satu atau lebih algoritma untuk memperoleh performa yang lebih akurat dan andal.
Amplifier	Perangkat elektronik yang digunakan untuk memperkuat sinyal, terutama sinyal biologis seperti EOG.
Navigasi	Metode atau proses untuk menentukan arah dan posisi suatu objek agar dapat bergerak menuju tujuan dengan tepat.
Filter	Perangkat elektronik atau algoritma yang berfungsi untuk menyaring sinyal berdasarkan frekuensi atau karakteristik tertentu.
Klasifikasi	Proses membedakan suatu data ke dalam kelompok atau kelas tertentu berdasarkan fitur dan karakteristik uniknya.



## DAFTAR SINGKATAN

EOG	<i>Electrooculography</i>
HMI	<i>Human Machine Interface</i>
DC	<i>Direct Current</i>
TPR	<i>True Positive Rate</i>
FNR	<i>False Negative Rate</i>
Ag/AgCl	<i>Silver/Silver-Chloride</i>



# BAB I PENDAHULUAN

## 1.1 Latar Belakang

Penyandang disabilitas adalah setiap orang yang mengalami keterbatasan fisik, intelektual, mental, atau sensorik dalam jangka waktu lama [1]. Berdasarkan data survei dari BAPPENAS pada tahun 2021, kelompok disabilitas dengan kesulitan berjalan (38,3%) menempati jumlah terbanyak kedua di Indonesia di bawah kelompok disabilitas melihat (63,7%) [2]. Mobilitas penyandang disabilitas dengan gangguan berjalan sangat terbatas, karena mereka memerlukan alat bantu untuk berpindah tempat. Orang yang memiliki keterbatasan pada kemampuan berpindah tempat seperti berjalan pada umumnya menggunakan bantuan kursi roda.

Seiring dengan kemajuan teknologi, pengembangan *Human Machine Interface* (HMI) menjadi solusi inovatif dalam menciptakan alat bantu mobilitas yang lebih canggih bagi penyandang disabilitas. HMI berperan penting sebagai jembatan penghubung antara manusia dan teknologi [3]. Beberapa penelitian sebelumnya telah mengembangkan kursi roda pintar yang dapat dikendalikan dengan menggunakan *joystick* dan kamera sebagai pendeteksi rintangan [4]. Akan tetapi, penggunaan *joystick* pada kursi roda pintar tidak bisa digunakan pada penderita disabilitas yang mempunyai lumpuh pada tangan dan kaki.

Solusi inovatif lain berupa kendali kursi roda pintar berbasis suara sebagai navigasi juga dilakukan oleh Ridia dkk [5]. Namun, pengendalian kursi roda ini memiliki kelemahan saat berada di keramaian. Efektivitas sistem ini di keramaian menurun, karena sistem menjadi rentan terhadap kesalahan perintah akibat kebisingan lingkungan sekitar [6]. Selain kendali berbasis *joystick* dan suara, terdapat pula teknologi pengendalian alat bantu yang menggunakan sinyal tubuh atau lebih dikenal dengan biosignal.

Biosignal merupakan sinyal listrik fisiologis yang dihasilkan oleh tubuh manusia [7]. Beberapa jenis biosignal yang umum dikenali meliputi *Electrooculography* (EOG), *Electromyography* (EMG), *Electrocardiography* (ECG), dan *electroencephalography* (EEG). EOG merekam perbedaan potensial listrik di sekitar mata yang dihasilkan oleh pergerakan bola mata [8]. EMG merekam sinyal yang dihasilkan oleh aktivitas otot rangka, sementara ECG mengukur aktivitas listrik otot jantung [9], [10]. Adapun EEG merekam sinyal elektrik dari aktivitas otak [11].

Dari berbagai jenis biosignal tersebut, EOG menjadi salah satu yang paling banyak digunakan dalam pengembangan alat bantu untuk mobilitas penyandang disabilitas. *Electrooculography* (EOG) adalah teknik yang digunakan untuk mendeteksi sinyal yang dihasilkan dari pergerakan mata berdasarkan perbedaan potensial antara retina dan kornea mata [12]. Penggunaan EOG yang dapat

membaca potensial listrik dari mata membuat EOG sangat cocok untuk digunakan dalam kursi roda pintar karena tidak terpengaruh oleh cacat tubuh.

Penelitian yang mengembangkan EOG sebagai pengendali kursi roda pintar merancang sistem kontrol kursi roda berbasis *Graphical User Interface* (GUI) yang menampilkan *flash button* sebagai indikator perintah dan kedipan mata sebagai pemilihan [13]. Penelitian lain juga merancang sistem klasifikasi sinyal lirikan mata (kiri, kanan, atas, dan bawah) berdasarkan nilai ambang batas untuk mengendalikan kursi roda menggunakan sinyal EOG. Dalam perancangan sistem ini, pengendalian kursi roda menggunakan gerakan mata mendapatkan hasil akurasi rata-rata sebesar 90% [14].

Penggunaan EOG pada kursi roda pintar mengharuskan pengguna untuk terus-menerus melirik agar kursi roda dapat bergerak [15]. Pergerakan mata secara terus-menerus ini dapat menyebabkan kelelahan pada mata pengguna [16]. Pengguna juga tidak dapat melihat secara bebas ke objek lain saat mengoperasikan kursi roda, dikarenakan akan mengganggu sistem operasinya [15]. Sehingga dalam penggunaan kursi roda EOG diperlukan perintah untuk mengaktifkan dan menonaktifkan sistem agar mata pengguna dapat beristirahat.

Penelitian lain telah mengembangkan sistem ON/OFF pada kursi roda [17]. Penelitian ini menggunakan polaritas sinyal untuk membedakan sinyal lirikan mata (kiri, kanan, atas, dan bawah), lalu mengklasifikasikan sinyal kedip mata (sadar dan tidak sadar) berdasarkan puncak sinyalnya, dimana sinyal lirik digunakan untuk mengendalikan arah pergerakan kursi roda dan sinyal kedip sadar digunakan untuk sistem ON/OFF pada kursi roda. Klasifikasi berdasarkan puncak sinyal dilakukan pada sinyal kedip sadar dan kedip tidak sadar karena kedua sinyal ini memiliki polaritas yang sama, namun memiliki puncak sinyal yang berbeda.

Pada penelitian selanjutnya dilakukan pengembangan sistem ON/OFF pada kursi roda EOG dengan mengklasifikasikan sinyal lirikan atas, kedip sadar, dan kedip tidak sadar berdasarkan fitur puncak sinyal menggunakan metode *decision tree* dan mengklasifikasikan lirik kanan, lirik kiri, lirik bawah berdasarkan fitur polaritas [18]. Pada pengujian ketepatan gerakan kursi roda diperoleh hasil akurasi total sebesar 97,89%. Namun, pada penelitian tersebut masih terdapat kesalahan klasifikasi dari metode yang digunakan untuk kendali kursi roda. Pada penelitian ini juga belum mampu mengatasi gangguan pada sistem pergerakan kursi roda yaitu sistem belum dapat mengabaikan atau membedakan sinyal perintah dan sinyal non perintah sebagai akibat gerakan refleks mata sehingga terjadi kesalahan navigasi saat menggunakan kursi roda.

Berdasarkan penelitian-penelitian sebelumnya, masih terdapat kesalahan dalam klasifikasi sinyal EOG dari beberapa metode yang digunakan. Selain itu, munculnya sinyal baru yang dihasilkan oleh sinyal balik dan gerakan refleks mata saat pengguna mengoperasikan kursi roda juga dapat menyebabkan kesalahan dalam navigasi pergerakan kursi roda. Oleh karena itu, diperlukan metode klasifikasi yang tahan terhadap *noise* dan dapat mengurangi risiko terjadinya

kesalahan klasifikasi sinyal agar dapat meningkatkan akurasi kinerja sistem sehingga dapat meningkatkan performa kursi roda. Begitupun diperlukan suatu sistem yang dapat mengabaikan sinyal balik dan sinyal gerakan refleks mata yang dapat mengatasi kesalahan navigasi pada kursi roda. Dengan demikian, penulis berminat melakukan penelitian dengan judul “ANALISIS SISTEM KENDALI KURSI RODA BERBASIS SINYAL ELECTROOCULOGRAPHY DENGAN METODE RANDOM FOREST”.

## 1.2 Rumusan Masalah

Berdasarkan latar belakang permasalahan diatas serta rujukan penelitian sebelumnya, masih terdapat kesalahan yang terjadi dalam membedakan sinyal dengan polaritas yang sama. Selain itu, adanya sinyal balik dan gerakan refleks mata saat menggerakkan kursi roda dapat menyebabkan kesalahan navigasi pada kursi roda. Oleh karena itu, untuk menghindari kesalahan klasifikasi dan sistem dapat membedakan sinyal perintah dengan sinyal non-perintah sebagai akibat adanya sinyal balik dan gerakan refleks mata, maka dirancang rumusan masalah sebagai berikut:

1. Bagaimana merancang sistem operasi kursi roda berdasarkan polaritas sinyal dan penerapan algoritma *random forest* untuk mengklasifikasikan 3 sinyal gerakan mata (lirik atas, kedip sadar, dan kedip tidak sadar ) ?
2. Bagaimana merancang sistem navigasi kursi roda agar dapat mengabaikan sinyal non-perintah seperti sinyal balik dan gerakan refleks mata?

## 1.3 Tujuan Penelitian

Berdasarkan rumusan masalah diatas, maka tujuan dari penelitian tugas akhir ini adalah sebagai berikut:

1. Merancang sistem operasi kursi roda berdasarkan polaritas sinyal dan menerapkan algoritma *random forest* dalam membedakan gerakan lirik atas, kedip sadar, dan kedip tidak sadar.
2. Merancang sistem navigasi kursi roda efektif untuk mencegah kesalahan deteksi sinyal non perintah seperti sinyal balik dan gerakan refleks mata.

## 1.4 Batasan Masalah

Penelitian ini dibatasi pada masalah tertentu agar tidak meluasnya pembahasan penelitian. Adapun batasan masalah dalam penelitian ini adalah sebagai berikut:

1. Penelitian ini berfokus pada perbaikan sistem gerakan untuk kursi roda EOG dengan menggunakan sinyal mata.
2. Data yang digunakan untuk pelatihan dan pengujian klasifikasi menggunakan fitur puncak sinyal.
3. Uji coba sistem dilakukan pada mahasiswa Universitas Andalas.

4. Perangkat yang digunakan selama pengambilan data, seperti laptop, sensor electrooculography (EOG), serta peralatan pendukung lainnya, harus dipastikan tidak terhubung langsung ke sumber listrik PLN untuk menghindari munculnya derau (*noise*) pada sinyal EOG.
5. Saat menggunakan kursi roda, posisi kepala pengguna harus tetap stabil dan tidak bergerak bebas agar sinyal yang diterima sensor EOG tetap akurat.
6. Kursi roda yang dikembangkan ini hanya diperuntukkan bagi pengguna yang tidak memiliki gangguan atau kelainan pada fungsi penglihatan.
7. Terdapat 7 masukan gerakan mata yang akan dioperasikan oleh kursi roda, yaitu gerakan melihat lurus ke depan, lirik kiri, lirik kanan, lirik atas, lirik bawah, kedip sadar dan kedip tidak sadar.

### **1.5 Manfaat Penelitian**

Manfaat dari penelitian tugas akhir ini yaitu untuk mengembangkan inovasi pada sistem kursi roda pintar yang memanfaatkan sinyal mata dalam pengoperasiannya. Penelitian ini berusaha untuk meningkatkan performa navigasi kursi roda dengan meningkatkan akurasi klasifikasi 3 gerakan sinyal mata dengan polaritas yang sama serta merancang sistem yang dapat mengabaikan sinyal non-perintah seperti sinyal balik dan gerakan refleks mata. Secara keseluruhan, sistem ini diharapkan menjadi solusi mobilitas yang lebih nyaman dan efektif bagi penyandang disabilitas yang mengalami kelumpuhan pada tangan dan kaki.

### **1.6 Sistematika Penulisan**

Penulisan tugas akhir ini disusun ke dalam beberapa bab secara sistematis guna memudahkan pembaca dalam memahami isi karya tulis ini. Adapun sistematika penulisan tugas akhir ini adalah sebagai berikut:

- |         |  |
|---------|--|
| Bab I   | Pendahuluan, terdiri dari latar belakang, rumusan masalah, tujuan penelitian, batasan masalah, manfaat penelitian, dan sistematika penulisan. Bab ini merupakan gambaran umum penelitian yang akan dilakukan.            |
| Bab II  | Tinjauan Pustaka, berisi tentang teori-teori pendukung yang akan digunakan sebagai pedoman dalam pembuatan tugas akhir ini, seperti penjelasan komponen, prinsip kerja, dan metode yang akan digunakan dalam penelitian. |
| Bab III | Metodologi Penelitian, berisi penjelasan tentang jenis dan prosedur yang digunakan untuk memecahkan permasalahan yang diangkat dalam penelitian.   |
| Bab IV  | Hasil dan Pembahasan, bab ini berisi penjelasan mengenai hasil yang telah diperoleh dari penelitian ini. Serta menjabarkan pembahasan terkait hasil yang telah diperoleh selama penelitian.                              |

Bab V      Kesimpulan, bab yang berisi tentang kesimpulan dari penelitian ini. Serta berisi saran terhadap penelitian selanjutnya yang ingin melanjutkan topik ini.



## BAB II TINJAUAN PUSTAKA

### 2.1 Kursi Roda

Kursi roda merupakan alat bantu mobilitas yang digunakan oleh individu dengan keterbatasan pergerakan pada kaki, baik disebabkan oleh penyakit, cedera, maupun kelainan fisik [19]. Kursi roda bisa dioperasikan dengan tangan pengguna, didorong oleh pihak lain, atau dengan bantuan sistem elektrik [20]. Terdapat beberapa jenis kursi roda yaitu:

#### 2.1.1 Kursi Roda Konvensional

Kursi roda konvensional merupakan kursi roda yang dijalankan secara manual dengan memutar roda pada kursi menggunakan tangan oleh pengguna atau didorong oleh orang lain. Jenis kursi roda ini, tidak dapat dikendalikan sendiri apabila pengguna memiliki disabilitas pada tangannya [21]. Bentuk kursi roda konvensional ini dapat dilihat pada Gambar 2.1.



**Gambar 2.1** Kursi Roda Konvensional

#### 2.1.2 Kursi Roda Elektrik

Kursi roda elektrik adalah kursi roda yang digerakkan dengan motor listrik dan memiliki kontrol navigasi. Pergerakan kursi roda ini biasanya dikontrol dengan menggunakan tuas seperti *joystick* agar bisa bergerak maju, mundur, berbelok, dan berhenti. Pada kursi roda elektrik terdapat dua buah motor DC sebagai penggerak dan baterai sebagai sumber listrik [22]. Bentuk kursi roda elektrik ini dapat dilihat pada Gambar 2.2.



**Gambar 2.2** Kursi Roda Elektrik

## 2.2 *Electrooculography* (EOG)

*Electrooculography* (EOG) adalah teknik pengukuran potensial listrik yang terjadi dari pergerakan mata [23]. Pergerakan pada mata menyebabkan variasi potensial listrik antara retina dan kornea mata [24]. Perbedaan potensial listrik pada mata terjadi karena kornea memiliki muatan positif dan retina memiliki muatan negatif, seperti yang dapat dilihat pada Gambar 2.3 [25].



**Gambar 2.3** Polaritas kornea dan retina mata [26]

Pembacaan sinyal EOG dapat dilakukan dengan menggunakan beberapa elektroda yang diletakkan disekitar mata [27]. Polaritas sinyal yang dihasilkan oleh sensor EOG akan linear terhadap perputaran bola mata [28]. Selain dari perputaran bola mata, sinyal EOG juga dipengaruhi oleh aktivitas kelopak mata dan otot disekitar mata. Sinyal *Electrooculography* (EOG) memiliki spesifikasi nilai amplitudo yang bervariasi pada setiap orang, namun berada pada rentang nilai antara 50 hingga  $3500\mu V$  dan memiliki rentang frekuensi dari 0-100 Hz, dimana sebagian informasi yang berguna memiliki frekuensi tidak melebihi 38 Hz [29].

### 2.2.1 Elektroda

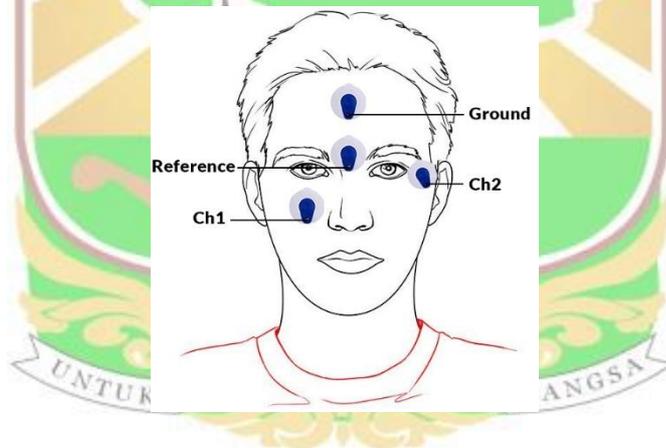
Elektroda adalah perangkat yang digunakan untuk membaca sinyal listrik pada tubuh [30]. Elektroda bekerja secara transduser, yaitu dengan mengubah arus listrik ion menjadi pergerakan elektron. Fungsi transduksi ini dilakukan oleh elektroda yang terdiri dari konduktor listrik yang bersentuhan dengan larutan ionik pada air di dalam tubuh. Interaksi antara elektron dalam elektroda dan ion dalam tubuh dapat sangat mempengaruhi kinerja dari elektroda [31].

Pada penelitian ini, digunakan jenis elektroda permukaan tubuh (*surface electrode*). *Surface electrode* adalah jenis elektroda langsung dipasangkan ke permukaan tubuh yang akan diukur sinyal listriknya [32]. Elektroda ini berbentuk *disk*, yang terdiri dari *silver-plated disk* sebagai elektroda yang dilapisi dengan AgCl dan elektrolit gel yang menutupinya, pada bagian luaran terdapat busa plastik yang diberi perekat untuk ditempelkan ke kulit [33]. Bentuk dari elektroda ini dapat dilihat pada Gambar 2.4.



**Gambar 2. 4** Elektroda *Surface*[34]

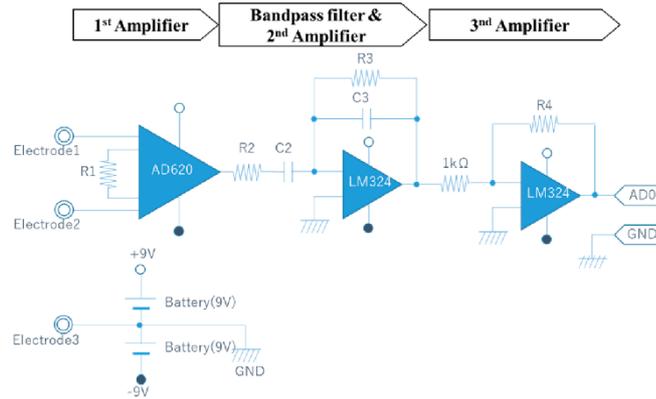
Sinyal EOG didapatkan dengan memasang empat buah elektroda *surface* disekitar area mata. Elektroda ini terdiri dari Ch1, Ch2, Channel Referensi dan Ground. Penempatan elektroda pada penelitian ini dapat dilihat pada Gambar 2.5 berikut.



**Gambar 2. 5** Posisi Elektroda

### 2.2.2 Rangkaian EOG

Rangkaian EOG terdiri dari rangkaian *amplifier* dan *filter*. Rangkaian *amplifier* atau rangkaian penguat diperlukan karena sinyal EOG memiliki rentang nilai yang terlalu kecil ( $50\text{-}3500\mu\text{V}$ ), maka diperlukan penguatan agar sinyal EOG dapat dibaca oleh inputan analog dari perangkat akuisisi data. Rangkaian *filter* diperlukan agar menghilangkan frekuensi *noise* yang tidak diinginkan dan mempertahankan frekuensi yang diinginkan ( $0,3\text{-}15\text{Hz}$ ) dari sinyal EOG mentah [35]. Rangkaian EOG ini dapat dilihat pada Gambar 2.6, yang terdiri dari rangkaian penguat pertama, rangkaian *band pass filter* dan penguat kedua, dan rangkaian penguat ketiga.



**Gambar 2. 6** Rangkaian EOG

Pada penguat pertama digunakan IC AD620AN sebagai komponen *amplifier*. IC AD620N adalah IC yang memiliki keakuratan yang tinggi untuk sebuah instrumentasi *amplifier* yang membutuhkan satu resistor eksternal untuk membuat penguatan dari 1 – 10000 [36]. Penguatan pada IC ini tergantung terhadap besarnya tahanan R1 berdasarkan persamaan 2.1 berikut.

$$G = \frac{49.4k\Omega}{R_G} + 1 \quad (2.1)$$

Pada penguat kedua dan ketiga digunakan IC LM324 sebagai komponen *amplifier*. IC LM324 merupakan penguat operasional dengan biaya rendah, quad dengan input diferensial sejati. *Quad Operational Amplifiers* ini dapat beroperasi pada rentang operasional yang luas yaitu antara 3 – 32 V dengan arus pada mode siaga (*quiescent current*)[36] . Penguatan pada IC ini tergantung terhadap resistor yang digunakan berdasarkan persamaan 2.2 dan 2.3 berikut.

Besarnya penguatan pada penguat kedua

$$G = \frac{R3}{R2} \quad (2.2)$$

Besarnya penguatan pada penguat ketiga

$$G = \frac{R5}{R4} \quad (2.3)$$

*filter* yang diterapkan pada rangkaian ini adalah *band pass filter*. *Band pass filter* merupakan sebuah rangkaian *filter* melewatkan sinyal yang berada pada frekuensi diantara frekuensi *cut-off* bawah dan frekuensi *cut-off* atas [37]. Pada rangkaian ini, nilai frekuensi *cut-off* bawah dan atas didapatkan berdasarkan persamaan 2.4 dan 2.5 berikut.

Frekuensi *low pass filter* (frekuensi *cut off high*)

$$f = \frac{1}{2\pi \cdot R3 \cdot C3} \quad (2.4)$$

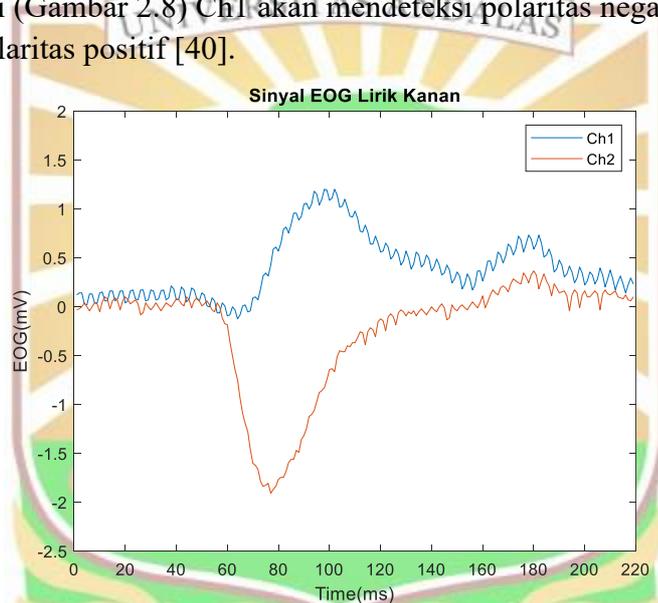
Frekuensi *high pass filter* (frekuensi *cut off low*)

$$f = \frac{1}{2\pi \cdot R2 \cdot C2} \quad (2.5)$$

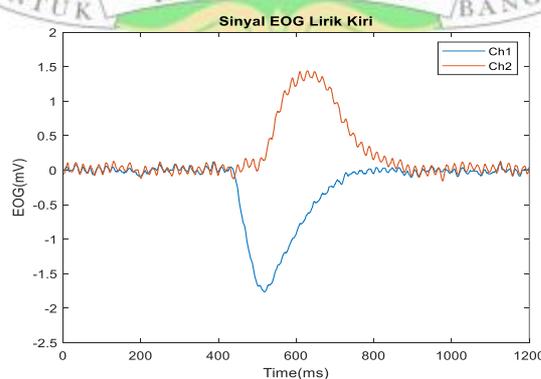
### 2.2.3 Bentuk Sinyal EOG

Pergerakan mata yang berbeda akan menghasilkan bentuk sinyal EOG yang berbeda juga. Hal ini dapat dilihat berdasarkan potensial yang ditangkap oleh sensor EOG pada mata, apabila lebih dekat ke kornea maka sensor akan menghasilkan potensial listrik positif, sedangkan apabila lebih dekat ke retina maka sensor akan menghasilkan potensial listrik negatif [38]. Pada umumnya sinyal EOG dibuat dengan dua pergerakan mata, satu digunakan untuk membangkitkan sinyal dan satu digunakan untuk mengembalikan sinyal ke posisi normal [39].

Berdasarkan posisi elektroda pada Gambar 2.5, saat pergerakan mata ke kanan (Gambar 2.7), elektroda Ch1 mendeteksi polaritas terdekat yaitu kornea mata kanan sehingga Ch1 akan berpolaritas positif, sebaliknya pada elektroda Ch2 akan mendeteksi retina sehingga Ch2 akan berpolaritas negatif. Begitu juga saat mata bergerak ke kiri (Gambar 2.8) Ch1 akan mendeteksi polaritas negatif dan Ch2 akan mendeteksi polaritas positif [40].



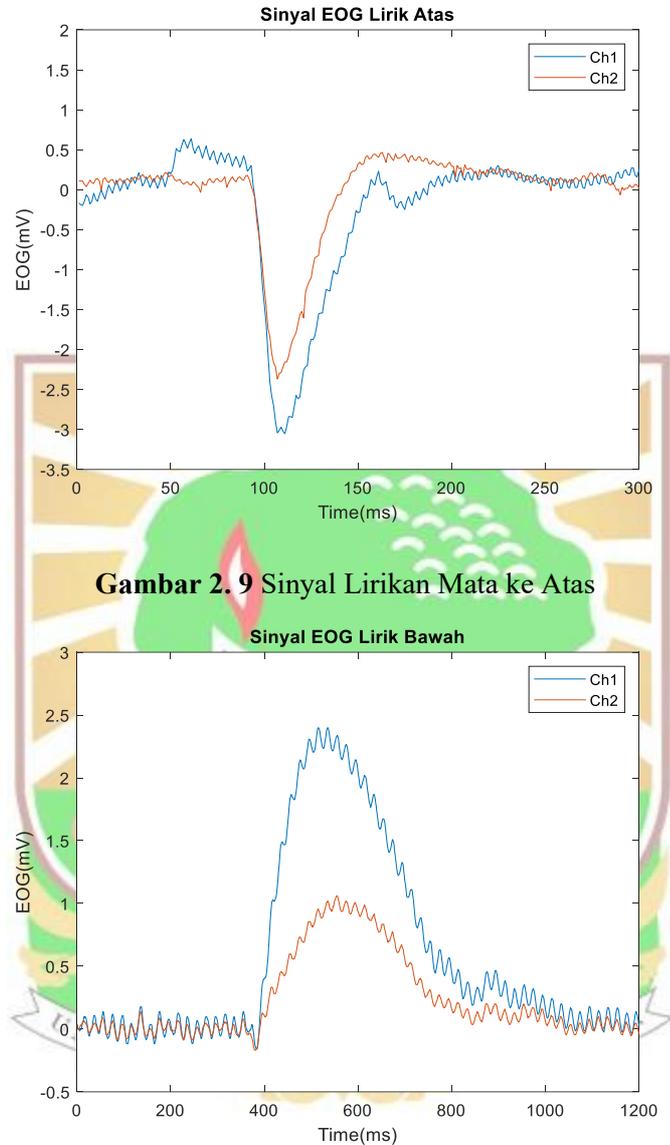
**Gambar 2. 7** Sinyal Lirik Mata ke Kanan



**Gambar 2. 8** Sinyal Lirik Mata ke Kiri

Selanjutnya, sinyal EOG untuk aktivitas lirik mata ke atas (Gambar 2.9) dan ke bawah (Gambar 2.10) juga dapat dibedakan berdasarkan polaritas sinyal

pada Ch1 dan Ch2. Saat mata melirik ke atas, maka Ch1 dan Ch2 akan mendeteksi polaritas negatif dari retina mata sehingga keduanya berpolaritas negatif, sedangkan saat melirik ke bawah, maka Ch1 dan Ch2 mendeteksi polaritas positif dari kornea mata sehingga sinyalnya berpolaritas positif [40].

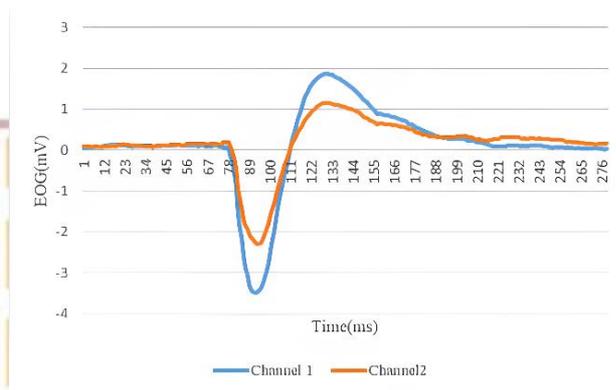


**Gambar 2. 9** Sinyal Lirik Mata ke Atas

**Gambar 2. 10** Sinyal Lirik Mata ke Bawah

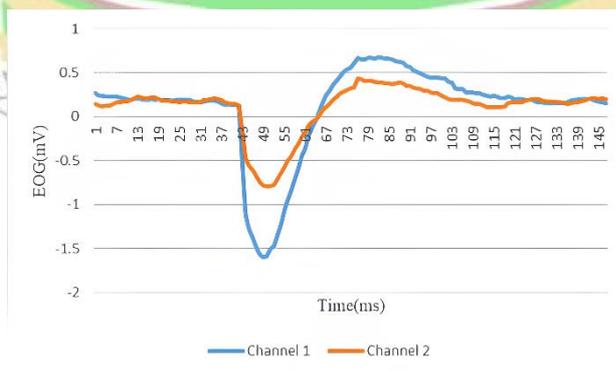
Sinyal EOG juga mampu mengidentifikasi sinyal mata yang dihasilkan saat terjadi kedipan. Kedipan mata merupakan hasil kontraksi otot-otot di sekitar kelopak mata yang memicu terbentuknya potensial listrik [41]. Rentang waktu gelombang sinyal berkedip sangat cepat dibandingkan dengan pergerakan mata secara vertikal dan horizontal. Rentang waktu yang cepat disebabkan oleh kecepatan tinggi dari kedipan mata sehingga durasi sinyal berlangsung dalam sepersekian detik. Frekuensi informasi saat sinyal mata berkedip berada dalam rentang 0,5 – 15 Hz [42].

Kedipan mata dibagi menjadi dua, yaitu kedipan spontan atau tidak sadar (*involuntary blink*) dan kedipan mata sengaja atau sadar (*voluntary blink*)[41]. Kedipan tidak sadar adalah kedipan yang terjadi tanpa rangsangan eksternal yang dikendalikan oleh generator sinyal *blink* di otak [41], kedipan ini bisa mengindikasikan mata kelelahan dan biasanya memiliki durasi sinyal yang lebih pendek dari kedipan reflek dan kedip sadar. Kedipan sadar adalah kedipan yang terjadi karena keputusan untuk berkedip. Kedipan sadar ini menghasilkan amplitudo yang lebih besar dibandingkan kedipan tidak sadar [14]. Saat mata berkedip dengan sengaja atau sadar, maka akan diperoleh sinyal EOG seperti yang ditunjukkan pada Gambar 2.11.



**Gambar 2. 11** Sinyal Berkedip Secara Sadar

Sinyal EOG kedipan spontan atau tidak sadar dianggap sebagai *noise* karena memiliki amplitudo pulsa yang kecil dan cenderung tidak konstan sehingga terkadang memiliki bentuk sinyal yang mirip dengan sinyal kedip sadar dengan amplitudo yang kecil. Bentuk dari sinyal kedip tidak sadar dapat dilihat pada Gambar 2.12.



**Gambar 2. 12** Sinyal Berkedip Secara Tidak Sadar

#### 2.2.4 Fitur Sinyal EOG

Pada sinyal EOG dibutuhkan ekstraksi fitur untuk mengambil ciri sinyal yang dapat menggambarkan karakteristik dari sinyal tersebut [43]. Sinyal EOG memiliki beberapa jenis fitur, yang dapat digunakan untuk mengenali dan

mengklasifikasikan beberapa gerakan pada mata [44]. Fitur ini terdiri dari luas sinyal, puncak sinyal, polaritas sinyal, dan perioda sinyal.

#### 1. Polaritas Sinyal

Polaritas sinyal EOG diperoleh berdasarkan nilai *threshold* yang telah dilewati. *Threshold* merupakan nilai batas (*cut off* atau *limit*) yang didapatkan dari hasil rentang nilai maksimum dan minimum dari sampel data sinyal mentah saat mata tidak melakukan gerakan (hanya melihat lurus ke depan). Nilai maksimum tersebut akan menjadi *threshold* positif dan nilai minimum tersebut akan menjadi *threshold* negatif.

Polaritas terbagi menjadi 2 jenis, yaitu polaritas positif dan polaritas negatif. Sinyal dikatakan berpolaritas positif apabila kurva suatu sinyal melewati nilai *threshold* positif. Sebaliknya, sinyal dikatakan berpolaritas negatif apabila kurva suatu sinyal melewati nilai *threshold* negatif [40]. Pada penelitian ini, nilai *threshold* yang digunakan adalah  $+0.2$  untuk polaritas positif dan  $-0.5$  untuk polaritas negatif. Polaritas sinyal yang diperoleh akan digunakan untuk mendeteksi sinyal lirik kiri, lirik kanan, dan lirik bawah.

#### 2. Luas Sinyal

Luas sinyal adalah daerah yang berada dibawah kurva sinyal EOG. Perhitungan luas sinyal dimulai dengan menjumlahkan seluruh nilai sinyal Ch1 dan Ch2 yang melewati nilai *threshold*. Perhitungan ini akan selesai saat sinyal kembali bernilai nol [44].

#### 3. Puncak Sinyal

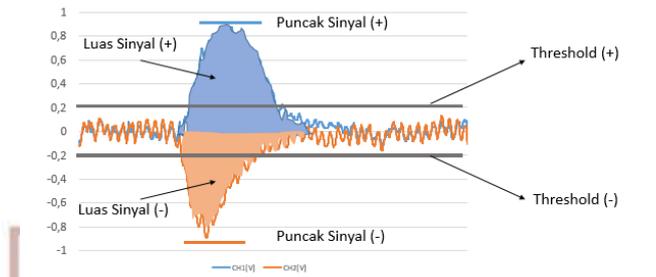
Puncak sinyal adalah amplitudo tertinggi dari sinyal yang memiliki bentuk setengah gelombang. Puncak ini terdiri menjadi dua, yaitu puncak sinyal positif, yang merupakan sinyal tertinggi yang melewati *threshold* positif dan puncak sinyal negatif, yang merupakan sinyal tertinggi yang melewati *threshold* negatif [44]. Pada penelitian ini digunakan fitur puncak sebagai input klasifikasi sinyal EOG untuk membedakan sinyal lirik atas, kedip sadar, dan kedip tidak sadar.

Puncak sinyal diperoleh dengan membandingkan nilai baru dari sebuah channel dengan nilai sebelumnya [44]. Nilai puncak positif didapatkan dari nilai sinyal sebuah channel yang telah melewati *threshold* positif, nilai tersebut akan terus dibandingkan dengan nilai terbaru pada channel tersebut hingga didapatkan nilai paling tinggi yang merupakan nilai puncak sinyal positif. Begitu juga untuk nilai puncak negatif yang didapatkan dari nilai sinyal sebuah channel yang telah melewati *threshold* negatif, nilai tersebut akan terus dibandingkan dengan nilai terbaru pada channel tersebut hingga

didapatkan nilai paling rendah yang merupakan nilai puncak sinyal negatif.

#### 4. Periode Sinyal

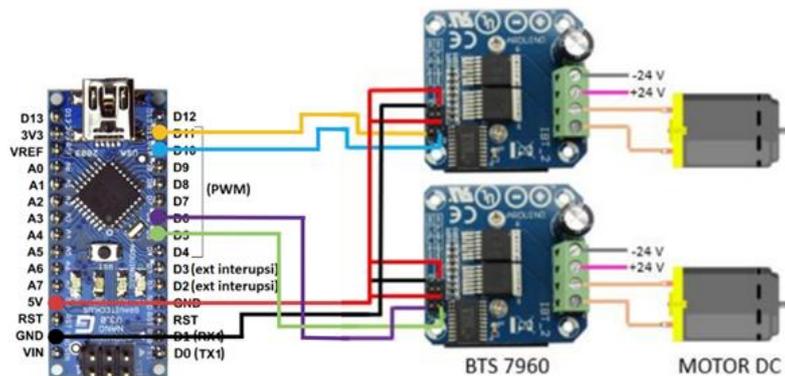
Periode sinyal merupakan durasi waktu yang diperlukan untuk membentuk sebuah sinyal utuh. Periode ini terjadi dalam waktu yang cepat dengan skala nilai *millisecond* (ms). Bentuk fitur sinyal secara keseluruhan dapat dilihat pada Gambar 2.13.



Gambar 2.13 Grafik Fitur Sinyal [45]

### 2.3 Rangkaian Kursi Roda

Rangkaian kursi roda terdiri dari 3 komponen utama, yaitu mikrokontroler, driver motor, dan motor DC. Mikrokontroler berfungsi sebagai pemberi sinyal masukan untuk driver motor dan diproses menjadi pergerakan motor pada kursi roda. Driver motor yang digunakan jenis motor BTS7960 dan motor yang digunakan jenis PG45 Series 775. Rangkaian kursi roda ini dapat dilihat pada Gambar 2.14 berikut.



Gambar 2.14 Rangkaian Kursi Roda [18]

### 2.4 NI USB 6008

NI USB 6008 merupakan perangkat DAQ (*data acquisition*) multifungsi yang menyediakan input/output dari perusahaan National Instruments [46]. NI USB-6008 digunakan untuk proses akuisisi data dalam merekam sinyal, termasuk sinyal electrooculography (EOG). Data yang dibaca oleh NI USB-6008 kemudian

dikirim ke komputer untuk diproses sesuai kebutuhan. Driver NI-DAQmx dan utilitas konfigurasi yang disediakan mempermudah proses konfigurasi dan pengukuran saat menggunakan USB-6008 [47].



**Gambar 2. 15** NI USB-6008 [47]

NI USB-6008 memiliki spesifikasi sebagai berikut [47]:

1. Tegangan *Input*: 1 V – 20 V
2. Analog *Input* (AI): NI USB-6008 memiliki 4 pin *differential input* dan 8 pin *single ended input*, dengan resolusi 12 bit dan 11 bit.
3. Analog *Output* (AO): NI USB-6008 memiliki 2 pin keluaran dengan resolusi 12 bit dengan rentang tegangan 0 V – 5 V, yang memiliki kecepatan output 150 Hz (sampel/detik)
4. *Digital Input* (DI) dan *Digital Output* (DO) pada NI USB-6008 terdiri dari 12 saluran pin yang dapat dikonfigurasi sesuai kebutuhan. Tegangan input rendah (*low*) berkisar antara -0,5 V hingga +0,8 V, sementara tegangan input tinggi (*high*) berada pada rentang +2,0 V hingga +5,8 V.
5. Counter pada NI USB-6008 dilengkapi dengan 1 pin counter yang memiliki resolusi 32 bit dan rentang tegangan kerja antara 0,8 V hingga 2 V.

## 2.5 Random Forest Classifier

*Random Forest Classifier* adalah metode klasifikasi yang terdiri dari beberapa kumpulan pohon keputusan (*decision tree*) yang menerapkan prinsip *majority vote* (suara terbanyak) untuk menentukan suatu keputusan akhir dalam klasifikasi. Pohon keputusan ini terdiri dari tiga bagian utama, yaitu *root node*, *internal node*, dan *leaf node* [48]. Proses pemisahan data pada setiap node dalam pohon keputusan dapat ditentukan dengan dua metode, yaitu menggunakan *nilai entropy* dan *information gain* (*gini index* dan *gini split*).

Entropy merupakan ukuran ketidakpastian atau *impurity* suatu himpunan data. Nilai entropy akan bernilai 0 apabila semua data dalam satu node berasal dari kelas yang sama (homogen), dan semakin tinggi apabila distribusi kelas semakin merata. Berikut merupakan persamaan dari *entropy* pada persamaan 2.6.

$$Entropy(s) = \sum_{i=1}^n -P_i \log_2 P_i \quad (2.6)$$

Sementara itu, Information Gain digunakan untuk memilih fitur terbaik dalam pemisahan data. Fitur dengan nilai *information gain* tertinggi dianggap paling efektif dalam mengurangi impurity. Rumus *information gain* dapat dilihat pada persamaan 2.7.

$$Gain(S, J) = Entropy(s) - \sum_{i=1}^n \frac{|S_i|}{|S|} \times Entropy(S_i) \quad (2.7)$$

Gini Impurity merupakan pengukuran seberapa tidak murni suatu node dalam pohon keputusan. Node dikatakan "murni" jika seluruh data di dalamnya berasal dari satu kelas. Semakin tinggi nilai Gini Impurity, semakin bercampur kelas-kelas di dalam node tersebut. Rumus *Gini Impurity* dapat dilihat pada persamaan 2.8, sedangkan rumus *gini split* dapat dilihat pada persamaan 2.9.

$$Gini(s) = 1 - \sum_{i=1}^n (P_i)^2 \quad (2.8)$$

$$Gini_{split} = \sum_{i=1}^n \frac{|S_i|}{|S|} \times Gini(S_i) \quad (2.9)$$

*Random Forest* bekerja berdasarkan teknik dasar yang dikenal sebagai Bagging (*Bootstrap Aggregating*). Bagging adalah metode *ensemble* yang bertujuan mengurangi variansi dan meningkatkan akurasi model prediksi dengan menggabungkan beberapa model yang dilatih pada subset data berbeda [49]. Berikut langkah-langkah dalam bagging:

1. Pemilihan Subset Data

Diambil subset data secara acak dengan penggantian dari dataset asli (proses ini disebut bootstrapping).

2. Pelatihan Model Independen

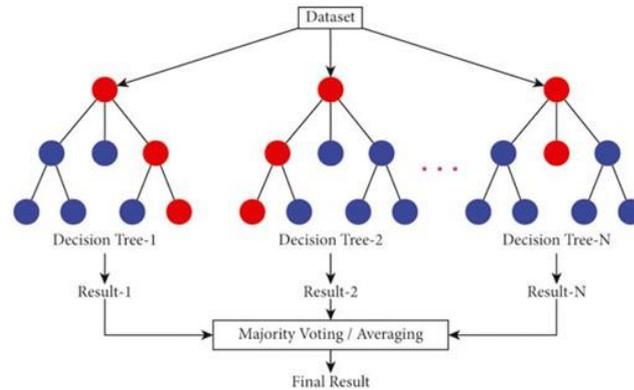
Masing-masing subset digunakan untuk melatih model (decision tree) secara terpisah.

3. Prediksi dan Agregasi

Untuk klasifikasi, hasil akhir ditentukan dengan pemungutan suara mayoritas. Sedangkan untuk regresi, digunakan rata-rata prediksi (agregasi).

Cara kerja Random Forest secara keseluruhan diawali dengan menentukan jumlah pohon yang akan dibangun sebanyak  $n$ . Selanjutnya, dilakukan teknik *bagging* dengan cara mengambil sampel acak dari fitur (kolom) dan sampel baris (data) untuk dilakukan pemodelan menggunakan *decision tree*, sehingga terbentuk banyak pohon sesuai jumlah yang ditentukan. Proses pembentukan pohon menggunakan *entropy* dan *information gain*, atau dapat juga menggunakan *gini index* dan *gini split*, untuk menentukan pemisahan data terbaik di setiap node.

Setelah seluruh pohon selesai dibangun, hasil akhir prediksi diperoleh berdasarkan mayoritas suara (*majority vote*) dari semua pohon yang telah terbentuk [50]. Jalur kerja algoritma menggunakan *random forest* ditunjukkan pada gambar 2.16 berikut.



**Gambar 2.16** Struktur Skema kerja *Random Forest*

## 2.6 Confusion Matrix

Evaluasi yang dapat digunakan untuk mengukur kinerja klasifikasi dapat menggunakan *confusion matrix*. *Confusion matrix* mengandung informasi berupa perbandingan hasil klasifikasi yang dilakukan oleh sistem dengan hasil klasifikasi sebenarnya. *Confusion matrix* dilakukan untuk mengetahui sejauh mana model bekerja sesuai dengan apa yang kita inginkan [51]. *Confusion matrix* dapat dinyatakan pada Tabel 2.1.

**Tabel 2.1** Confusion Matrix

Aktual	Prediksi	
	Negatif (0)	Positif (1)
Negatif (0)	TN	FP
Positif (1)	FN	TP

Penjelasan dari TN, FN, FP, dan TP adalah sebagai berikut :

1. True Negative (TN) : Data negatif yang diprediksi benar sebagai negatif.
2. False Positive (FP) : Data negatif yang salah diprediksi sebagai positif.
3. False Negative (FN) : Data positif yang salah diprediksi sebagai negatif.
4. True Positive (TP) : Data positif yang diprediksi benar sebagai positif.

Ukuran performa yang dapat digunakan untuk melihat hasil evaluasi model dapat menggunakan *accuracy* (persamaan 2.10), *recall* (persamaan 2.11), *precision* (persamaan 2.12), dan *F1-score* (persamaan 2.13)[52] .

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \cdot 100\% \quad (2.10)$$

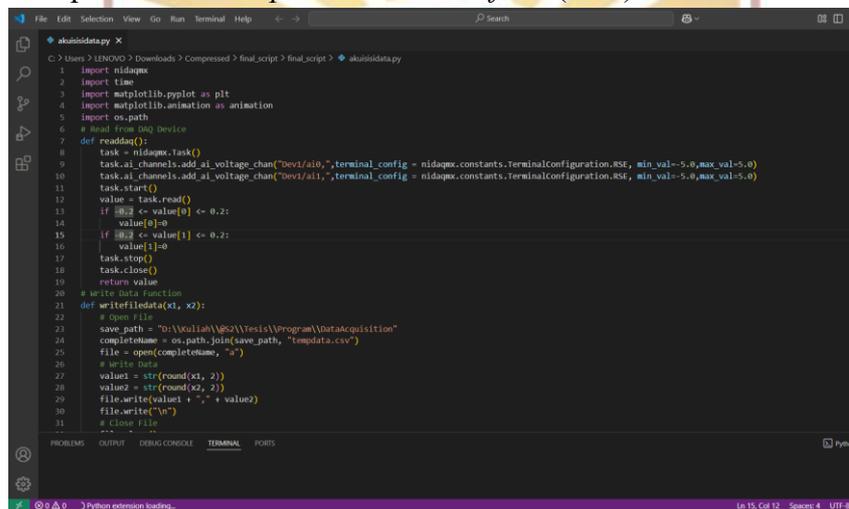
$$Recall = \frac{TP}{TP + FN} \cdot 100\% \quad (2.11)$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \cdot 100\% \quad (2.12)$$

$$\text{F1 - Score} = \frac{2 \times \text{precision} \times \text{recall}}{\text{precision} + \text{recall}} \quad (2.13)$$

## 2.7 Visual Studio Code

Visual studio code merupakan perangkat code editor dari perusahaan microsoft. Perangkat ini didukung oleh beberapa sistem operasi seperti windows, Mac OS , dan linux. Visual studio digunakan untuk membuat program pengembangan software pada komputer yang dapat digunakan dalam banyak bahasa pemrograman seperti C++, Python, Java,dll [53]. Berikut tampilan aplikasi Visual Studio pada gambar 2.17 berikut. Pada penelitian ini, *Visual Studio Code* digunakan untuk membuat program menggunakan bahasa *Python* yang mencakup akuisisi data sinyal *EOG*, pengiriman data, klasifikasi dengan algoritma *Random Forest*, serta pembuatan *Graphical User Interface* (GUI).



```

1 import nidaqmx
2 import time
3 import matplotlib.pyplot as plt
4 import matplotlib.animation as animation
5 import os.path
6 # Read from NIQ Device
7 def readdata():
8     task = nidaqmx.Task()
9     task.ai_channels.add_ai_voltage_chan("Dev1/aio", terminal_config = nidaqmx.constants.TerminalConfiguration.RSE, min_val=-5.0, max_val=5.0)
10    task.ai_channels.add_ai_voltage_chan("Dev1/ai1", terminal_config = nidaqmx.constants.TerminalConfiguration.RSE, min_val=-5.0, max_val=5.0)
11    task.start()
12    value = task.read()
13    if value[0] <= 0.2:
14        value[0]=0
15    if value[1] <= 0.2:
16        value[1]=0
17    task.stop()
18    task.close()
19    return value
20 # Write Data Function
21 def writefiledata(x1, x2):
22     # Open File
23     save_path = "D:\Vallah\@G2\Tesis\Program\DataAcquisition"
24     completeName = os.path.join(save_path, "tempdata.csv")
25     file = open(completeName, "a")
26     # Write Data
27     value1 = str(round(x1, 2))
28     value2 = str(round(x2, 2))
29     file.write(value1 + "," + value2)
30     file.write("\n")
31     # Close File

```

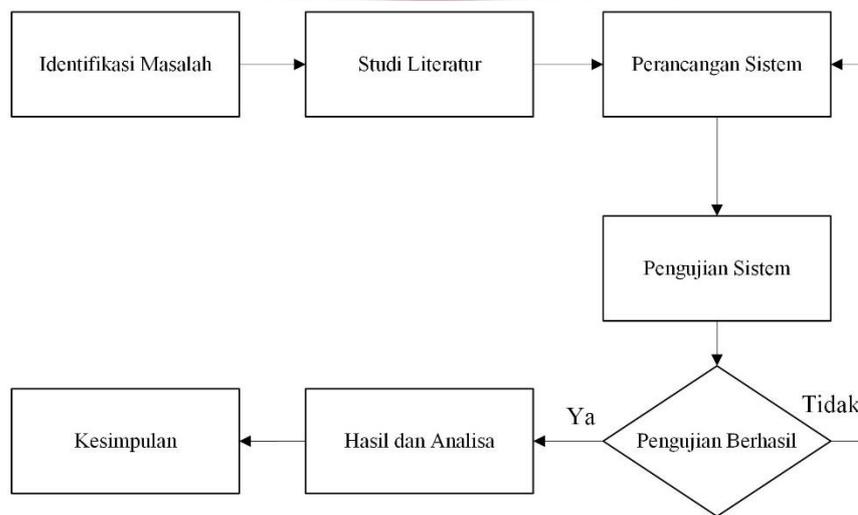
Gambar 2.17 Tampilan Interface Visual Studio Code



## BAB III METODOLOGI PENELITIAN

### 3.1 Jenis dan Tahapan Penelitian

Penelitian tugas akhir ini menggunakan jenis penelitian eksperimental (*experimental research*). Penelitian eksperimental ini bertujuan untuk menguji hasil yang didapatkan setelah mengimplementasikan suatu rancangan. Terdapat beberapa tahapan dalam penelitian ini, yaitu identifikasi masalah, studi literatur, perancangan sistem, pengujian sistem, hasil dan analisis, serta kesimpulan. Tahapan penelitian ini dapat dilihat pada Gambar 3.1.



Gambar 3. 1 Diagram Alir Alir Tahapan Penelitian

Penjelasan dari diagram alir tahapan penelitian ini adalah sebagai berikut:

1. Identifikasi Masalah

Identifikasi masalah bertujuan untuk memastikan bahwa penelitian yang akan dilakukan benar-benar baru dan belum pernah dilakukan sebelumnya. Proses ini dilakukan dengan mengumpulkan informasi dari penelitian terdahulu yang masih memiliki kekurangan, sehingga penelitian tersebut dapat dikembangkan lebih lanjut.

2. Studi Literatur

Studi literatur dilakukan untuk mengumpulkan informasi dan teori yang relevan dengan topik penelitian. Sumber informasi dan teori tersebut dapat diperoleh dari buku, jurnal, artikel, internet, maupun laporan penelitian terdahulu.

### 3. Perancangan Sistem

Perancangan sistem dalam penelitian ini mencakup perancangan perangkat keras dan perangkat lunak. Perancangan perangkat keras bertujuan untuk menggambarkan keseluruhan bentuk dari perangkat keras yang akan digunakan. Sedangkan perancangan perangkat lunak bertujuan untuk menjelaskan sistem klasifikasi gerakan mata menggunakan metode *random forest*. Selain itu, program juga dilengkapi dengan fitur *signal locking time* untuk mendukung pengoperasian kursi roda.

### 4. Pengujian Sistem

Pengujian sistem dilakukan untuk mengevaluasi performa dari sistem yang telah dirancang. Sebelum tahap pengujian, program diimplementasikan ke dalam sistem kursi roda. Pengujian dilaksanakan melalui percobaan pergerakan secara realtime pada kursi roda yang telah dibuat. Selanjutnya, performa kursi roda akan dianalisis untuk memastikan kesesuaiannya dengan sistem yang telah dirancang. Apabila pengujian sistem belum sesuai dengan yang diharapkan, maka dilakukan perancangan sistem kembali dengan melakukan evaluasi.

### 5. Hasil dan Analisis

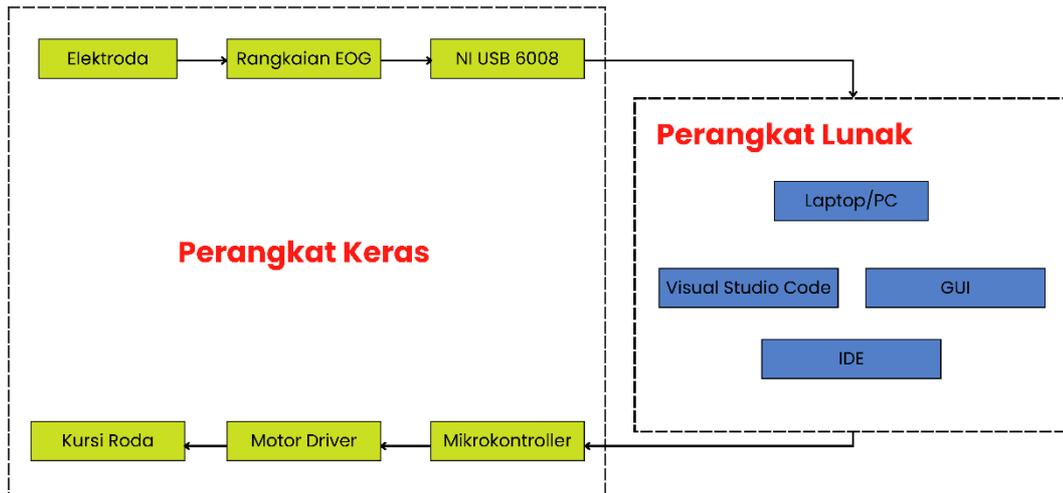
Hasil yang didapatkan, berasal dari data pengujian sistem yang telah dilakukan. Selanjutnya hasil yang diperoleh akan dianalisis sehingga akan menghasilkan kesimpulan mengenai kinerja sistem secara keseluruhan.

### 6. Kesimpulan

Tahap ini berfungsi sebagai pelaporan hasil dan analisis yang dapat disimpulkan dari penelitian tugas akhir. Hasil yang diperoleh pada tahap ini menjadi dasar untuk mengevaluasi keberhasilan sistem dan memberikan rekomendasi untuk pengembangan lebih lanjut.

## 3.2 Perancangan Sistem

Terdapat dua perancangan sistem yang dilakukan pada penelitian tugas akhir ini, yaitu perancangan perangkat keras dan perangkat lunak. Secara keseluruhan tahapan perancangan sistem ini dapat dilihat pada Gambar 3.2 berikut.



**Gambar 3.2** Diagram Alir Tahapan Perancangan Sistem

Gambar 3.2 merupakan diagram alir dari tahapan perancangan sistem kendali kursi. Penjelasan dari tahapan perancangan sistem ini adalah sebagai berikut:

1. Elektroda

Elektroda adalah perangkat yang berfungsi untuk menangkap sinyal dari pergerakan mata. Jenis elektroda yang digunakan dalam penelitian ini adalah elektroda permukaan tubuh (*surface electrode*). Elektroda ditempatkan di area sekitar mata dengan posisi peletakan yang mengacu pada penelitian sebelumnya. Selanjutnya, elektroda tersebut dihubungkan ke rangkaian EOG.

2. Rangkaian EOG

Rangkaian EOG merupakan rangkaian yang berfungsi untuk memperkuat dan menyaring *noise* dari sinyal yang ditangkap oleh elektroda. Rangkaian ini terdiri dari rangkaian *amplifier* (penguat sinyal) dan rangkaian *filter* (penyaring sinyal).

3. NI USB-6008

NI USB-6008 adalah perangkat keras yang berfungsi untuk mengkonversi sinyal analog ke digital dari rangkaian EOG. Perangkat ini membutuhkan laptop untuk memproses data. Selanjutnya data yang telah diproses digunakan untuk mengendalikan kursi roda.

4. Laptop

Laptop merupakan perangkat keras yang berfungsi menerima dan mengolah data berupa sinyal EOG dari NI USB-6008, lalu mengirimkannya kembali ke NI USB-6008. Proses tersebut dilakukan dengan menggunakan *software* Visual Studio Code. Berikut proses yang terjadi di laptop:

a. Ekstraksi fitur

Sinyal EOG dari NI USB-6008 akan diproses dengan melakukan ekstraksi fitur. Ekstraksi fitur ini, bertujuan untuk menentukan ciri suatu sinyal yang dapat membedakan jenis pergerakan mata. Terdapat dua fitur sinyal yang digunakan pada penelitian ini, yaitu fitur puncak dan polaritas sinyal.

1) Deteksi jenis pergerakan mata

Setelah diperoleh nilai puncak dari pergerakan mata, maka akan dilakukan pendeteksian jenis pergerakan mata menggunakan klasifikasi nilai puncak sinyal dengan metode *random forest* untuk membedakan sinyal kedip sadar, kedip tidak sadar, dan lirik atas, serta polaritas sinyal untuk membedakan lirik kiri, lirik kanan, dan lirik bawah.

2) Pengambilan keputusan

Setelah jenis pergerakan mata dideteksi, maka akan dilakukan pengambilan keputusan berdasarkan masing-masing jenis pergerakan mata untuk mengoperasikan kursi roda. Apabila terdeteksi jenis pergerakan mata melihat ke atas, maka akan membuat keputusan untuk kursi roda maju, begitu juga untuk jenis pergerakan mata lainnya. Keputusan akan dikirimkan ke NI USB-6008 untuk dieksekusi ke dalam sistem pergerakan kursi roda.

3) *Locking Time*

Setelah pengambilan keputusan dilakukan, sistem akan mengaktifkan mekanisme *locking time* dengan waktu tunda untuk mengantisipasi kemungkinan terjadinya sinyal non-perintah yang dapat memengaruhi keputusan yang telah diambil sebelumnya. Selama periode *locking* ini, sistem akan menahan setiap deteksi pergerakan mata, sehingga tidak akan ada pengambilan keputusan baru. Waktu tunda akan disesuaikan secara dinamis berdasarkan analisis kondisi sinyal mata, untuk memastikan bahwa sinyal balik atau pergerakan mata yang tidak diinginkan tidak terdeteksi sebagai perintah baru.

5. Visual Studio Code

*Visual studio code* merupakan *software IDE (Integrated Development Environment)* yang berguna untuk membuat program akuisisi data sinyal EOG, pengiriman data, klasifikasi pergerakan lirikan mata dengan *random forest*, serta *locking time* dengan menggunakan bahasa pemrograman *Python* yang dapat dilihat pada lampiran B. Pada Visual studio code juga digunakan untuk membuat *Graphical User Interface (GUI)* yang menampilkan arah pergerakan kursi roda.

## 6. Graphical User Interface (GUI)

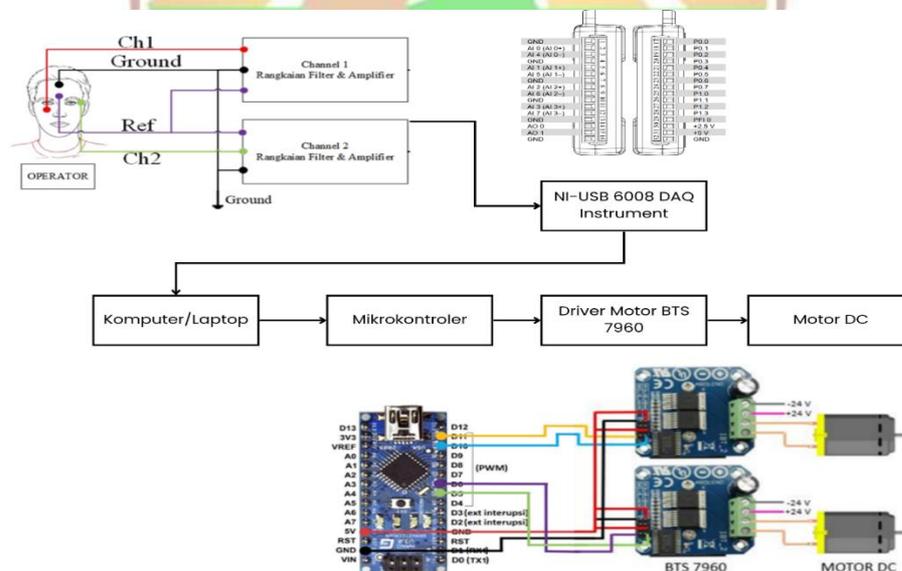
GUI digunakan untuk menampilkan hasil perekaman sinyal EOG secara realtime. Pada tampilan GUI juga akan ditampilkan arah pergerakan kendali kursi roda secara real time.

## 7. Kursi Roda

Pada kursi roda terdapat dua motor DC dan dua motor *driver* serta *mikrokontroler* yang akan dijalankan berdasarkan perintah dari NI USB-6008. Perintah tersebut berasal dari hasil klasifikasi sinyal EOG yang telah diproses melalui perangkat lunak di laptop.

### 3.2.1 Perancangan Perangkat Keras

Pada perancangan kursi roda, terdapat beberapa perangkat keras dengan fungsi yang berbeda. Perangkat keras untuk kursi roda ini, terdiri dari elektroda, rangkaian EOG, NI USB-6008, dan laptop, mikrokontroler driver motor, dan motor DC. Pada awalnya elektroda akan membaca sinyal EOG dari mata, kemudian sinyal EOG akan masuk ke rangkaian EOG untuk dilakukan proses penguatan dan filtering agar mendapatkan sinyal yang berkualitas dan mudah dianalisis. Selanjutnya sinyal dari rangkaian EOG dikirim ke perangkat akuisisi data, yakni NI USB-6008. Sinyal dari NI USB-6008 akan diolah pada laptop menggunakan *software* Visual Studio Code untuk mendapatkan hasil keputusan. Kemudian hasil keputusan akan dikirimkan ke rangkaian kursi roda, sehingga mikrokontroler dapat mengatur driver motor dan menggerakkan motor DC (kursi roda) berdasarkan hasil keputusan tersebut. Secara keseluruhan rancangan perangkat keras ini dapat dilihat pada Gambar 3.3 berikut.

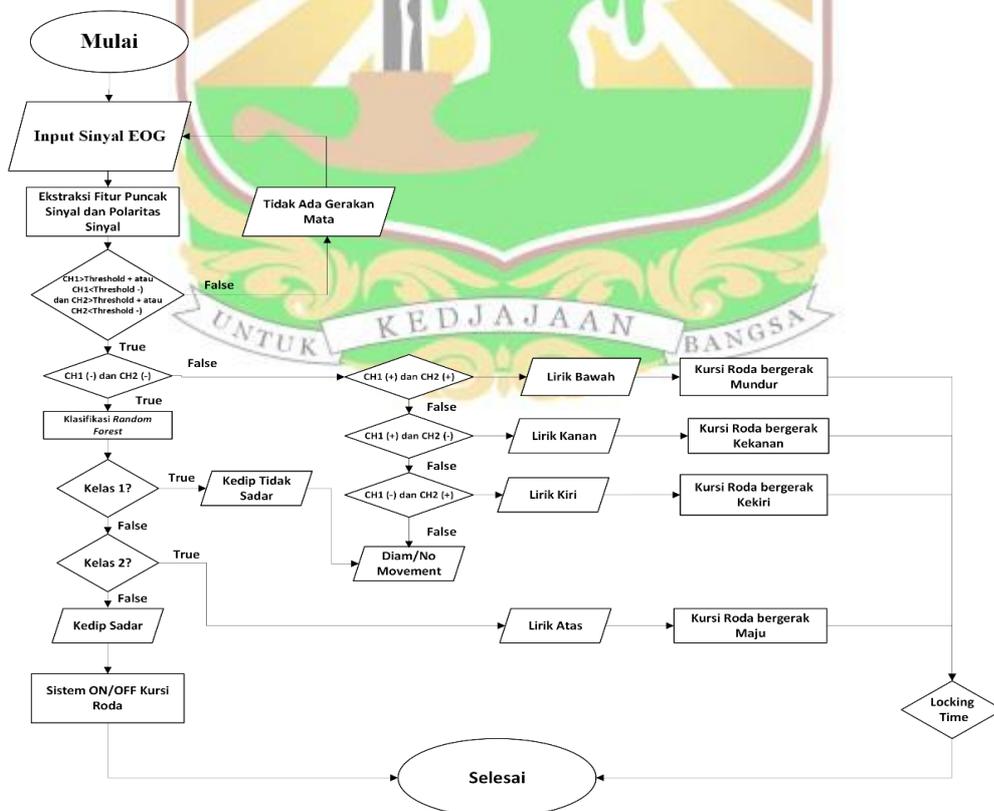


Gambar 3. 3 Rancangan Perangkat Keras

### 3.2.2 Perancangan Perangkat Lunak

Perancangan perangkat lunak dilakukan menggunakan *software* Visual Studio Code. Visual Studio Code digunakan untuk mengolah sinyal EOG menggunakan metode *random forest* dan memprogram keseluruhan sistem kursi roda. Pengoperasian kursi roda untuk maju, mundur, belok kiri, belok kanan, dan ON/OFF dilakukan oleh aktivitas mata yang dideteksi menggunakan sensor *electrooculography*. Sinyal *electrooculography* yang telah didapatkan akan diolah untuk mendapatkan fitur sinyalnya. Fitur sinyal EOG yang digunakan pada perancangan ini, yaitu fitur polaritas dan puncak sinyal.

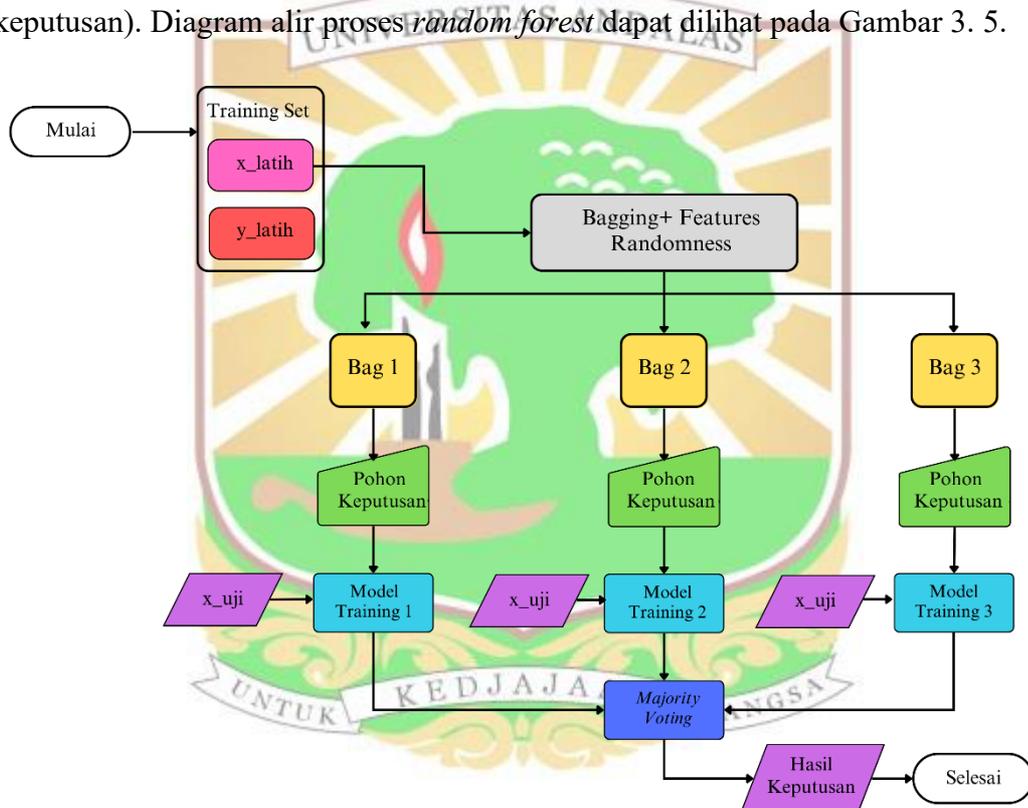
Fitur polaritas sinyal digunakan untuk mendeteksi aktivitas lirik ke bawah, ke kanan, dan ke kiri karena sinyal dari ketiga jenis lirik tersebut dapat dibedakan langsung polaritasnya. Hasil dari fitur polaritas sinyal ini digunakan sebagai input untuk arah gerak mundur, belok kanan, dan belok kiri pada kursi roda. Sedangkan untuk lirik atas, kedip sadar, dan kedip tidak sadar pada mata digunakan ekstraksi fitur puncak sinyal untuk mengklasifikasikannya menggunakan metode *random forest*. Hasil klasifikasi ini digunakan sebagai input untuk arah gerak maju, sistem ON/OFF, dan kondisi diam pada kursi roda. Khusus setelah terdeteksi sinyal gerakan lirik atas, lirik bawah, lirik kiri, dan lirik kanan maka akan diberikan *locking time*. Untuk lebih jelasnya, rancangan diagram alir sistem perangkat lunak pada kursi roda ini, dapat dilihat pada Gambar 3.4.



Gambar 3. 4 Diagram Alir Sistem Perangkat Lunak

### 3.2.2.1 Perancangan Klasifikasi dengan Random Forest

*Random Forest* merupakan implementasi dari *homogeneous ensemble learning*, yaitu pembelajaran *ensemble* yang menggunakan model dasar yang sama, dalam hal ini *Decision Tree*. Selain menerapkan teknik *bagging* (*bootstrap aggregating*), *Random Forest* juga mengadopsi *feature randomness*, di mana setiap model *Decision Tree* dilatih menggunakan subset data latih dan subset fitur yang dipilih secara acak. Pada setiap *bag* di latih menggunakan model yang sama yaitu *Decision Tree* sehingga menghasilkan model terlatih dengan hasil yang berbeda beda dikarenakan dataset yang berbeda walau menggunakan model yang sama. Hal ini menghasilkan variasi antar model, yang menjadi kekuatan utama dalam meningkatkan kemampuan generalisasi. Selanjutnya prediksi tiap model akan di satukan menggunakan *majority voting* untuk mendapatkan nilai prediksi (hasil keputusan). Diagram alir proses *random forest* dapat dilihat pada Gambar 3. 5.

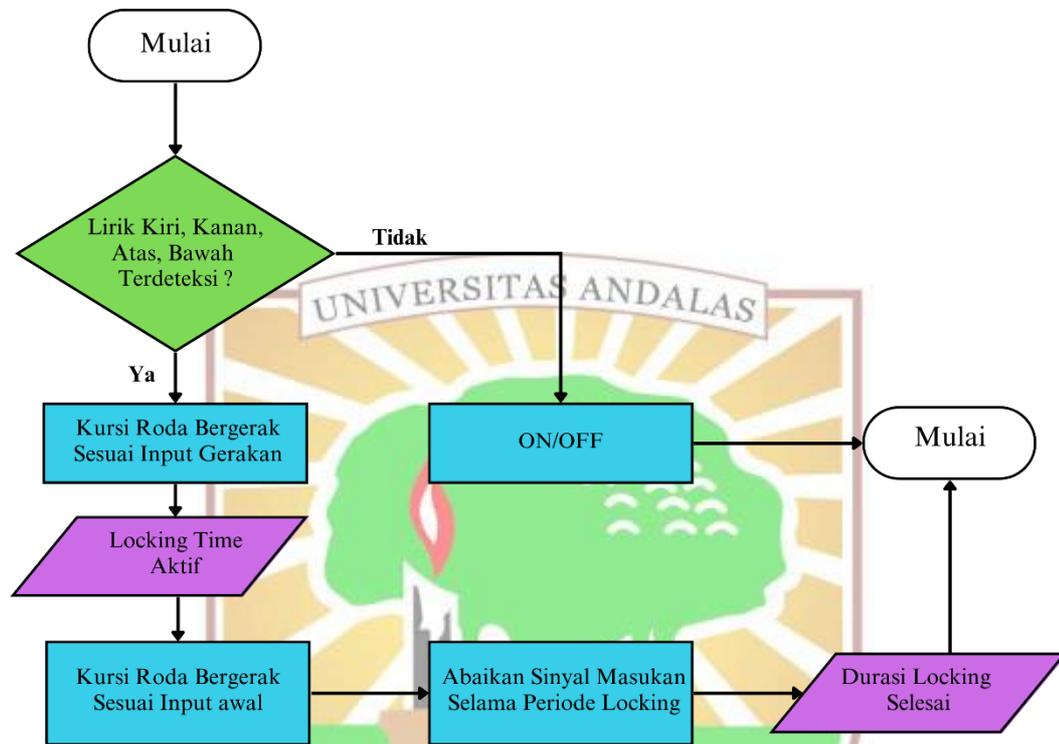


Gambar 3. 5 Diagram Alir Proses *Random Forest*

### 3.2.2.2 Perancangan Locking Time Signal

Pada pengendalian kursi roda ini diperlukan perancangan *locking time* untuk meningkatkan ketepatan sistem dalam mengenali perintah yang valid serta meminimalkan risiko kesalahan akibat sinyal yang tidak diinginkan. *Locking time* adalah periode waktu tertentu setelah sistem menerima sinyal perintah, di mana seluruh sinyal masukan selanjutnya akan diabaikan selama periode waktu *locking* yang diberikan. Fungsi utama dari *locking time* adalah untuk mencegah sistem salah mengartikan sinyal-sinyal tidak diinginkan yang muncul setelah perintah utama,

seperti sinyal refleks, gerakan tidak disengaja, maupun sinyal balik (*rebound signal*) akibat aktivitas otot mata. Tanpa *locking time*, sinyal tersebut dapat dikenali sebagai perintah baru sehingga berpotensi menyebabkan pergerakan kursi roda yang tidak sesuai keinginan pengguna. Diagram alir proses kerja *locking time* dapat dilihat pada Gambar 3.6.



**Gambar 3. 6** Diagram Alir Proses Kerja *Locking Time*

Gambar 3.6 menampilkan diagram alir proses kerja fitur *locking time* pada sistem. Sistem terlebih dahulu mendeteksi jenis input yang masuk. Apabila input yang masuk yaitu kedip sadar, dan kedip tidak sadar, maka perintah akan dijalankan tanpa mengaktifkan fitur *locking time*. Fitur *locking time* hanya aktif ketika sistem mendeteksi gerakan lirikan (kiri, kanan, atas, atau bawah). Setelah sistem mendeteksi arah lirikan, kursi roda akan bergerak sesuai dengan arah tersebut. Untuk menghindari respon ganda atau input yang tidak stabil, sistem mengaktifkan fitur *locking time*, yaitu periode waktu tertentu di mana semua input tambahan diabaikan selama durasi *locking*. Setelah durasi *locking* selesai, sistem kembali ke kondisi awal untuk menerima input baru.

### 3.3 Rancangan Pengujian

Rancangan percobaan dilakukan untuk menguji kinerja sistem kendali kursi roda berbasis sinyal *electrooculography* (EOG) yang telah dirancang, baik dari sisi akurasi klasifikasi sinyal maupun keakuratan pergerakan kursi roda sesuai perintah. Percobaan ini juga bertujuan mengevaluasi keefektifan metode *random forest* dalam mengklasifikasikan sinyal EOG serta pengaruh penerapan *locking time*

terhadap kestabilan sistem. Terdapat 3 pengujian yang dilakukan menggunakan beberapa responden. Adapun pengujian-pengujian tersebut adalah sebagai berikut.

### 1. Pengujian Klasifikasi Sinyal EOG dengan Metode *Random Forest*

Pengujian ini bertujuan untuk mengevaluasi kemampuan sistem dalam mengklasifikasikan sinyal EOG, meliputi lirikan atas, kedip sadar, dan kedip tidak sadar. Data yang digunakan berasal dari dataset sinyal EOG yang dibagi menjadi data latih dan data uji. Model *random forest* yang telah dilatih akan dievaluasi menggunakan data uji untuk mengetahui tingkat akurasi dari model *random forest*. Adapun parameter utama yang digunakan dalam pengembangan model *random forest* meliputi jumlah pohon keputusan (*number of trees*) dan kedalaman maksimal pohon (*maximum depth*).

Tabel 3. 1 Confusion Matrix

Kelas Asli	Kelas Prediksi		TR (%)	FR (%)
	1	2		
1	TP	FN	TP/TP+FN	FN/TP+FN
2	FP	TN	TN/FP+TN	FP/FP+TN
Akurasi (%)			TP+TN/TP+FN+FP+TN	

Pengujian akurasi dilakukan menggunakan *confusion matrix* yang dapat dilihat pada Tabel 3.1. *Confusion matrix* digunakan untuk menilai seberapa baik sebuah model dapat memprediksi target dari sebuah data fitur. Pada *confusion matrix* terdapat beberapa istilah yang digunakan. TP (*True Positive*) merupakan jumlah sampel dari kelas 1 yang benar diprediksi sebagai kelas 1. FN (*False Negative*) merupakan jumlah sampel dari kelas 1 yang salah diprediksi sebagai kelas 2. FP (*False Positive*) merupakan jumlah sampel dari kelas 2 yang salah diprediksi sebagai kelas 1. TN (*True Negative*) merupakan jumlah sampel dari kelas 2 yang benar diprediksi sebagai kelas 2. TR (*True Rate*) merupakan rasio dari jumlah sampel suatu kelas yang diprediksi dengan benar dibagi dengan jumlah sample prediksi suatu kelas. FR (*False Rate*) merupakan rasio dari jumlah sampel suatu kelas yang diprediksi secara salah dibagi dengan jumlah sampel prediksi suatu kelas. Akurasi merupakan perbandingan dari jumlah prediksi yang benar dengan total jumlah sampel prediksi. Berikut merupakan langkah-langkah pengujian klasifikasi sinyal EOG dengan metode *Random Forest* :

1. Preprocessing data dan ekstraksi fitur puncak sinyal (CH1 MAX, CH1 MIN, CH2 MAX, CH2 MIN).
2. Pemisahan Data Latih dan Data Uji
3. Pelatihan Model Random Forest

Model *random forest* dilatih menggunakan data latih yang terdiri atas fitur nilai puncak sebagai input, serta kelas aktivitas mata sebagai target. Proses pelatihan dilakukan dengan variasi parameter yang ditentukan.

4. Prediksi dan Klasifikasi Data Uji

## 5. Evaluasi Model Menggunakan Confusion Matrix

Hasil klasifikasi dibandingkan dengan kelas sebenarnya untuk membentuk confusion matrix yang ditampilkan seperti dalam Tabel 3.1. Evaluasi dilakukan untuk mengetahui performa model dalam mengenali sinyal EOG gerakan mata.

## 2. Pengujian Pengaturan Durasi *Locking Time*

Pengujian ini bertujuan untuk mengevaluasi efektivitas penerapan *locking time* dalam mencegah kesalahan deteksi akibat sinyal refleks, gerakan tidak disengaja, maupun sinyal balik (*rebound signal*) yang muncul setelah perintah utama dilakukan. Durasi *locking time* yang telah dirancang diuji melalui serangkaian percobaan untuk memastikan bahwa sistem hanya merespons perintah yang sah dan mengabaikan sinyal-sinyal yang tidak diinginkan. Berikut merupakan langkah-langkah pengujian durasi *locking time* :

1. Sistem diaktifkan dan menerima perintah utama dari sinyal EOG.
2. Setelah perintah dijalankan, *locking time* diaktifkan.
3. Berikan sinyal gangguan selama periode *locking time*.
4. Analisis apakah sistem mengabaikan sinyal gangguan dan tetap menjalankan perintah awal.
5. Ulangi pengujian dengan beberapa variasi durasi *locking time* untuk setiap gerakan mata.
6. Evaluasi efektivitas setiap durasi dalam mengabaikan sinyal non-perintah (sinyal balik dan gerakan refleks mata)

## 3. Pengujian Kinerja Pergerakan Kursi Roda

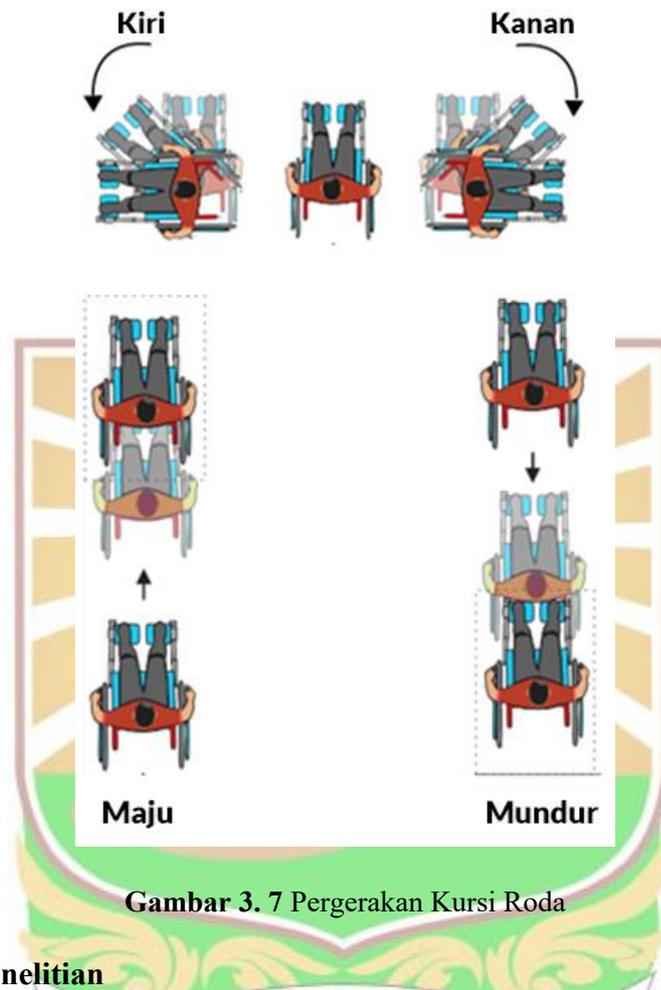
Pada pengujian ini, sistem diuji secara keseluruhan untuk memastikan pergerakan kursi roda sesuai dengan perintah yang diberikan melalui aktivitas mata *responden*. Pengujian mencakup arah gerak maju, mundur, belok kanan, belok kiri, berhenti, serta ON/OFF sistem. Evaluasi dilakukan berdasarkan tingkat keberhasilan sistem dalam mengeksekusi perintah secara tepat. Berikut merupakan langkah-langkah pengujian kinerja kursi roda :

1. Aktifkan sistem kursi roda melalui perintah ON dari sinyal EOG (kedip sadar).
2. Berikan perintah gerakan satu per satu: maju, mundur, belok kanan, belok kiri, berhenti.
3. Amati respons kursi roda terhadap setiap perintah yang diberikan.
4. Catat apakah setiap perintah berhasil dijalankan dengan benar atau terjadi kesalahan respon.

### 3.4 Rancangan Operasi Kursi Roda

Pada penelitian ini, kursi roda dioperasikan menggunakan aktivitas mata yaitu, lirikan dan kedipan. Aktivitas mata berupa lirikan terdiri dari lirikan mata ke kanan, kiri, atas, dan bawah, sedangkan untuk aktivitas mata berupa kedipan terdiri dari kedip sadar dan kedip tidak sadar. Lirikan mata ke kanan akan mengendalikan kursi roda untuk bergerak ke kanan, lirikan mata ke kiri akan mengendalikan kursi roda untuk bergerak ke kiri, lirikan mata ke atas akan mengendalikan kursi roda

untuk bergerak maju, lirikan mata ke bawah akan mengendalikan kursi roda untuk bergerak mundur, kedip tidak sadar menginstruksikan diam (*no movement*), serta kedipan sadar akan mengaktifkan atau menonaktifkan kursi roda. Pergerakan kursi roda yang dirancang dapat dilihat pada Gambar 3.7.



Gambar 3. 7 Pergerakan Kursi Roda

### 3.4 Jadwal Penelitian

Penelitian tugas akhir dilaksanakan di gedung Departemen Teknik Elektro Universitas Andalas, dengan jadwal penelitian yang dapat dilihat pada tabel 3.2 berikut ini :

Tabel 3. 2 Jadwal Penelitian

No	Kegiatan	April				Mei				Juni				Juli			
		1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1	Studi Literatur	█															
2	Perancangan Sistem					█											
3	Pengambilan Data									█							
4	Pengolahan Data													█			
5	Pembuatan program													█			
5	Klasifikasi di VS Code													█			

No	Kegiatan	April				Mei				Juni				Juli			
		1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
6	Pengujian dan Evaluasi Sistem																
7	Analisis Hasil Akhir																

### 3.5 Analisis Sistem

Setelah dilakukan pengujian terhadap sistem yang dibuat, maka hasil pengujian tersebut akan dianalisis. Adapun beberapa poin yang akan dianalisis dalam penelitian ini adalah sebagai berikut:

#### 1. Analisis Ketepatan klasifikasi sinyal EOG dengan metode *Random Forest*

Analisis ini difokuskan pada kinerja model *random forest* dalam mengklasifikasikan sinyal EOG berdasarkan tiga jenis aktivitas mata, yaitu lirikan atas, kedip sadar, dan kedip tidak sadar. Hasil klasifikasi dievaluasi menggunakan parameter akurasi berdasarkan *confusion matrix* yang dihasilkan.

#### 2. Analisis Pengaturan Durasi *Locking Time*

Analisis ini dilakukan dengan membandingkan beberapa durasi *locking time* untuk menentukan nilai optimal yang mampu meminimalkan kesalahan deteksi tanpa mengurangi kecepatan respons sistem. Tujuan utamanya adalah memperoleh pengaturan *locking time* yang efektif dalam mencegah salah interpretasi sinyal.

#### 3. Analisis Ketepatan Pergerakan kursi roda

Analisis ini difokuskan untuk ketepatan kursi roda dalam bergerak maju, mundur, belok kiri, belok kanan, berhenti, serta ON/OFF berdasarkan sinyal lirikan dan kedipan mata. Evaluasi dilakukan dengan membandingkan jumlah perintah yang diberikan dengan jumlah perintah yang berhasil dijalankan oleh sistem secara tepat. Persentase keberhasilan menjadi indikator utama dalam menentukan apakah sistem dapat merespons perintah pengguna secara akurat.

## BAB IV HASIL DAN PEMBAHASAN

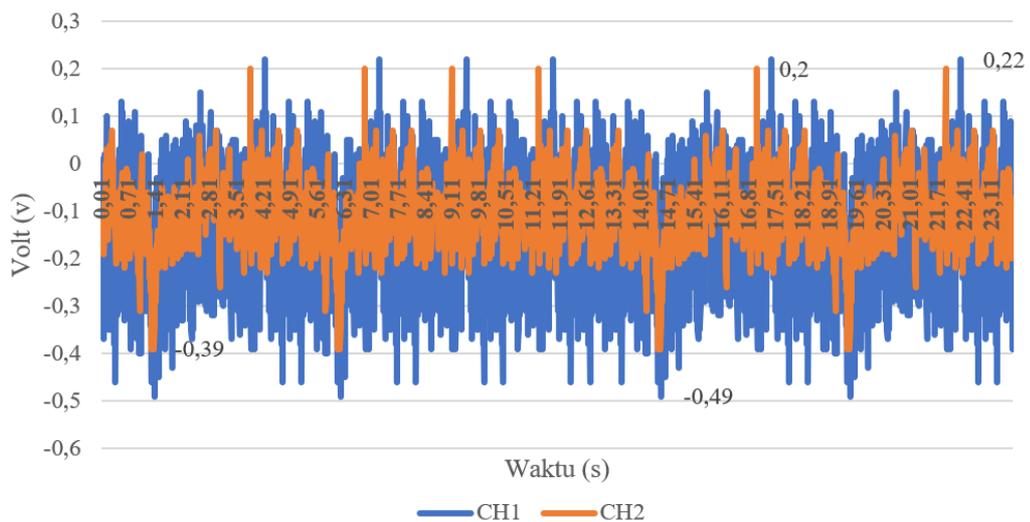
### 4.1 Penetapan Nilai Threshold

Untuk membedakan antara polaritas positif dan negatif pada sinyal EOG, diperlukan penetapan nilai ambang batas atau *threshold*. Nilai *threshold* ini berfungsi sebagai batas yang digunakan untuk memisahkan sinyal polaritas positif (+) dan negatif (-) secara jelas. Penentuan *threshold* dilakukan saat mata berada dalam kondisi diam dan memandang lurus ke depan, sehingga sinyal yang dihasilkan menunjukkan keadaan netral tanpa adanya pergerakan mata.

Dalam proses ini, digunakan dua jenis *threshold*, yaitu *threshold* positif dan *threshold* negatif. *Threshold* positif diperoleh dengan mengambil dan membulatkan ke atas nilai maksimum sinyal saat mata tidak bergerak, sedangkan *threshold* negatif diperoleh dengan membulatkan ke bawah nilai minimum sinyal dalam kondisi yang sama. Dengan adanya kedua nilai *threshold* ini, sistem dapat lebih mudah mendeteksi perubahan polaritas sinyal EOG saat terjadi pergerakan mata.

Perekaman sinyal EOG saat mata dalam kondisi diam dilakukan dengan cara menginstruksikan responden untuk tetap fokus memandang lurus ke depan tanpa melakukan gerakan bola mata maupun kedipan selama proses akuisisi data berlangsung. Langkah ini bertujuan untuk menjaga kestabilan sinyal dan memastikan data yang diperoleh benar-benar merepresentasikan kondisi mata dalam keadaan diam tanpa aktivitas tambahan. Dari hasil perekaman, diperoleh sebanyak 2400 sampel data sinyal diam. Data tersebut digunakan sebagai dasar dalam menentukan nilai *threshold* positif dan negatif. Seluruh data hasil perekaman dapat dilihat secara lengkap pada Lampiran A.

Penentuan nilai *threshold* dilakukan dengan mengambil nilai maksimum dari keseluruhan data sinyal diam. Pendekatan ini digunakan untuk memberikan jarak yang cukup antara titik nol sinyal dengan batas *threshold*, sehingga sistem dapat beroperasi lebih stabil dan tidak terlalu sensitif terhadap fluktuasi kecil akibat *noise* atau perubahan sinyal minor. Contoh visualisasi bentuk sinyal diam yang telah direkam dapat dilihat pada Gambar 4.1.



**Gambar 4. 1** Grafik Sinyal Diam

**Tabel 4. 1** Analisis Nilai *Threshold*

Nilai Sinyal	CH1 (Volt)	CH2 (Volt)
Maksimum	0,22	0,22
Minimum	-0,49	-0,39

Berdasarkan Tabel 4.1, diperoleh bahwa nilai maksimum sinyal saat mata dalam kondisi diam pada kedua channel adalah sebesar 0,22 volt, sedangkan nilai minimumnya sebesar -0,49 volt pada CH1 dan -0,39 pada CH2. Berdasarkan rentang tersebut, ditetapkan nilai *threshold* sebesar 0,3 volt untuk *threshold* positif dan -0,5 volt untuk *threshold* negatif. Setelah *threshold* ditentukan, maka dapat ditentukan polaritas dan puncak dari masing-masing jenis sinyal pergerakan mata. Jika nilai sinyal melebihi *threshold* positif (0,3 Volt), maka sinyal tersebut dikategorikan memiliki polaritas positif, dan nilai tertingginya disebut sebagai puncak positif. Sebaliknya, jika nilai sinyal lebih kecil dari *threshold* negatif (-0,5 Volt), maka sinyal tersebut memiliki polaritas negatif, dan nilai terendahnya disebut sebagai puncak negatif.

#### 4.2 Pengambilan Data Sinyal Pergerakan Mata

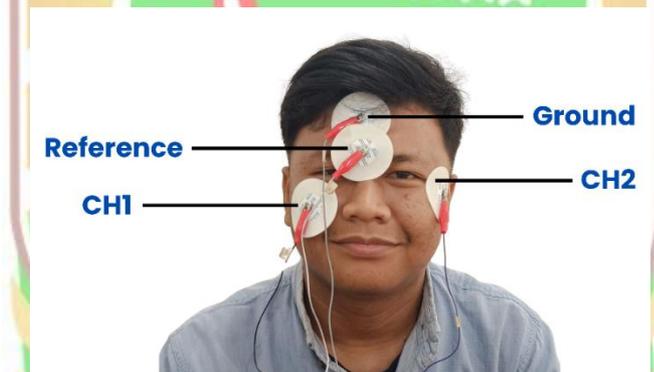
Pengambilan data dilakukan untuk masing-masing jenis pergerakan mata yang dibutuhkan dalam sistem kendali kursi roda berbasis sinyal Electrooculography (EOG). Kriteria pemilihan responden adalah mahasiswa Universitas Andalas yang tidak memiliki kelainan atau gangguan pada mata. Total terdapat 10 responden yang terdiri dari 9 laki-laki dan 1 perempuan.

Setiap responden diminta untuk melakukan setiap jenis pergerakan mata, yaitu kedip sadar, kedip tidak sadar, lirik kanan, lirik kiri, lirik atas, dan lirik bawah untuk setiap jenis gerakan. Adapun gerakan lirik kiri, kanan, dan bawah dilakukan

satu sesi perekaman untuk masing-masing jenis, tanpa pengulangan. Khusus untuk gerakan kedip sadar, kedip tidak sadar, dan lirik atas, dilakukan pengulangan tiga kali karena data dari jenis gerakan ini akan digunakan untuk pelatihan model klasifikasi Random Forest. Sinyal EOG direkam menggunakan perangkat akuisisi data dengan frekuensi sampling 100 Hz (satu sampel setiap 0,01 detik).

Untuk data gerakan kedip sadar, kedip tidak sadar, dan lirik atas, dilakukan ekstraksi fitur nilai puncak berupa CH1 Max, CH1 Min, CH2 Max, dan CH2 Min. Ekstraksi ini bertujuan untuk memperoleh representasi karakteristik sinyal antar jenis pergerakan. Untuk lirik kiri, kanan, dan bawah, data tidak diekstraksi dalam bentuk fitur puncak. Klasifikasi dilakukan berdasarkan perbedaan polaritas (positif/negatif) dari channel CH1 dan CH2 yang merupakan karakteristik unik dari arah gerakan horizontal dan vertikal mata.

Tujuan dari pengambilan data ini adalah untuk mengamati karakteristik bentuk sinyal yang dihasilkan oleh masing-masing jenis pergerakan mata. Setelah itu, dapat ditentukan parameter fitur yang tepat untuk menentukan jenis sinyal dari masing-masing jenis pergerakan mata yang dibutuhkan. Adapun rancangan penempatan elektroda pada responden untuk pengambilan data dapat dilihat pada gambar 4.2 berikut.

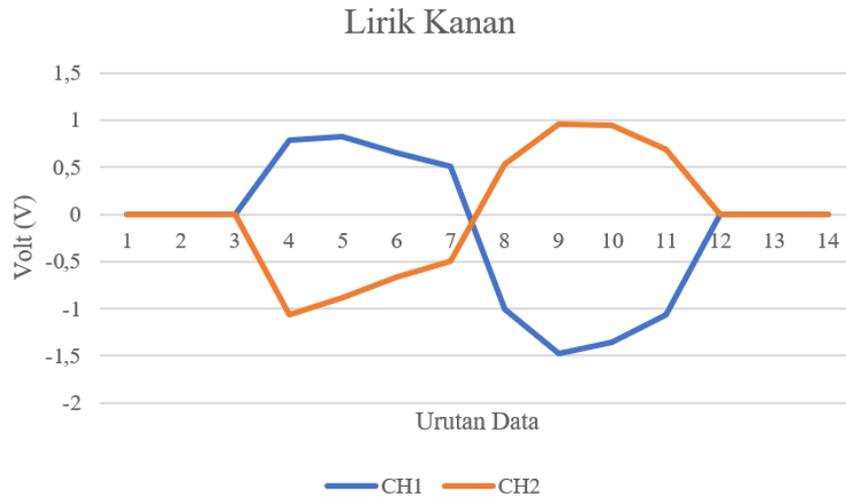


**Gambar 4. 2** Posisi Peletakan Elektroda Pengambilan Data Sinyal EOG

#### 4.2.1 Data Sinyal Lirik Kanan

Pengambilan data awal dilakukan dengan merekam sinyal yang dihasilkan dari pergerakan mata ke arah kanan (lirik kanan). Proses perekaman dimulai saat subjek berada dalam kondisi melihat lurus ke depan, di mana posisi kornea berada di tengah, kemudian melakukan pergerakan ke arah kanan, dan diakhiri dengan kembalinya pandangan ke posisi awal (kornea kembali ke tengah). Hasil perekaman menunjukkan bahwa bentuk sinyal lirik kanan dari kelima responden memiliki pola yang serupa, sebagaimana ditampilkan pada Gambar 4.3. Berdasarkan grafik tersebut, pergerakan lirik kanan ditandai dengan sinyal pada CH1 yang melewati *threshold* positif dan sinyal pada CH2 yang melewati *threshold* negatif, yang mengindikasikan perpindahan kornea dari posisi tengah ke kanan. Setelah itu, sinyal kembali berada dalam rentang *threshold*, yang menunjukkan kornea berada tetap di posisi kanan. Selanjutnya, sinyal CH1 melewati *threshold* negatif dan

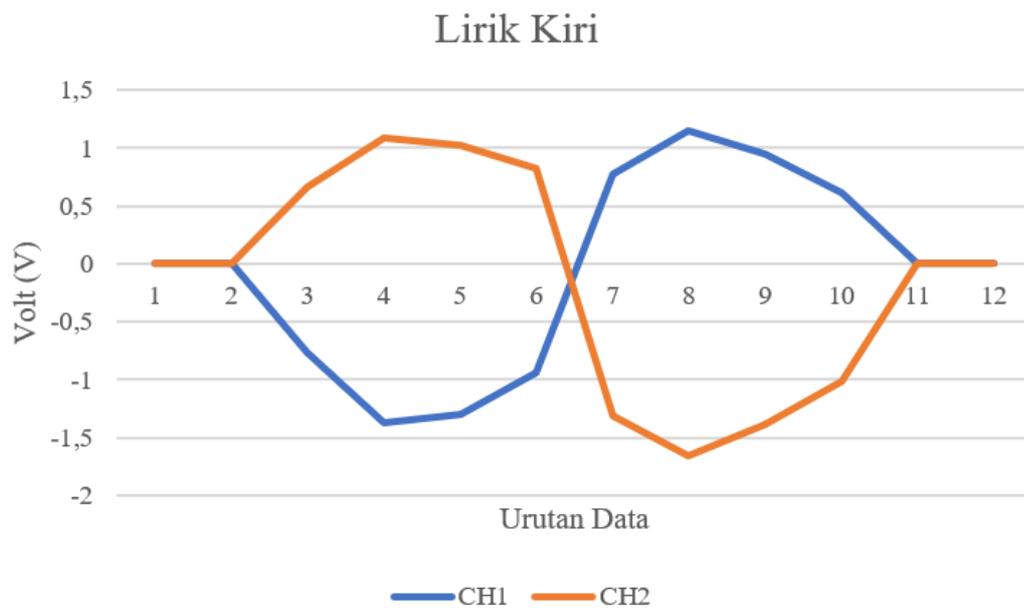
sinyal CH2 melewati *threshold* positif, yang merepresentasikan kembalinya kornea dari posisi kanan ke tengah. Pergerakan ini berakhir saat sinyal kembali berada dalam rentang *threshold*, menandakan bahwa kornea telah stabil di posisi tengah.



**Gambar 4.3** Sinyal Lirik Kanan

#### 4.2.2 Data Sinyal Lirik Kiri

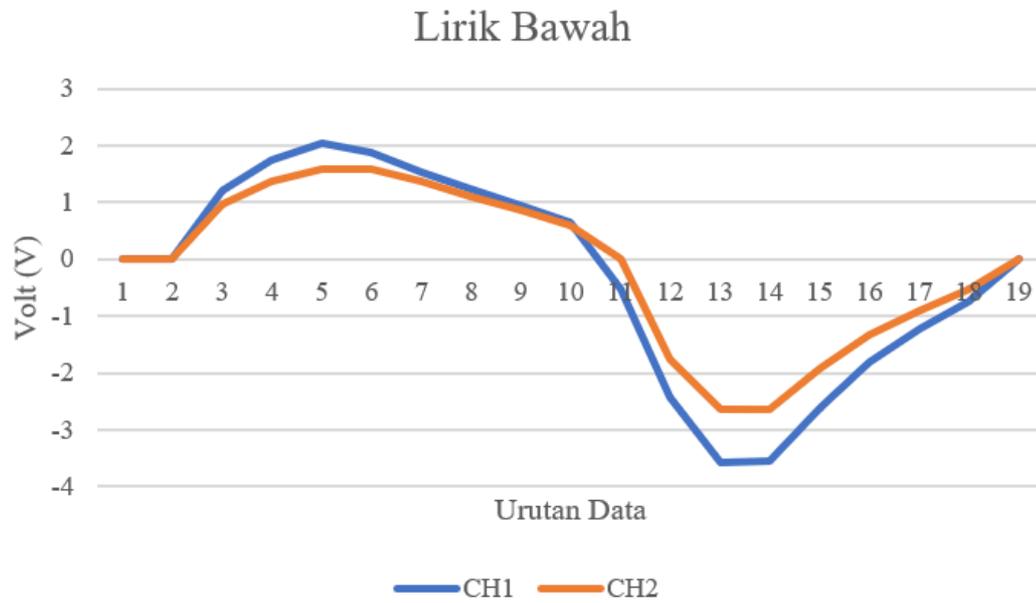
Pengambilan data untuk gerakan lirik mata ke kiri dilakukan dengan mengarahkan responden agar menggerakkan mata ke arah kiri. Proses ini dimulai ketika mata berada dalam posisi netral, yaitu melihat lurus ke depan (kornea berada di tengah). Selanjutnya, responden menggerakkan mata ke arah kiri (kornea bergeser dari posisi tengah ke sisi kiri), dan kemudian mengembalikan pandangan ke posisi semula. Berdasarkan Gambar 4.4, dapat diketahui bahwa awal sinyal lirik kiri terjadi saat sinyal pada kanal CH2 melewati *threshold* positif dan sinyal CH1 melewati *threshold* negatif, menandakan pergerakan kornea dari tengah ke kiri. Setelah itu, sinyal tetap berada di luar *threshold* (mengindikasikan posisi kornea masih di sisi kiri). Saat kornea bergerak kembali ke tengah, sinyal CH2 melewati *threshold* negatif dan CH1 melewati *threshold* positif. Sinyal dianggap berakhir ketika kedua channel kembali berada dalam rentang *threshold*, menunjukkan bahwa kornea telah kembali ke posisi netral



**Gambar 4. 4 Sinyal Lirik Kiri**

#### 4.2.3 Data Sinyal Lirik Bawah

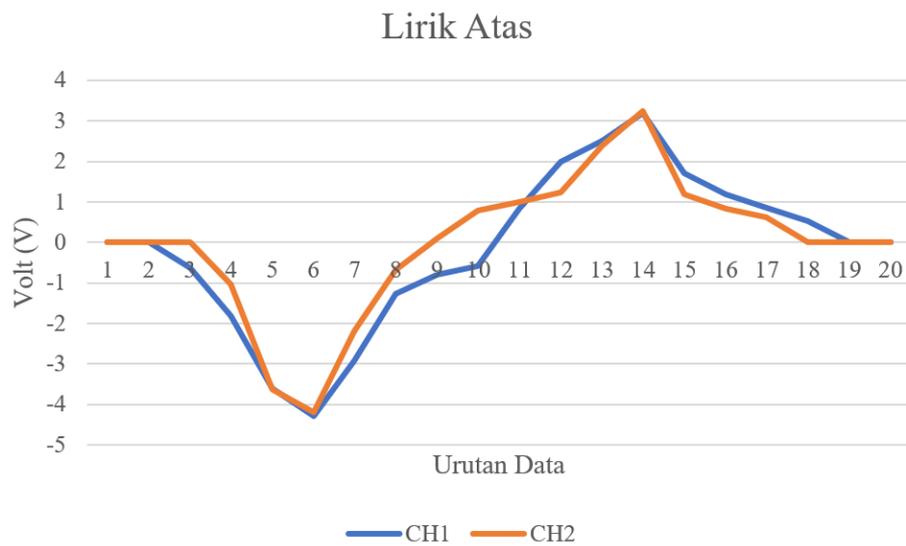
Pengambilan data untuk gerakan lirik bawah dilakukan dengan mengarahkan responden agar melakukan gerakan melirik ke bawah. Gerakan ini diawali ketika mata berada dalam posisi menatap lurus ke depan (kornea terletak di tengah), lalu dilanjutkan dengan pergerakan mata ke arah bawah (kornea berpindah dari posisi tengah ke bawah). Pergerakan ini selesai ketika mata kembali menatap lurus (kornea bergerak kembali dari bawah ke posisi tengah). Berdasarkan Gambar 4.5, sinyal lirikan ke bawah mulai terlihat ketika CH1 dan CH2 melewati *threshold* positif, yang menandai pergerakan kornea ke bawah. Selanjutnya, nilai sinyal tetap berada di atas *threshold* ini selama kornea berada di posisi bawah. Setelah itu, sinyal pada CH1 dan CH2 melewati *threshold* negatif sebagai tanda kornea mulai bergerak kembali ke atas, dan sinyal dianggap berakhir ketika kedua channel menunjukkan nilai di bawah *threshold*, yang menandakan kornea telah kembali ke posisi tengah.



**Gambar 4. 5 Sinyal Lirik Bawah**

#### 4.2.4 Data Sinyal Lirik Atas

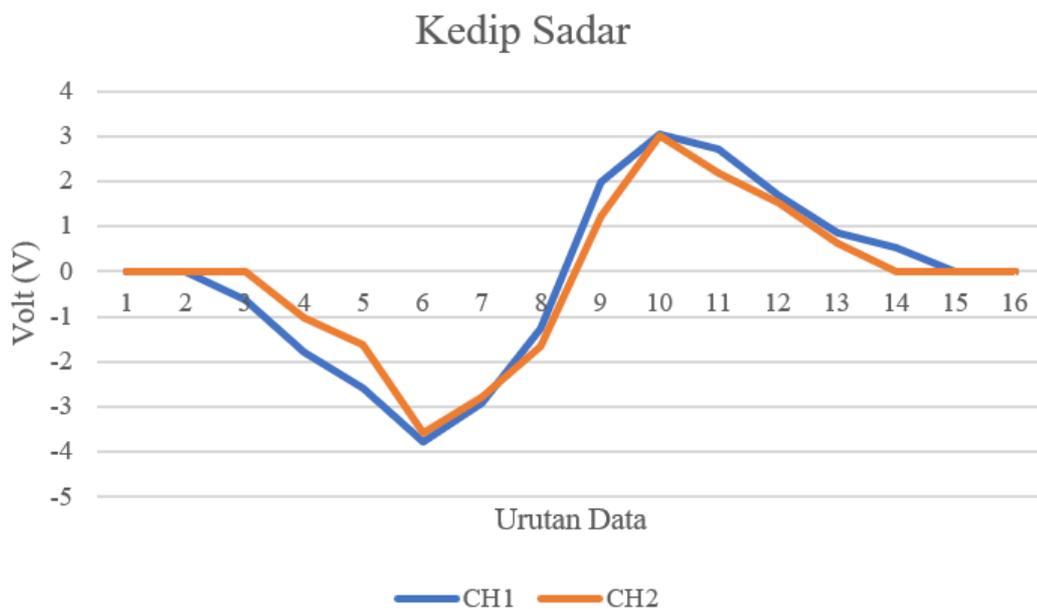
Pengambilan data untuk pergerakan lirik atas dilakukan dengan merekam sinyal saat mata berpindah dari posisi netral ke atas, lalu kembali ke posisi semula. Hasil perekaman menunjukkan pola sinyal lirik atas seperti ditunjukkan pada Gambar 4.6. Berdasarkan gambar 4.6, pergerakan dimulai saat sinyal CH1 dan CH2 melewati ambang negatif (kornea bergerak ke atas), kemudian sinyal kembali dalam rentang *threshold* (kornea tetap di atas). Kemudian, dilanjutkan dengan sinyal melewati ambang positif saat mata kembali ke tengah, dan diakhiri ketika sinyal stabil dalam rentang *threshold* menandakan posisi netral.



**Gambar 4. 6 Sinyal Lirik Atas**

#### 4.2.5 Data Sinyal Kedip Sadar

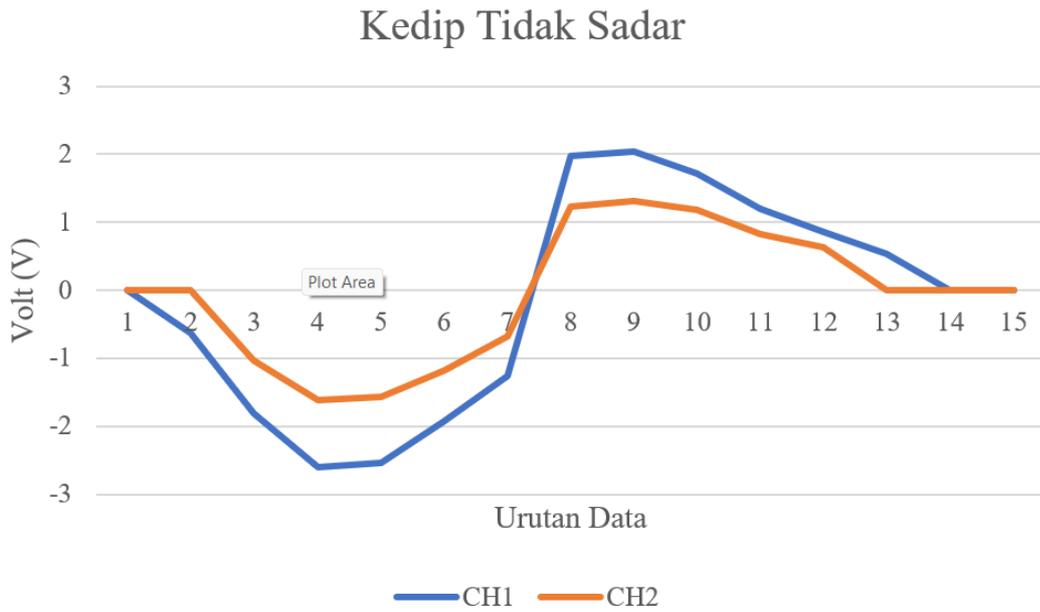
Pengambilan data untuk pergerakan kedip sadar dilakukan dengan merekam sinyal saat mata berpindah dari posisi netral, melakukan satu kali kedip sadar (kedipan ditekan), lalu kembali terbuka ke posisi semula. Hasil perekaman menunjukkan pola sinyal kedip sadar sebagaimana ditampilkan pada Gambar 4.7. Berdasarkan pada grafik gambar 4.7 tersebut, pergerakan diawali dengan sinyal CH1 dan CH2 melewati ambang negatif (awal kedip sadar). Kemudian, dilanjutkan dengan sinyal melewati ambang positif (akhir kedip sadar), dan diakhiri saat sinyal kembali dalam rentang *threshold*, menandakan mata telah terbuka dan kornea kembali di posisi tengah.



Gambar 4. 7 Sinyal Kedip Sadar

#### 4.2.6 Data Sinyal Kedip Tidak Sadar

Pengambilan data untuk pergerakan kedip tidak sadar dilakukan dengan merekam sinyal saat mata berpindah dari posisi netral, melakukan satu kali kedip tidak sadar (kedipan tanpa tekanan), lalu kembali terbuka ke posisi semula. Hasil perekaman menunjukkan bentuk grafik sinyal sebagaimana ditampilkan pada Gambar 4.8. Berdasarkan grafik pada gambar 4.8, pergerakan diawali dengan sinyal CH1 dan CH2 melewati ambang negatif (awal kedip tidak sadar). Kemudian dilanjutkan dengan sinyal melewati ambang positif (akhir kedip tidak sadar), dan diakhiri saat sinyal kembali dalam rentang *threshold*, menandakan mata telah terbuka dan kornea kembali di posisi tengah.



**Gambar 4. 8** Sinyal Lirik Tidak Sadar

#### 4.2.7 Analisis Fitur Sinyal Pergerakan Mata

Berdasarkan bentuk sinyal dari enam jenis pergerakan mata, maka dapat ditentukan fitur sinyal untuk membedakan jenis masing-masing sinyal. Beberapa jenis sinyal dapat dibedakan langsung menggunakan fitur polaritasnya. Penggunaan fitur polaritas untuk membedakan jenis sinyal dapat dilihat pada Tabel 4.2.

**Tabel 4. 2** Fitur Polaritas Berdasarkan Jenis Sinyal Pergerakan Mata

Jenis Pergerakan Mata	Channel EOG	Polaritas Sinyal Awal	Polaritas Sinyal Akhir
Lirik Kanan	CH1	+	-
	CH2	-	+
Lirik Kiri	CH1	-	+
	CH2	+	-
Lirik Bawah	CH1	+	-
	CH2	+	-
Lirik Atas	CH1	-	+
	CH2	-	+
Kedip Sadar	CH1	-	+
	CH2	-	+
Kedip Tidak Sadar	CH1	-	+
	CH2	-	+

Berdasarkan Tabel 4.2, terlihat bahwa setiap jenis sinyal memiliki dua bentuk polaritas, yakni polaritas pada awal dan akhir sinyal. Kedua polaritas ini

menunjukkan nilai yang saling berlawanan; jika polaritas awal bernilai positif, maka polaritas akhirnya akan bernilai negatif, dan begitu pula sebaliknya. Hal ini disebabkan karena polaritas awal mencerminkan arah gerakan pertama, sedangkan polaritas akhir menggambarkan gerakan mata yang kembali ke posisi awal. Pada tabel tersebut juga terlihat bahwa sinyal lirik ke kiri, kanan, dan bawah memiliki perbedaan polaritas yang jelas antara awal dan akhirnya, sehingga ketiganya dapat dikenali langsung melalui fitur polaritas ini. Sementara itu, untuk sinyal lirik atas, kedip sadar, dan kedip tidak sadar, ditemukan kesamaan polaritas antara sinyal awal dan akhir, sehingga tidak cukup hanya menggunakan polaritas sebagai pembeda. Oleh karena itu, diperlukan fitur tambahan berupa nilai puncak sinyal, yang penggunaannya untuk membedakan ketiga sinyal tersebut dijelaskan pada Tabel 4.3.

**Tabel 4.3** Fitur Puncak Sinyal Berdasarkan Jenis Sinyal Pergerakan Mata

Jenis Pergerakan Mata	Puncak Positif		Puncak Negatif	
	Max CH1	Max CH2	Min CH1	Min CH2
Kedip Sadar	3.08	3.18	-3.77	-3.7
Kedip Tidak Sadar	1.5	1	-2	-2.1
Lirik Atas	3.32	3.24	-4.28	-4.2

Berdasarkan Tabel 4.3, dapat diketahui bahwa sinyal lirik ke atas, kedipan sadar, dan kedipan tidak sadar memiliki variasi nilai puncak positif dan negatif yang berbeda satu sama lain. Perbedaan ini memungkinkan penggunaan fitur nilai puncak sebagai dasar untuk mengidentifikasi masing-masing jenis sinyal tersebut. Namun demikian, nilai puncak yang dijadikan acuan tidak dapat ditetapkan secara langsung karena adanya variasi antar individu. Dengan kata lain, setiap orang dapat menghasilkan nilai puncak yang berbeda untuk jenis sinyal yang sama. Oleh sebab itu, diperlukan kumpulan data yang luas mengenai nilai puncak dari ketiga jenis sinyal ini yang berasal dari banyak individu, serta suatu algoritma yang mampu mengenali pola-pola tersebut. Dalam penelitian ini, digunakan algoritma *random forest* sebagai metode untuk mengklasifikasikan sinyal kedip sadar, kedip tidak sadar, dan lirik atas berdasarkan karakteristik nilai puncaknya.

### 4.3 Hasil Perancangan Model *Random Forest*

Model *Random Forest* digunakan untuk mengklasifikasikan sinyal-sinyal yang memiliki nilai polaritas sama, yakni kedip sadar, kedip tidak sadar, dan lirik atas yang ditandai dengan polaritas negatif pada CH1 dan CH2. Klasifikasi dilakukan dengan menggunakan fitur puncak sinyal, yang terdiri dari nilai puncak positif CH1 (CH1 Max), nilai puncak negatif CH1 (CH1 Min), nilai puncak positif CH2 (CH2 Max), nilai puncak negatif CH2 (CH2 Min). Masing-masing jenis sinyal diberi label kelas numerik untuk memudahkan proses pelatihan model dan evaluasi hasil klasifikasi. Label kelas untuk setiap jenis sinyal dapat dilihat pada Tabel 4.4.

**Tabel 4. 4** Data Kelas

No	Jenis Sinyal	Kelas
1	Kedip Sadar	1
2	Kedip Tidak Sadar	2
3	Lirik Atas	3

Jumlah data yang digunakan sebanyak 3772 data. Data dibagi dengan komposisi 70% menjadi data latih dan 30% menjadi data uji. Data latih digunakan dalam proses pelatihan menggunakan algoritma *Random Forest* sehingga menghasilkan model berupa kumpulan pohon keputusan yang dapat mengklasifikasikan jenis gerakan. Sementara itu, data uji digunakan untuk mengukur tingkat akurasi model. Adapun rincian data untuk masing-masing kelas disajikan pada Tabel 4.5 berikut.

**Tabel 4. 5** Pembagian Data Latih dan Data Uji

Kelas	Data Latih	Data Uji	Total
1	864	370	1234
2	843	361	1204
3	933	401	1334
Total	2640	1132	3772

Pembuatan model dilakukan menggunakan Google Colab dengan memanfaatkan library *Scikit-learn*. Pada algoritma *Random Forest*, *sklearn* menyediakan berbagai parameter yang dapat diatur untuk meningkatkan performa model, seperti jumlah pohon (*n\_estimators*) dan kedalaman maksimal pohon (*max\_depth*).

Dalam penerapan metode *Random Forest* ini, parameter yang diuji adalah jumlah pohon (*number of trees*) dan kedalaman pohon (*maximum depth*). Pemilihan kedua parameter tersebut dilakukan karena jumlah pohon memengaruhi kestabilan dan akurasi prediksi model, sedangkan kedalaman pohon menentukan kompleksitas pohon keputusan yang dibentuk. Pengaturan nilai yang tepat pada kedua parameter ini bertujuan untuk memperoleh model yang optimal dalam mengklasifikasikan gestur. Tabel 4.6 berikut menunjukkan parameter yang digunakan dalam pengujian metode *Random Forest*.

**Tabel 4. 6** Parameter Pengujian *Random Forest*

No	Parameter	Nilai Parameter
1	Jumlah Pohon	25,50,100
2	Kedalaman Maksimal Pohon	3,5,7,10

Tabel 4.6 menyajikan kombinasi parameter jumlah pohon dan kedalaman pohon yang digunakan dalam pengujian algoritma *Random Forest* untuk klasifikasi gestur. Pemilihan jumlah pohon sebanyak 25, 50, dan 100 serta variasi kedalaman

pohon sebesar 3, 5, 7, dan 10 dilakukan untuk mengevaluasi pengaruh kompleksitas model terhadap akurasi dan efisiensi waktu komputasi pada algoritma Random Forest. Variasi jumlah pohon bertujuan untuk menganalisis pengaruh peningkatan jumlah pohon terhadap kestabilan dan akurasi prediksi, karena semakin banyak pohon cenderung menghasilkan model yang lebih stabil dan akurat, meskipun memerlukan waktu pelatihan yang lebih lama. Berdasarkan pengujian awal, jumlah pohon di bawah 25 menghasilkan akurasi yang kurang dari 95%, sehingga dianggap kurang optimal untuk klasifikasi ini. Sementara itu, peningkatan jumlah pohon di atas 100 tidak memberikan peningkatan akurasi yang signifikan dibandingkan konfigurasi 100 pohon, namun berdampak pada peningkatan waktu komputasi. Oleh karena itu, kombinasi variasi parameter tersebut dianggap representatif dan efisien untuk menentukan konfigurasi model yang optimal berdasarkan aspek akurasi dan waktu komputasi.

#### 4.3.1 Pengujian Akurasi Model

Pengujian akurasi dilakukan untuk mempermudah proses evaluasi dan penentuan parameter terbaik berdasarkan tingkat akurasi untuk setiap nilai parameter. Parameter pada tabel 4.6 merupakan parameter yang digunakan untuk melihat performa akurasi dari model *random forest* yang dibangun.

**Tabel 4. 7** Confusion Matrix

Kelas Asli	Kelas Prediksi		TR (%)	FR (%)
	1	2		
1	TP	FN	TP/TP+FN	FN/TP+FN
2	FP	TN	TN/FP+TN	FP/FP+TN
Akurasi (%)			TP+TN/TP+FN+FP+TN	

Pengujian akurasi dilakukan menggunakan *confusion matrix* yang dapat dilihat pada Tabel 4.7. *Confusion matrix* digunakan untuk menilai seberapa baik sebuah model dapat memprediksi target dari sebuah data fitur. Pada *confusion matrix* terdapat beberapa istilah yang digunakan. TP (*True Positive*) merupakan jumlah sampel dari kelas 1 yang benar diprediksi sebagai kelas 1. FN (*False Negative*) merupakan jumlah sampel dari kelas 1 yang salah diprediksi sebagai kelas 2. FP (*False Positive*) merupakan jumlah sampel dari kelas 2 yang salah diprediksi sebagai kelas 1. TN (*True Negative*) merupakan jumlah sampel dari kelas 2 yang benar diprediksi sebagai kelas 2. TR (*True Rate*) merupakan rasio dari jumlah sampel suatu kelas yang diprediksi dengan benar dibagi dengan jumlah sample prediksi suatu kelas. FR (*False Rate*) merupakan rasio dari jumlah sampel suatu kelas yang diprediksi secara salah dibagi dengan jumlah sampel prediksi suatu kelas. Akurasi merupakan perbandingan dari jumlah prediksi yang benar dengan total jumlah sampel prediksi.

## 1. Akurasi dengan parameter 25 pohon

Pengujian ini dilakukan dengan 4 variasi kedalaman maksimal pohon yang berbeda agar dapat melihat pengaruh perbedaan nilai kedalaman pohon terhadap akurasi klasifikasi model *random forest*. Berikut disajikan *confusion matrix* dari hasil pengujian parameter 25 pohon dengan variasi kedalaman maksimal 3,5,7, dan 10.

**Tabel 4. 8** Jumlah Pohon 25 dengan Kedalaman Maksimal 3

Jenis Sinyal	Kelas	Kelas Prediksi			TR	FR
	Asli	1	2	3		
Kedip Sadar	1	332	0	38	89,72%	10,28%
Kedip Tidak Sadar	2	0	361	0	100%	0%
Lirik Atas	3	7	0	394	98,25%	1,75%
Akurasi					96%	

**Tabel 4. 9** Jumlah Pohon 25 dengan Kedalaman Maksimal 5

Jenis Sinyal	Kelas	Kelas Prediksi			TR	FR
	Asli	1	2	3		
Kedip Sadar	1	352	0	18	95,13%	4,87%
Kedip Tidak Sadar	2	0	361	0	100%	0%
Lirik Atas	3	5	0	396	98,75%	1,25%
Akurasi					97,96%	

**Tabel 4. 10** Jumlah Pohon 25 dengan Kedalaman Maksimal 7

Jenis Sinyal	Kelas	Kelas Prediksi			TR	FR
	Asli	1	2	3		
Kedip Sadar	1	352	0	18	95,13%	4,87%
Kedip Tidak Sadar	2	0	361	0	100%	0%
Lirik Atas	3	5	0	396	98,75%	1,25%
Akurasi					97,96%	

**Tabel 4. 11** Jumlah Pohon 25 dengan Kedalaman Maksimal 10

Jenis Sinyal	Kelas	Kelas Prediksi			TR	FR
	Asli	1	2	3		
Kedip Sadar	1	352	0	18	95,13%	4,87%
Kedip Tidak Sadar	2	0	361	0	100%	0%
Lirik Atas	3	3	0	398	99,25%	0,75%
Akurasi					98,14%	

## 2. Akurasi dengan parameter 50 pohon

Pengujian ini dilakukan dengan 4 variasi kedalaman maksimal pohon yang berbeda agar dapat melihat pengaruh perbedaan nilai kedalaman pohon terhadap akurasi klasifikasi model *random forest*. Berikut disajikan *confusion matrix* dari hasil pengujian parameter 50 pohon dengan variasi kedalaman maksimal 3,5,7, dan 10.

**Tabel 4. 12** Jumlah Pohon 50 dengan Kedalaman Maksimal 3

Jenis Sinyal	Kelas Asli	Kelas Prediksi			TR	FR
		1	2	3		
Kedip Sadar	1	335	0	35	90,54%	9,46%
Kedip Tidak Sadar	2	0	361	0	100%	100%
Lirik Atas	3	7	0	394	98,25%	1,75%
Akurasi					96,29%	

**Tabel 4. 13** Jumlah Pohon 50 dengan Kedalaman Maksimal 5

Jenis Sinyal	Kelas Asli	Kelas Prediksi			TR	FR
		1	2	3		
Kedip Sadar	1	352	0	18	95,13%	4,87%
Kedip Tidak Sadar	2	0	361	0	100%	0%
Lirik Atas	3	6	0	395	99,25%	1,5%
Akurasi					97,88%	

**Tabel 4. 14** Jumlah Pohon 50 dengan Kedalaman Maksimal 7

Jenis Sinyal	Kelas Asli	Kelas Prediksi			TR	FR
		1	2	3		
Kedip Sadar	1	362	0	8	97,83%	2,17%
Kedip Tidak Sadar	2	0	361	0	100%	0%
Lirik Atas	3	3	0	398	97%	3%
Akurasi					98,28%	

**Tabel 4. 15** Jumlah Pohon 50 dengan Kedalaman Maksimal 10

Jenis Sinyal	Kelas Asli	Kelas Prediksi			TR	FR
		1	2	3		
Kedip Sadar	1	362	0	8	97,83%	2,17%
Kedip Tidak Sadar	2	0	361	0	100%	0%
Lirik Atas	3	3	0	398	97%	3%
Akurasi					98,22%	

### 3. Akurasi dengan parameter 100 pohon

Pengujian ini dilakukan dengan 4 variasi kedalaman maksimal pohon yang berbeda agar dapat melihat pengaruh perbedaan nilai kedalaman pohon terhadap

akurasi klasifikasi model *random forest*. Berikut disajikan *confusion matrix* dari hasil pengujian parameter 100 pohon dengan variasi kedalaman maksimal 3,5,7, dan 10.

**Tabel 4. 16** Jumlah Pohon 100 dengan Kedalaman Maksimal 3

Jenis Sinyal	Kelas Asli	Kelas Prediksi			TR	FR
		1	2	3		
Kedip Sadar	1	335	0	35	90,54%	9,46%
Kedip Tidak Sadar	2	0	361	0	100%	100%
Lirik Atas	3	7	0	394	98,25%	1,75%
Akurasi					96,29%	

**Tabel 4. 17** Jumlah Pohon 100 dengan Kedalaman Maksimal 5

Jenis Sinyal	Kelas Asli	Kelas Prediksi			TR	FR
		1	2	3		
Kedip Sadar	1	352	0	18	95,13%	4,87%
Kedip Tidak Sadar	2	0	361	0	100%	0%
Lirik Atas	3	6	0	395	99,25%	1,5%
Akurasi					97,88%	

**Tabel 4. 18** Jumlah Pohon 100 dengan Kedalaman Maksimal 7

Jenis Sinyal	Kelas Asli	Kelas Prediksi			TR	FR
		1	2	3		
Kedip Sadar	1	365	0	5	98,65%	1,35%
Kedip Tidak Sadar	2	0	361	0	100%	0%
Lirik Atas	3	0	0	401	100%	2%
Akurasi					99,55%	

**Tabel 4. 19** Jumlah Pohon 100 dengan Kedalaman Maksimal 10

Jenis Sinyal	Kelas Asli	Kelas Prediksi			TR	FR
		1	2	3		
Kedip Sadar	1	365	0	5	98,65%	1,35%
Kedip Tidak Sadar	2	0	361	0	100%	0%
Lirik Atas	3	0	0	401	100%	2%
Akurasi					99,55%	

#### 4.3.2 Pengujian Waktu Komputasi Model *Random Forest*

Setelah melakukan pengujian pada variasi nilai parameter yang digunakan. Selanjutnya, dilakukan pengujian waktu komputasi model yang bertujuan untuk menentukan parameter terbaik dari segi waktu komputasi model *random forest*.

**Tabel 4. 20** Waktu Komputasi Random Forest

Parameter	Waktu			Rata-Rata
	Waktu 1	Waktu 2	Waktu 3	
<b>Jumlah Pohon 25</b>				
kedalaman 3	0.036	0.011	0.01	0.019
kedalaman 5	0.088	0.087	0.098	0.091
kedalaman 7	0.183	0.045	0.035	0.0877
kedalaman 10	0.353	0.102	0.102	0.1857
<b>Jumlah Pohon 50</b>				
kedalaman 3	0.154	0.145	0.146	0.148
kedalaman 5	0.228	0.182	0.178	0.196
kedalaman 7	0.195	0.197	0.291	0.2277
kedalaman 10	0.185	0.284	0.307	0.2587
<b>Jumlah Pohon 100</b>				
kedalaman 3	0.31	0.409	0.287	0.335
kedalaman 5	0.335	0.333	0.476	0.381
kedalaman 7	0.501	0.342	0.378	0.407
kedalaman 10	0.529	0.503	0.602	0.408

Berdasarkan hasil pengujian yang ditampilkan pada Tabel 4.20, jumlah pohon pada algoritma *Random Forest* menunjukkan pengaruh signifikan terhadap durasi komputasi. Secara umum, peningkatan jumlah pohon berbanding lurus dengan peningkatan waktu komputasi yang dibutuhkan oleh model. Adapun rata-rata waktu komputasi untuk konfigurasi 25, 50, dan 100 pohon tercatat masing-masing sebesar 0,095 detik, 0,207 detik, dan 0,383 detik. Selain itu, parameter *maximum depth* atau kedalaman maksimal pohon juga berkontribusi terhadap peningkatan waktu komputasi, meskipun dampaknya tidak sebesar jumlah pohon.

#### 4.3.3 Pemilihan Parameter Model Random Forest

Setelah dilakukan serangkaian pengujian terhadap tingkat akurasi dan waktu komputasi untuk masing-masing variasi parameter, tahap selanjutnya adalah melakukan analisis perbandingan tingkat akurasi dan waktu komputasi untuk menentukan parameter yang paling optimal. Pemilihan parameter terbaik didasarkan pada kombinasi antara tingkat akurasi yang tinggi dan waktu komputasi yang baik. Dengan demikian, tidak hanya mempertimbangkan kinerja model dari sisi akurasi, tetapi juga efisiensi waktu eksekusi, yang menjadi aspek utama untuk implementasi sistem secara *real-time*. Perbandingan nilai akurasi dan waktu komputasi untuk setiap parameter dapat dilihat pada tabel 4.21 berikut.

**Tabel 4. 21** Akurasi dan Waktu Komputasi Model *Random Forest*

Parameter		Akurasi (%)	Waktu Komputasi (s)
Jumlah Pohon	Kedalaman Pohon		
25	3	96	0.019
	5	97,96	0.091
	7	97,96	0.0877
	10	98,14	0.1857
50	3	96,29	0.148
	5	97,88	0.196
	7	98,28	0.2277
	10	98,22	0.2587
100	3	96,29	0.335
	5	97,88	0.381
	7	99,55	0.407
	10	99,55	0.545

Berdasarkan Tabel 4.21, parameter dengan akurasi data uji tertinggi sebesar 99,55% diperoleh pada konfigurasi jumlah pohon sebanyak 100 dan kedalaman pohon 7, dengan waktu komputasi sebesar 0,407 detik. Meskipun terdapat konfigurasi lain yang memiliki waktu komputasi lebih cepat, seperti konfigurasi dengan jumlah pohon 50 dan kedalaman pohon 7 yang hanya membutuhkan waktu komputasi sebesar 0,2277 detik, namun akurasi yang dihasilkan masih berada di bawah nilai tertinggi tersebut, yaitu 98,28%. Dengan mempertimbangkan kebutuhan sistem kendali kursi roda berbasis sinyal *electrooculography* (EOG), yaitu akurasi yang tinggi dengan waktu komputasi yang tetap efisien, maka dipilih parameter jumlah pohon 100 dan kedalaman pohon 7 sebagai konfigurasi paling optimal. Oleh karena itu, model *Random Forest* dengan konfigurasi jumlah pohon 100 dan kedalaman 7 yang dipilih untuk diimplementasikan secara langsung pada sistem kendali kursi roda berbasis sinyal *electrooculography* (EOG).

#### 4.4 Pengujian Durasi Locking Time

Durasi *locking time* merupakan salah satu parameter penting dalam sistem kendali kursi roda berbasis sinyal *electrooculography* (EOG). *Locking time* berfungsi sebagai jeda penguncian sinyal yang mencegah sistem mendeteksi perintah baru dalam kurun waktu tertentu setelah sinyal perintah diterima. Pengaturan ini bertujuan untuk meminimalkan kesalahan deteksi akibat refleksi, gerakan tidak disengaja, maupun sinyal balik yang dapat terjadi setelah pergerakan mata.

Pengujian ini dilakukan dengan beberapa variasi durasi *locking time* untuk menentukan nilai yang paling optimal, yaitu durasi yang mampu menekan tingkat kesalahan deteksi tanpa mengurangi kecepatan respon sistem, sehingga tetap menjaga keseimbangan antara akurasi sistem dan kenyamanan pengguna. Berikut

disajikan hasil pengujian beberapa variasi durasi *locking time* dalam Tabel 4.22 berikut.

**Tabel 4. 22** Durasi Locking Time

Jenis Perintah	Durasi Locking Time	Hasil Pengujian
Lirik Atas dan Lirik Bawah	2	Terlalu singkat, masih terjadi kesalahan deteksi
	3	Waktu paling optimal, cukup untuk mencegah salah deteksi tanpa mengurangi kecepatan sistem.
	4	Terlalu lama, mengurangi responsivitas sistem
Lirik Kanan dan Lirik Kiri	1	Waktu <i>locking</i> terlalu singkat, sistem masih rentan mendeteksi sinyal balik
	1.5	Durasi optimal, mencegah deteksi palsu tanpa mengurangi kecepatan respon sistem
	2	Durasi terlalu lama, memperlambat respon sistem terhadap perintah berikutnya.

Berdasarkan hasil pengujian yang ditampilkan pada Tabel 4.22, dapat disimpulkan bahwa pengaturan durasi *locking time* optimal berbeda untuk setiap jenis perintah guna mencapai keseimbangan antara akurasi deteksi dan kecepatan respon sistem. Untuk perintah lirik atas dan lirik bawah, durasi 3 detik dipilih sebagai *locking time* terbaik karena mampu secara efektif mencegah salah deteksi akibat sinyal balik atau gerakan tidak disengaja tanpa mengurangi responsivitas sistem. Sementara itu, untuk perintah lirik kanan dan lirik kiri durasi 1.5 detik dipilih sebagai pengaturan paling optimal. Durasi ini cukup untuk menghindari deteksi sinyal palsu akibat refleks atau sinyal balik, namun tetap menjaga sistem tetap responsif terhadap perintah selanjutnya, sehingga pengguna tidak mengalami jeda yang mengganggu dalam pengoperasian kursi roda berbasis sinyal EOG.

#### 4.5 Implementasi Kursi Roda

Setelah diperoleh fitur *polaritas* yang sesuai untuk mengklasifikasikan sinyal lirik kanan, lirik kiri, dan lirik bawah, serta model *Random Forest* yang optimal untuk mengklasifikasikan sinyal lirik atas, kedip sadar, dan kedip tidak sadar, serta dengan pengimplementasian *locking time* untuk meminimalkan kesalahan deteksi, maka dapat ditentukan klasifikasi jenis sinyal secara keseluruhan yang menjadi dasar pengoperasian sistem kendali kursi roda, sebagaimana ditampilkan pada Tabel 4.23.

**Tabel 4. 23** Hubungan Sinyal dan Output Kursi Roda

Jenis Pergerakan	Parameter	Polaritas Sinyal	Output Kursi Roda
Lirik Kanan	CH1	Positif	Belok Kanan, <i>locking time</i> 1,5 s,
	CH2	Negatif	
Lirik Kiri	CH1	Negatif	Belok Kiri, <i>locking time</i> 1,5 s
	CH2	Positif	
Lirik Bawah	CH1	Positif	Mundur, <i>locking time</i> 3 s
	CH2	Positif	
Lirik Atas	CH1	Negatif	Output Model Random Forest(3) = maju, <i>locking time</i> 3 s
	CH2	Negatif	
Kedip Sadar	CH1	Negatif	Output Model Random Forest(1) = ON/OFF
	CH2	Negatif	
Kedip tidak Sadar	CH1	Negatif	Output Model Random Forest(2) = lanjut operasi
	CH2	Negatif	

Pengujian pada kursi roda dilakukan oleh responden dan memerlukan beberapa komponen pendukung agar proses pengujian berjalan dengan lancar. Pengimplementasian perangkat keras yang digunakan pada kursi roda ini dapat dilihat pada Gambar 4.9.



**Gambar 4. 9** Pengimplementasian Perangkat Keras Kursi Roda

Beberapa komponen perangkat keras sistem kendali kursi roda ditampilkan pada Gambar 4.9, dengan penjelasan sebagai berikut:

1. Elektroda  
Empat buah elektroda digunakan untuk akuisisi sinyal, dengan penempatan sesuai hasil kajian dari tinjauan pustaka yang telah dilakukan.
2. Kotak EOG  
Berisi rangkaian *Electrooculography (EOG)* dan perangkat akuisisi data NI USB-6008, yang berfungsi untuk mengonversi sinyal mata menjadi sinyal digital.
3. Laptop  
Digunakan untuk menjalankan program pengolahan data, termasuk proses klasifikasi sinyal serta tampilan antarmuka (*GUI*) yang telah dirancang.
4. Rangkaian Kursi Roda  
Terdiri atas mikrokontroler *Arduino Nano*, motor driver BS7960, dan motor DC. Rangkaian ini berfungsi untuk menggerakkan kursi roda berdasarkan hasil keputusan yang dikirimkan dari laptop.

Pembuatan antarmuka (*GUI*) program untuk sistem kendali kursi roda dilakukan menggunakan bahasa pemrograman Python pada *Visual Studio Code*. Antarmuka ini dirancang untuk menampilkan hasil deteksi arah gerakan kursi roda secara real-time berdasarkan sinyal EOG yang diterima. Tampilan *GUI* disusun secara sederhana dan informatif untuk memudahkan pengguna dalam memantau arah gerakan yang terdeteksi oleh sistem. Antarmuka program untuk kursi roda ini dapat dilihat pada Gambar 4.10 berikut.



Gambar 4. 10 Tampilan GUI

## 4.6 Pengujian Kursi Roda

Pengujian yang dilakukan pada kursi roda terdiri dari tiga tahapan, yaitu pengujian ketepatan klasifikasi untuk mengevaluasi metode *random forest*, pengujian pergerakan kursi roda, dan pengujian lintasan kursi roda. Seluruh tahapan pengujian ini menjadi parameter penentu keberhasilan penerapan metode *Random Forest* sebagai model *machine learning* dalam sistem kendali kursi roda berbasis sinyal EOG.

### 4.6.1 Pengujian Ketepatan Klasifikasi *Random Forest*

Pengujian pertama yang dilakukan adalah pengujian terhadap model *Random Forest* yang telah dirancang dan dilatih. Model ini hanya diuji untuk mengklasifikasikan sinyal-sinyal yang memiliki karakteristik polaritas negatif, yaitu sinyal lirik atas, sinyal kedip sadar, dan sinyal kedip tidak sadar. Proses pengujian dilakukan terhadap 10 orang responden, di mana masing-masing responden menjalani tiga kali percobaan. Pada setiap percobaan, responden diminta untuk melakukan gerakan sesuai perintah yang diberikan oleh peneliti, kemudian sistem akan memproses sinyal dan model *Random Forest* akan melakukan prediksi terhadap jenis gerakan tersebut. Hasil dari pengujian akurasi model dapat dilihat pada Tabel 4.24 berikut.

Tabel 4. 24 Pengujian Klasifikasi Metode Random Forest

No	Responden	Percobaan	Kelas Prediksi		
			Kedip sadar	Tidak Sadar	Lirik Atas
1	A	1	✓	✓	✓
		2	✓	✓	✓
		3	✓	✓	✓
2	B	1	✓	✓	✓
		2	✓	✓	✓
		3	✓	✓	✓
3	C	1	✓	✓	✓
		2	✓	✓	×
		3	✓	✓	✓
4	D	1	✓	✓	✓
		2	✓	✓	✓
		3	✓	✓	✓
5	E	1	✓	✓	✓
		2	✓	✓	✓
		3	✓	✓	✓
6	F	1	✓	✓	✓

		2	✓	✓	✓
		3	✓	✓	✓
7	G	1	✓	✓	✓
		2	✓	✓	✓
		3	✓	✓	✓
8	H	1	✓	✓	✓
		2	✓	✓	✓
		3	✓	✓	✓
9	I	1	✓	✓	✓
		2	✓	✓	✓
		3	✓	✓	✓
10	J	1	✓	✓	✓
		2	✓	✓	✓
		3	✓	✓	✓
	Akurasi		100%	100%	96,67%
	Akurasi Total			98,9%	

Berdasarkan hasil pengujian yang ditampilkan pada Tabel 4.24, model *Random Forest* yang digunakan dalam sistem klasifikasi sinyal menunjukkan kinerja yang cukup baik. Akurasi klasifikasi untuk sinyal kedip sadar adalah sebesar 100%, kedip tidak sadar sebesar 100%, dan lirik atas sebesar 96,67%, dengan total akurasi keseluruhan sebesar 98,9%. Dengan demikian, *error* total yang dihasilkan adalah sebesar 1,1%.

Hasil ini menunjukkan bahwa metode klasifikasi berbasis *Random Forest* mampu secara efektif membedakan ketiga jenis sinyal yang memiliki karakteristik polaritas negatif, yaitu kedip sadar, kedip tidak sadar, dan lirik atas. Akurasi tertinggi terdapat pada sinyal kedip sadar dan kedip tidak sadar sebesar 100%. Sementara itu sinyal lirik atas memperoleh akurasi 96,67%. Walaupun masih ada kesalahan, akurasi ini menunjukkan bahwa pola sinyal ini sangat konsisten dan mudah dikenali oleh model random forest. Dengan demikian, tingkat akurasi total sebesar 98,9% dan *error* sebesar 1,1% masih dapat diterima dan cukup layak untuk diimplementasikan pada sistem kendali kursi roda berbasis sinyal EOG.

#### 4.6.2 Pengujian Ketepatan Gerakan Kursi Roda

Pengujian ketepatan pergerakan kursi roda dilakukan dengan mengevaluasi kesesuaian antara instruksi gerakan mata yang diberikan dan respons pergerakan aktual dari kursi roda. Pengujian ini dilakukan terhadap 10 responden, di mana masing-masing responden diminta untuk melakukan seluruh jenis gerakan mata yang dibutuhkan sesuai instruksi. Setiap responden melakukan 3 kali percobaan, sehingga dapat dievaluasi konsistensi sistem dalam menerjemahkan sinyal EOG menjadi perintah pergerakan kursi roda. Hasil pengujian ketepatan pergerakan kursi roda ditampilkan pada Tabel 4.25.

**Tabel 4. 25** Hasil Pengujian Ketepatan Gerakan Kursi Roda

No	Responden	Percobaan	Kelas Prediksi				
			ON/OFF	Kanan	Kiri	Maju	Mundur
1	A	1	ON/OFF	Kanan	Kiri	Maju	Mundur
		2	ON/OFF	Kanan	Kiri	Maju	Mundur
		3	MAJU	Kanan	Kiri	Maju	Mundur
2	B	1	ON/OFF	Kanan	Kiri	Maju	Mundur
		2	ON/OFF	Kanan	Kiri	Maju	Mundur
		3	ON/OFF	Kanan	Kiri	Maju	Mundur
3	C	1	ON/OFF	Kanan	Kiri	Maju	Mundur
		2	ON/OFF	Kanan	Kiri	Maju	Mundur
		3	ON/OFF	Kanan	Kiri	Maju	Mundur
4	D	1	ON/OFF	Kanan	Kiri	Maju	Mundur
		2	ON/OFF	Kanan	Kiri	ON/OFF	Mundur
		3	ON/OFF	Kanan	Kiri	Maju	Mundur
5	E	1	ON/OFF	Kanan	Kiri	Maju	Mundur
		2	ON/OFF	Kanan	Kiri	Maju	Mundur
		3	ON/OFF	Kanan	Kiri	Maju	Mundur
6	F	1	ON/OFF	Kanan	Kiri	Maju	Mundur
		2	ON/OFF	Kanan	Kiri	Maju	Mundur
		3	ON/OFF	Kanan	Kiri	Maju	Mundur
7	G	1	ON/OFF	Kanan	Kiri	Maju	Mundur
		2	ON/OFF	Kanan	Kiri	Maju	Mundur
		3	ON/OFF	Kanan	Kiri	Maju	Mundur
8	H	1	ON/OFF	Kanan	Kiri	Maju	Mundur
		2	ON/OFF	Kanan	Kiri	Maju	Mundur
		3	ON/OFF	Kanan	Kiri	Maju	Mundur
9	I	1	ON/OFF	Kanan	Kiri	Maju	Mundur
		2	ON/OFF	Kanan	Kiri	Maju	Mundur
		3	ON/OFF	Kanan	Kiri	Maju	Mundur
10	J	1	ON/OFF	Kanan	Kiri	Maju	Mundur
		2	ON/OFF	Kanan	Kiri	Maju	Mundur

No	Responden	Percobaan	Kelas Prediksi				
			ON/OFF	Kanan	Kiri	Maju	Mundur
		3	ON/OFF	Kanan	Kiri	Maju	Mundur
	Akurasi		97%	100%	100%	97%	100%
	Akurasi Total			98,8%			

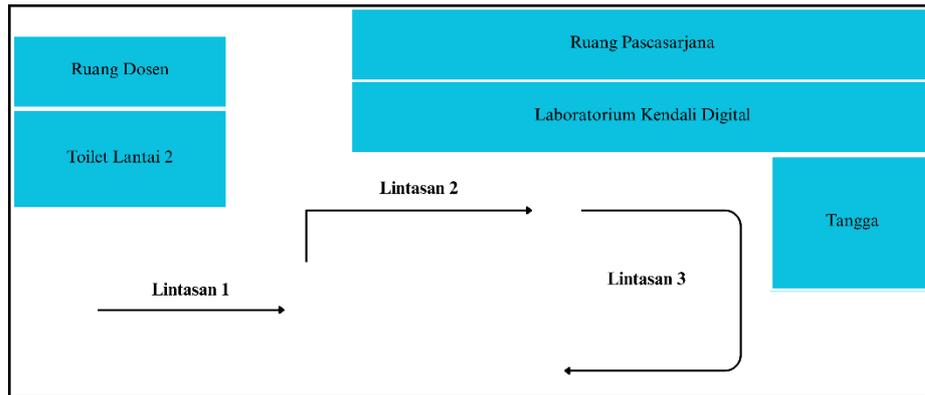
Berdasarkan hasil pengujian yang ditampilkan pada Tabel 4.25, akurasi deteksi gerakan kursi roda menunjukkan hasil yang cukup tinggi dan dapat diterima. Akurasi untuk perintah ON/OFF adalah sebesar 97%, perintah kanan dan kiri masing-masing mencapai 100%, perintah maju sebesar 97%, dan perintah mundur sebesar 100%. Adapun total akurasi keseluruhan sistem adalah 97%. Gerakan dengan tingkat akurasi 100%, yaitu perintah kanan, kiri, dan mundur, memiliki karakteristik sinyal yang mudah dibedakan, ditandai dengan perbedaan polaritas yang jelas pada sinyal-sinyal EOG. Hal ini memudahkan sistem dalam mengenali dan mengklasifikasikan perintah dengan tingkat kesalahan yang sangat kecil, bahkan nol.

Sementara itu, akurasi untuk maju masih menunjukkan adanya kesalahan deteksi, meskipun nilainya tergolong kecil. Faktor utama yang memengaruhi ketidaksempurnaan tersebut adalah adanya kemiripan pola sinyal antara sinyal kedip sadar dan lirik atas. Selain itu, faktor keterampilan pengguna dalam melakukan gerakan sesuai instruksi juga berpengaruh terhadap akurasi, termasuk keterbatasan waktu dalam proses pelatihan pengguna sebelum pengujian dilakukan.

Meskipun demikian, tingkat akurasi yang dicapai, terutama total akurasi sebesar 98,8%, dapat dikategorikan sangat baik dan layak untuk implementasi. Artinya, sistem kendali kursi roda berbasis sinyal EOG dengan metode *Random Forest* dan klasifikasi berdasarkan polaritas sinyal sudah mampu mengenali perintah gerakan dengan tingkat ketepatan yang tinggi dan *error* yang minimal.

#### 4.6.3 Pengujian Kursi Roda menggunakan Lintasan

Pengujian lintasan dilakukan untuk menguji kemampuan kursi roda dalam bergerak mengikuti jalur tertentu yang telah ditetapkan. Tujuan utama dari pengujian ini adalah untuk mengevaluasi kelancaran pergerakan kursi roda serta mengukur performa sistem berdasarkan waktu tempuh dalam mengikuti lintasan yang telah ditentukan. Pengujian dilakukan dalam lintasan dengan panjang 20 meter terhadap 10 responden, di mana masing-masing responden diberikan kesempatan 1 kali percobaan untuk mengoperasikan kursi roda mengikuti lintasan uji. Lintasan yang digunakan dalam pengujian ini berada di lantai 2 Departemen Teknik Elektro, Universitas Andalas, dengan rute yang telah ditentukan dan dirancang untuk menguji berbagai arah gerakan kursi roda. Denah lintasan uji dapat dilihat pada Gambar 4.11.



**Gambar 4. 11** Lintasan Pengujian Kursi Roda

Pada pengujian lintasan, responden memulai dengan mengaktifkan kursi roda menggunakan perintah *on/off*, lalu bergerak maju hingga mencapai batas akhir lintasan pertama dan berhenti. Selanjutnya, responden melanjutkan ke lintasan kedua dengan melakukan belokan ke kiri, bergerak lurus ke depan, berbelok ke kanan, lalu kembali bergerak lurus hingga lintasan kedua selesai. Setelah itu, responden masuk ke lintasan ketiga, bergerak maju mengikuti jalur lintasan sebelumnya, melakukan belokan ke kanan di area depan tangga lantai atas, kemudian setelah menghadap ke arah posisi awal (*start*), responden kembali bergerak maju hingga mencapai titik akhir lintasan ketiga. Hasil dari pengujian lintasan kursi roda dapat dilihat pada tabel 4.26 berikut.

**Tabel 4. 26** Hasil Pengujian Performa Kursi Roda terhadap Lintasan

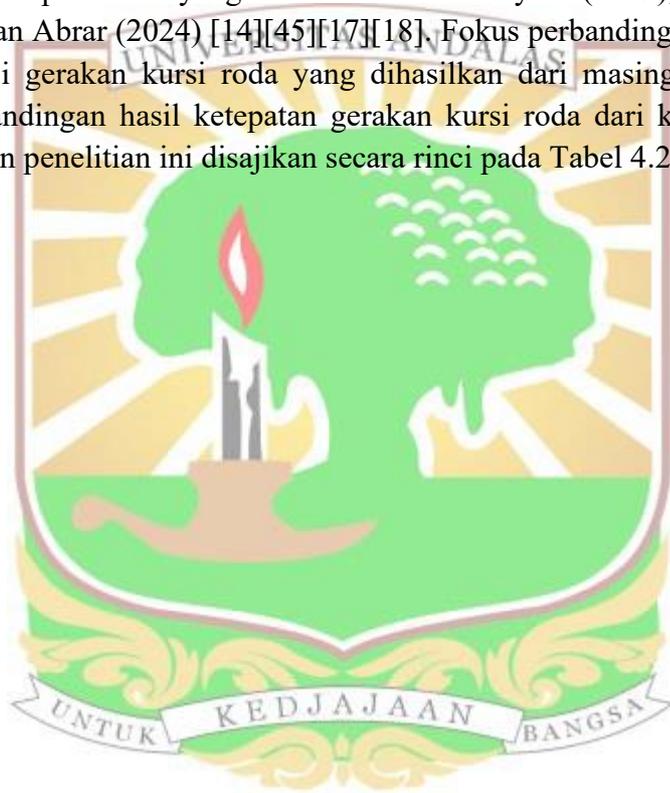
No	Responden	Lintasan 1	Lintasan 2	Lintasan 3	Total Waktu Satu Putaran (s)	Kecepatan Gerak Kursi Roda (m/s)
1	A	✓	✓	✓	79,17	0,25
2	B	✓	✓	✓	82,68	0,24
3	C	✓	✓	✓	80,81	0,247
4	D	✓	✓	✓	77,80	0,257
5	E	✓	✓	✓	77,70	0,257
6	F	✓	✓	✓	81,12	0,246
7	G	✓	✓	✓	79,93	0,25
8	H	✓	✓	✓	79,09	0,25
9	I	✓	✓	✓	80,23	0,249
10	J	✓	✓	✓	79,27	0,252
Rata – rata					79,78	0,249

Tabel 4.26 menyajikan hasil pengujian performa kursi roda terhadap tiga jenis lintasan yang dilalui oleh 10 responden. Berdasarkan data tersebut, seluruh

responden berhasil menyelesaikan ketiga lintasan dengan baik dan lancar. Secara keseluruhan, rata-rata total waktu tempuh satu putaran lintasan adalah 79,78 detik. Berdasarkan total panjang lintasan yang ditempuh, kecepatan rata-rata kursi roda dihitung sebesar 0,249 m/s, yang menunjukkan performa sistem kendali kursi roda cukup stabil dan responsif dalam mengikuti perintah dari pengguna.

#### **4.7 Perbandingan dengan Penelitian Sebelumnya**

Setelah dilakukan pengujian terhadap ketepatan gerakan kursi roda, hasil yang diperoleh dalam penelitian ini kemudian dibandingkan dengan beberapa studi sebelumnya guna memperoleh gambaran yang lebih komprehensif mengenai performa sistem kendali yang digunakan. Penelitian yang dijadikan acuan perbandingan meliputi studi yang dilakukan oleh Bhuyain (2019), Fityah (2021), Arif (2022), dan Abrar (2024) [14][45][17][18]. Fokus perbandingan terletak pada tingkat akurasi gerakan kursi roda yang dihasilkan dari masing-masing sistem kendali. Perbandingan hasil ketepatan gerakan kursi roda dari ketiga penelitian tersebut dengan penelitian ini disajikan secara rinci pada Tabel 4.27.



**Tabel 4. 27** Perbandingan terhadap penelitian sebelumnya

Peneliti	Klasifikasi	Metode	Aplikasi	Parameter		Akurasi Gerak n Kursi roda
				Akurasi Model	Kesalahan Navigasi ON/ OFF	
Bhuyain (2019) [14]	Nilai <i>Threshold</i>	Analisis Durasi sinyal	MATLAB dan Arduino IDE	-	-	8% 90%
Fityah (2021) [45]	Fitur Polaritas dan Puncak Sinyal (2 sinyal)	Jaringan Syaraf Tiruan	MATLAB dan LABVIEW	97%	14%	5% 93,3%
Arif (2022) [17]	Fitur Polaritas dan Puncak Sinyal (2 sinyal)	SVM ( <i>Support Vector Machine</i> )	MATLAB dan LABVIEW	97,20%	9%	4,44% 96,29%
Abrar (2024) [18]	Fitur Polaritas dan Puncak Sinyal (3 Sinyal)	<i>Decision Tree</i>	<i>Visual Studio Code</i> dan Arduino IDE	91%	5%	0% 98,8%
Gilang (2025)	Fitur Polaritas dan Puncak Sinyal (3 sinyal)	<i>Random Forest</i>	<i>Visual Studio Code</i> dan Arduino IDE	99,55%	0%	3% 98,8%

Tabel 4.27 menampilkan perbandingan terhadap penelitian sebelumnya serta metode dan fitur yang digunakan pada masing-masing penelitian. Penelitian Bhuyain menggunakan parameter *threshold* untuk membedakan semua sinyal EOG dan menggunakan metode perbedaan polaritas serta durasi sinyal untuk mengklasifikasikan jenis pergerakan. Penelitian Fityah dan Arif juga menggunakan parameter *threshold*, namun ditambah dengan fitur puncak sinyal pada dua sinyal (lirik atas dan kedip sadar). Walaupun menggunakan fitur yang sama, metode yang digunakan oleh Fityah dan Arif berbeda. Pada penelitian Fityah, proses klasifikasi dilakukan menggunakan metode Jaringan Syaraf Tiruan (JST), sedangkan pada penelitian Arif menggunakan metode *Support Vector Machine* (SVM). Sementara itu, penelitian oleh Abrar menggunakan fitur puncak sinyal pada 3 sinyal (lirik atas,

kedip sadar, dan kedip tidak sadar) dengan menggunakan metode *decision tree*. Pada penelitian ini juga menggunakan fitur puncak sinyal untuk klasifikasi 3 sinyal (lirik atas, kedip sadar, dan kedip tidak sadar), namun menggunakan metode klasifikasi *random forest*.

Perbandingan aplikasi untuk perancangan sistem klasifikasi dan pergerakan kursi roda juga menunjukkan perbedaan. Pada penelitian sebelumnya, proses klasifikasi sinyal dilakukan menggunakan MATLAB, sementara untuk pengendalian kursi roda digunakan dua jenis program, yaitu Arduino IDE dan LABVIEW. Pada penelitian ini, program klasifikasi dirancang menggunakan *Visual Studio Code*, sebuah IDE yang fleksibel dan dapat diintegrasikan dengan berbagai bahasa pemrograman. Bahasa pemrograman yang digunakan adalah *Python* karena memiliki banyak pustaka dan library yang mendukung proses pengembangan sistem klasifikasi. Sementara itu, untuk pengendalian kursi roda digunakan Arduino IDE karena mikrokontroler yang digunakan adalah Arduino Nano.

Berdasarkan hasil perbandingan akurasi model, pada penelitian Bhuyain tidak dicantumkan nilai akurasi model secara eksplisit sehingga tidak dapat dibandingkan secara langsung. Sementara itu, pada penelitian Fityah diperoleh akurasi model sebesar 97% menggunakan metode Jaringan Syaraf Tiruan (JST). Pada penelitian Arif, akurasi model meningkat menjadi 97,20% dengan menggunakan metode *Support Vector Machine* (SVM). Selanjutnya, pada penelitian Abrar, walaupun menggunakan klasifikasi tiga sinyal, akurasi model yang diperoleh sebesar 91% dengan metode *Decision Tree*. Pada penelitian ini, dilakukan pengembangan lebih lanjut dengan mengganti metode klasifikasi menjadi *Random Forest* untuk klasifikasi 3 sinyal (lirik atas, kedip sadar, dan kedip tidak sadar). Hasilnya, sistem berhasil mencapai akurasi model tertinggi sebesar 99,5%. Peningkatan ini menunjukkan bahwa walaupun fitur yang digunakan tidak mengalami perubahan, pemilihan algoritma klasifikasi yang lebih tepat dan efektif tetap mampu memberikan pengaruh besar terhadap kemampuan sistem dalam mengenali pola sinyal EOG secara lebih akurat.

Berdasarkan hasil perbandingan kesalahan navigasi, pada penelitian Bhuyain belum diterapkan fitur gerakan ON/OFF, namun pada gerakan maju tercatat kesalahan sebesar 8%. Pada penelitian Fityah, diperoleh kesalahan ON/OFF sebesar 14% dan kesalahan gerakan maju sebesar 5%. Sementara itu, penelitian Arif menunjukkan kesalahan ON/OFF sebesar 9% dan kesalahan pada gerakan maju sebesar 4,44%. Pada penelitian Abrar, kesalahan ON/OFF berhasil ditekan hingga 5%, dan tidak ditemukan kesalahan pada gerakan maju (0%). Dalam penelitian ini, diperoleh kesalahan navigasi ON/OFF sebesar 0% dan kesalahan pada gerakan maju sebesar 3%.

Perbandingan terakhir dilakukan terhadap akurasi pada saat pengujian ketepatan pergerakan kursi roda. Pada penelitian Bhuyain diperoleh akurasi sebesar 90%, penelitian Fityah sebesar 93,3%, dan Arif sebesar 96,29%. Penelitian Abrar

dengan *decision tree* menghasilkan akurasi sebesar 98,8. Sementara itu, pada penelitian ini diperoleh akurasi sebesar 98,8% yang menunjukkan peningkatan akurasi dari penelitian sebelumnya. Hal ini mengindikasikan bahwa metode klasifikasi yang digunakan dalam penelitian ini berhasil meningkatkan ketepatan pergerakan kursi roda secara signifikan dibandingkan penelitian-penelitian sebelumnya. Selain itu, pada penelitian ini juga ditambahkan fitur locking yang berfungsi untuk menjaga kesalahan deteksi perintah baru.



## BAB V KESIMPULAN DAN SARAN

### 5.1 Kesimpulan

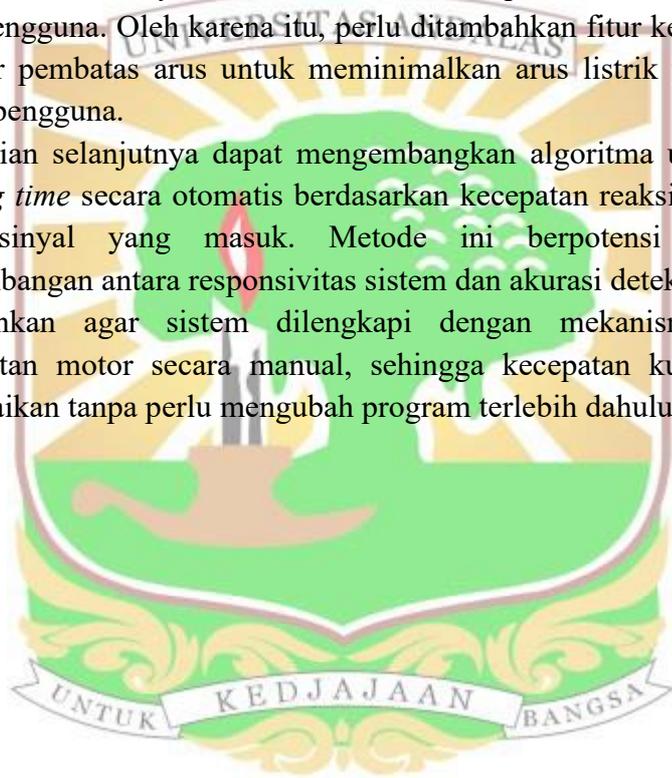
Penelitian ini menghasilkan sistem klasifikasi berbasis *Random Forest* yang mampu membedakan tiga jenis sinyal mata, yaitu lirik atas, kedip sadar, dan kedip tidak sadar menggunakan fitur puncak sinyal. Sistem ini juga mampu mengklasifikasikan arah lirik kiri, kanan, dan bawah berdasarkan perbedaan polaritas antar channel sinyal EOG. Berdasarkan rangkaian pengujian yang telah dilakukan, dapat disimpulkan beberapa hal berikut:

1. Proses klasifikasi tiga jenis sinyal dengan polaritas yang mirip (kedip sadar, kedip tidak sadar, dan lirik atas) dilakukan menggunakan metode *random forest*, dengan fitur puncak sinyal. Hasil akurasi model *random forest* menunjukkan akurasi sebesar 99,55% pada data uji. Hasil Akurasi model ini lebih besar daripada penelitian sebelumnya.
2. Pengujian pengenalan gerakan lirik atas, kedip sadar, dan kedip tidak sadar menunjukkan akurasi sebesar 98,9% berdasarkan pengujian terhadap 10 responden. Hal ini menunjukkan performa model yang stabil dan efektif dalam membedakan sinyal dengan karakteristik polaritas sinyal yang sama.
3. Pengujian *locking time* menunjukkan bahwa durasi *locking time* yang optimal berbeda untuk tiap jenis perintah. Durasi 3 detik efektif digunakan untuk lirik atas dan bawah, sedangkan 1,5 detik optimal untuk lirik kiri dan kanan. Durasi ini mampu menghindari kesalahan deteksi akibat sinyal balik dan gerakan refleks mata tanpa mengurangi responsivitas sistem, sehingga meningkatkan akurasi dan kenyamanan pengguna dalam mengoperasikan kursi roda berbasis EOG.
4. Pengujian kendali kursi roda dilakukan terhadap 10 responden untuk mengevaluasi akurasi gerakan kursi roda, serta performa navigasi pada lintasan. Akurasi pergerakan kursi roda tercatat sebesar 98,8%. Nilai akurasi ini lebih besar daripada beberapa penelitian sebelumnya.
5. Untuk pengujian performa lintasan, evaluasi dilakukan berdasarkan kelancaran dan total waktu tempuh yang dibutuhkan responden dalam menyelesaikan satu putaran lintasan. Rata-rata waktu tempuh yang diperoleh adalah 79,78 detik, yang menunjukkan bahwa sistem mampu merespons perintah dengan baik.
6. Hasil yang didapat pada penelitian ini menunjukkan peningkatan dari penelitian sebelumnya. Perbandingan penelitian ini dengan penelitian sebelumnya terjadi peningkatan akurasi model dan juga akurasi pengujian ketepatan klasifikasi dan pergerakan kursi roda. Dengan demikian, tujuan untuk meningkatkan performa kursi roda berhasil tercapai.

## 5.2 Saran

Penelitian yang telah dilaksanakan selama periode tugas akhir ini masih memiliki sejumlah keterbatasan dan membuka peluang untuk pengembangan lebih lanjut. Oleh karena itu, dalam rangka mendukung kesinambungan riset mengenai sistem kendali kursi roda berbasis sinyal EOG, peneliti menyampaikan beberapa saran yang diharapkan dapat menjadi acuan dalam penelitian lanjutan. Adapun saran-saran tersebut antara lain:

1. Penelitian selanjutnya disarankan untuk mencoba algoritma yang lebih kompleks seperti *Extreme Gradient Boosting* (XGBoost) maupun jaringan syaraf tiruan (JST) modern. Algoritma ini berpotensi meningkatkan akurasi klasifikasi sinyal secara signifikan.
2. Sistem akuisisi sinyal EOG masih memiliki potensi risiko kejutan listrik bagi pengguna. Oleh karena itu, perlu ditambahkan fitur keamanan seperti resistor pembatas arus untuk meminimalkan arus listrik yang masuk ke tubuh pengguna.
3. Penelitian selanjutnya dapat mengembangkan algoritma untuk mengatur *locking time* secara otomatis berdasarkan kecepatan reaksi pengguna atau pola sinyal yang masuk. Metode ini berpotensi meningkatkan keseimbangan antara responsivitas sistem dan akurasi deteksi perintah.
4. Disarankan agar sistem dilengkapi dengan mekanisme pengaturan kecepatan motor secara manual, sehingga kecepatan kursi roda dapat disesuaikan tanpa perlu mengubah program terlebih dahulu.



## DAFTAR PUSTAKA

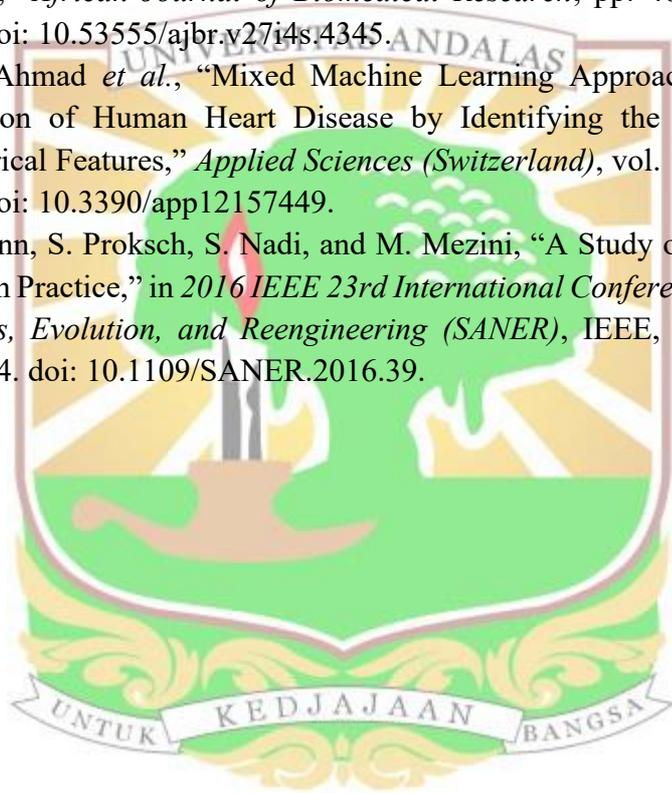
- [1] Republik Indonesia, “UNDANG-UNDANG REPUBLIK INDONESIA NOMOR 8 TAHUN 2016 TENTANG PENYANDANG DISABILITAS,” JAKARTA, 2016.
- [2] Staf Ahli Bidang Sosial dan Penanggulangan Kemiskinan Kementerian Perencanaan Pembangunan Nasional, “Kajian Disabilitas Tinjauan Peningkatan Akses dan Taraf Hidup Penyandang Disabilitas Indonesia : Aspek Sosio-Ekonomi dan Yuridis,” Jakarta, 2021.
- [3] A. Pollini *et al.*, “Human-Machine Interaction Design in Adaptive Automation,” in *Procedia Computer Science*, Elsevier B.V., 2025, pp. 1034–1044. doi: 10.1016/j.procs.2025.01.165.
- [4] A. Hartman and V. K. Nandikolla, “Human-Machine Interface for a Smart Wheelchair,” *Journal of Robotics*, vol. 2019, 2019, doi: 10.1155/2019/4837058.
- [5] A. K. Ridia, A. Hidayat, and Derisma, “PENERAPAN METODE FUZZY LOGIC PADA KURSI RODA ELEKTRIK DENGAN KENDALI SUARA,” *Seminar Nasional Sains dan Teknologi*, vol. 2017, pp. 1–2, 2017.
- [6] T. A. Angwyn and Thiang, “KONTROL MINIATUR KURSI RODA DENGAN PERINTAH SUARA BERBASIS VOICE RECOGNITION MODULE,” *Jurnal Teknik Elektro*, vol. 16, no. 2, pp. 71–75, 2023.
- [7] E. J. Rechy-Ramirez and H. Hu, “Bio-signal based control in assistive robots: a survey,” *Digital Communications and Networks*, vol. 1, no. 2, pp. 85–101, Apr. 2015, doi: 10.1016/j.dcan.2015.02.004.
- [8] M. A. Skoglund, M. Andersen, M. M. Shiell, G. Keidser, M. L. Rank, and S. Rotger-Griful, “Comparing In-ear EOG for Eye-Movement Estimation With Eye-Tracking: Accuracy, Calibration, and Speech Comprehension,” *Front Neurosci*, vol. 16, Jun. 2022, doi: 10.3389/fnins.2022.873201.
- [9] F. Davarinia and A. Maleki, “SSVEP-gated EMG-based decoding of elbow angle during goal-directed reaching movement,” *Biomed Signal Process Control*, vol. 71, p. 103222, Jan. 2022, doi: 10.1016/j.bspc.2021.103222.
- [10] S. K. Saini and R. Gupta, “Artificial intelligence methods for analysis of electrocardiogram signals for cardiac abnormalities: state-of-the-art and future challenges,” *Artif Intell Rev*, vol. 55, no. 2, pp. 1519–1565, Feb. 2022, doi: 10.1007/s10462-021-09999-7.
- [11] M. Sazgar and M. G. Young, “Overview of EEG, Electrode Placement, and Montages,” in *Absolute Epilepsy and EEG Rotation Review*, Cham: Springer International Publishing, 2019, pp. 117–125. doi: 10.1007/978-3-030-03511-2\_5.
- [12] F. A. Naoum, S. K. Fakhrey, J. A. Abdulameer, and N. S. A. Zidan, “Analysis of Electrooculogram in Detecting Eye Movements Associated

- with Brain Injury,” *European Journal of Theoretical and Applied Sciences*, vol. 2, no. 5, pp. 516–529, Sep. 2024, doi: 10.59324/ejtas.2024.2(5).50.
- [13] Q. Huang *et al.*, “An EOG-Based Human-Machine Interface for Wheelchair Control,” *IEEE Trans Biomed Eng*, vol. 9294, no. c, pp. 1–11, 2017, doi: 10.1109/TBME.2017.2732479.
- [14] Md. F. Bhuyain, Md. A.-U. Kabir Shawon, N. Sakib, T. Faruk, Md. K. Islam, and K. M. Salim, “Design and Development of an EOG-based System to Control Electric Wheelchair for People Suffering from Quadriplegia or Quadriparesis,” in *2019 International Conference on Robotics, Electrical and Signal Processing Techniques (ICREST)*, IEEE, Jan. 2019, pp. 460–465. doi: 10.1109/ICREST.2019.8644378.
- [15] A. N. Rajesh, S. Chandralingam, T. Anjaneyulu, and K. Satyanarayana, “EOG Controlled Motorized Wheelchair for Disabled Persons,” *International Scholarly and Scientific Research & Innovation*, vol. 8, no. 5, pp. 302–305, 2014.
- [16] R. Z. Marandi, P. Madeleine, Ø. Omland, N. Vuillerme, and A. Samani, “Eye movement characteristics reflected fatigue development in both young and elderly individuals,” *Sci Rep*, pp. 1–10, 2018, doi: 10.1038/s41598-018-31577-1.
- [17] M.A. Rahman, “Perancangan Sistem On/Off pada Kursi Roda EOG untuk Menghindari Kesalahan Navigasi Akibat Mata Lelah menggunakan Metode Jaringan Saraf Tiruan,” Universitas Andalas, 2022.
- [18] M. Abrar A.B, “Perancangan Sistem ON/OFF pada Kursi Roda Menggunakan Sensor Electrooculography dengan Metode Decision Tree untuk Menghindari Kesalahan Navigasi pada Mata,” Universitas Andalas, Padang, 2024.
- [19] A. Akbar, G. Abdel, N. Masikki, A. N. Aliansyah, and N. Z. D. L. Mulyawati, “Perancangan Sistem Monitoring Navigasi Kursi Roda Berbasis Mikrokontroler,” *Jurnal Teknik Elektro dan Vokasional*, vol. 7, no. 1, pp. 45–52, 2021.
- [20] S. Annirohman, F. D. Puspitarini, and D. Natalia, “RANCANGAN KURSI RODA ELEKTRIK DENGAN REMOT DAN TUAS,” *Jurnal Teknik*, vol. 12, no. 01, pp. 89–98, 2023.
- [21] M. A. Nursasongko, I. A. E. Zaeni, and A. N. Handayani, “Kendali Model Kursi Roda dengan Electromyograf dan Accelerometer Menggunakan Metode Jaringan Saraf Tiruan,” *ALINIER JURNAL*, vol. 1, no. 2, 2020.
- [22] F. Fahrozi, “Perancangan Pengontrol Otomatis dan Pengatur Posisi Tempat Duduk pada Kursi,” *JURNAL PERANCANGAN, MANUFAKTUR, MATERIAL, DAN ENERGI*, vol. 2, no. 1, pp. 45–52, 2020.
- [23] K. S. Moon, S. Q. Lee, J. S. Kang, A. Hnat, and D. B. Karen, “A Wireless Electrooculogram (EOG) Wearable Using Conductive Fiber Electrode,”

- Electronics (Basel)*, vol. 12, no. 3, p. 571, Jan. 2023, doi: 10.3390/electronics12030571.
- [24] A. López, M. Fernández, H. Rodríguez, and F. Ferrero, “Development of an EOG-based system to control a serious game,” *Measurement*, vol. 127, no. April, pp. 481–488, 2018.
- [25] S. Debbarma, S. Nabavi, and S. Bhadra, “A Wireless Flexible Electrooculogram Monitoring System With Printed Electrodes,” in *2021 IEEE International Instrumentation and Measurement Technology Conference (I2MTC)*, IEEE, May 2021, pp. 1–6. doi: 10.1109/I2MTC50364.2021.9459971.
- [26] L. R. Sullivan, “Technical Tips: Eye Movement Monitoring,” *American Journal of EEG Technology*, vol. 33, no. 2, pp. 135–147, Jun. 2015, doi: 10.1080/00029238.1993.11080442.
- [27] R. Reda, M. Tantawi, H. shedeed, and M. F. Tolba, “Analyzing Electrooculography (EOG) for Eye Movement Detection,” 2020, pp. 179–189. doi: 10.1007/978-3-030-14118-9\_18.
- [28] M. I. Rusydi, M. Sasaki, and S. Ito, “Affine transform to reform pixel coordinates of EOG signals for controlling robot manipulators using gaze motions,” *Sensors (Switzerland)*, vol. 14, no. 6, pp. 10107–10123, 2014, doi: 10.3390/s140610107.
- [29] A. López, F. Ferrero, and O. Postolache, “High-Performance Analog Front-End (AFE) for EOG Systems,” *Electronics (Basel)*, vol. 9, no. 970, pp. 1–15, 2020.
- [30] A. Fauzan and U. T. Madura, “Peran Penting Sensor dan Aktuoator Dalam Biomedik Serta Perancangan Elektromiografi ( EMG ) Untuk Mendeteksi Aktivitas Listrik Pada Otot Manusia,” *Seminar Nasional Fortei*, vol. 7, no. 4, pp. 602–609, 2021.
- [31] K. Polachan, B. Chatterjee, S. Weigand, and S. Sen, “Human Body–Electrode Interfaces for Wide-Frequency Sensing and Communication: A Review,” *Nanomaterials*, vol. 11, no. 8, p. 2152, Aug. 2021, doi: 10.3390/nano11082152.
- [32] F. Florentinus and B. Setiawan, “Pengukuran Kekuatan Kontraksi Otot Pada Bagian Torso Tubuh Menggunakan Sensor Elektromiografi,” *Seminar Nasional Instrumentasi, Kontrol dan Otomasi (SNIKO)*, pp. 10–11, 2018.
- [33] J. X. J. Zhang and K. Hoshino, “Implantable and wearable sensors,” in *Molecular Sensors and Nanodevices*, Elsevier, 2019, pp. 489–545. doi: 10.1016/B978-0-12-814862-4.00008-9.
- [34] Ambu, “Ambu ® Cardiology Sensors Ambu ® electrodes Ambu ® BlueSensor electrodes,” 2022.
- [35] A. López, F. Ferrero, J. R. Villar, and O. Postolache, “High-Performance Analog Front-End (AFE) for EOG Systems,” *Electronics (Basel)*, vol. 9, no. 6, p. 970, Jun. 2020, doi: 10.3390/electronics9060970.

- [36] Analog Devices, “AD620 Datasheet (PDF) - Analog Devices” Accessed: Apr. 27, 2025. [Online]. Available: <https://www.alldatasheet.com/datasheet-pdf/pdf/48090/AD/AD620.html>
- [37] A. A. Bisu, “DESIGN AND SIMULATION OF A 4 TH ORDER HIGH FREQUENCY BANDPASS FILTER FOR RADAR COMMUNICATION SYSTEMS,” 2024. [Online]. Available: <https://nampjournals.org.ng>
- [38] M. Sasaki, M. S. A. Bin Suhaimi, K. Matsushita, S. Ito, and M. I. Rusydi, “Robot Control System Based on Electrooculography and Electromyogram,” *Journal of Computer and Communications*, vol. 03, no. 11, pp. 113–120, 2015, doi: 10.4236/jcc.2015.311018.
- [39] S. N. Hernández Pérez, F. D. Pérez Reynoso, C. A. G. Gutiérrez, M. D. los Á. Cosío León, and R. Ortega Palacios, “EOG Signal Classification with Wavelet and Supervised Learning Algorithms KNN, SVM and DT,” *Sensors*, vol. 23, no. 9, May 2023, doi: 10.3390/s23094553.
- [40] M. I. Rusydi, T. Okamoto, S. Ito, and M. Sasaki, “Rotation Matrix to Operate a Robot Manipulator for 2D Analog Tracking Objects Using Electrooculography,” *Robotics*, vol. 3, pp. 289–309, 2014, doi: 10.3390/robotics3030289.
- [41] M. SASAKI, S. ITO, K. TAKEDA, T. OKAMOTO, and M. I. RUSYDI, “Developing a two-link robot arm controller using voluntary blink.,” *Journal of the Japan Society of Applied Electromagnetics and Mechanics*, vol. 22, no. 4, pp. 475–481, 2014, doi: 10.14243/jsaem.22.475.
- [42] S. N. Abbas, “Eye Blinking EOG Signals as Biometrics,” *Biometric Security and Privacy, Signal Processing for Security Technologies*, 2017, doi: 10.1007/978-3-319-47301-7.
- [43] L. Liu *et al.*, “A Novel Fatigue Driving State Recognition and Warning Method Based on EEG and EOG Signals,” *J Healthc Eng*, vol. 2021, pp. 1–10, Nov. 2021, doi: 10.1155/2021/7799793.
- [44] M. I. Rusydi, M. Bahri, R. S. Ryaldi, F. Akbar, K. Matsuhita, and M. Sasaki, “Recognition of horizontal gaze motion based on electrooculography using tsugeno fuzzy logic,” *IOP Conf Ser Mater Sci Eng*, vol. 602, no. 1, 2019, doi: 10.1088/1757-899X/602/1/012029.
- [45] F. Fityah, “Perancangan Sistem On/Off pada Kursi Roda EOG untuk Menghindari Kesalahan Navigasi Akibat Mata Lelah menggunakan Metode Jaringan Saraf Tiruan,” Universitas Andalas, 2021.
- [46] “USB-6008-Multifunction I/O Device,” ni.com. Accessed: Nov. 27, 2023. [Online]. Available: <https://www.ni.com/en-id/support/model.usb-6008.html>
- [47] TechTeach, “A quick guide to national instruments USB-6008,” TechTeach.com. Accessed: Nov. 27, 2023. [Online]. Available: <http://techt teach.no/tekdok/usb6008/>

- [48] A. F. Bulagang, N. G. Weng, J. Mountstephens, and J. Teo, "A review of recent approaches for emotion classification using electrocardiography and electrodermography signals," Jan. 01, 2020, *Elsevier Ltd.* doi: 10.1016/j.imu.2020.100363.
- [49] G. Ngo, R. Beard, and R. Chandra, "Evolutionary bagging for ensemble learning," Sep. 2022, doi: 10.1016/j.neucom.2022.08.055.
- [50] A. F. Anjani, D. Anggraeni, and I. M. Tirta, "Implementasi Random Forest Menggunakan SMOTE untuk Analisis Sentimen Ulasan Aplikasi Sister for Students UNEJ," *Jurnal Nasional Teknologi dan Sistem Informasi*, vol. 9, no. 2, pp. 163–172, Sep. 2023, doi: 10.25077/teknosi.v9i2.2023.163-172.
- [51] S. Sathyanarayanan, "Confusion Matrix-Based Performance Evaluation Metrics," *African Journal of Biomedical Research*, pp. 4023–4031, Nov. 2024, doi: 10.53555/ajbr.v27i4s.4345.
- [52] G. N. Ahmad *et al.*, "Mixed Machine Learning Approach for Efficient Prediction of Human Heart Disease by Identifying the Numerical and Categorical Features," *Applied Sciences (Switzerland)*, vol. 12, no. 15, Aug. 2022, doi: 10.3390/app12157449.
- [53] S. Amann, S. Proksch, S. Nadi, and M. Mezini, "A Study of Visual Studio Usage in Practice," in *2016 IEEE 23rd International Conference on Software Analysis, Evolution, and Reengineering (SANER)*, IEEE, Mar. 2016, pp. 124–134. doi: 10.1109/SANER.2016.39.



## LAMPIRAN

### Lampiran A. Data Uji dan Data Latih

Link:

<https://docs.google.com/spreadsheets/d/1AhRtPaL0zHPHHuH9yF9dEHEkWBtzRvzw/edit?usp=sharing&ouid=114040571083008088440&rtpof=true&sd=true>



## Lampiran B. Program

### Lampiran B.1 Program Akuisisi Data

```
import nidaqmx
import time
import matplotlib.pyplot as plt
import matplotlib.animation as animation
import os

# Baca dari DAQ Device
def readdaq():
    task = nidaqmx.Task()
        task.ai_channels.add_ai_voltage_chan("Dev1/ai0",
terminal_config=nidaqmx.constants.TerminalConfiguration.RSE, min_val=-5.0,
max_val=5.0)
        task.ai_channels.add_ai_voltage_chan("Dev1/ai1",
terminal_config=nidaqmx.constants.TerminalConfiguration.RSE, min_val=-5.0,
max_val=5.0)
    task.start()
    value = task.read()
    if -0.5 <= value[0] <= 0.5:
        value[0] = 0
    if -0.5 <= value[1] <= 0.5:
        value[1] = 0
    task.stop()
    task.close()
    return value

# Simpan data ke file CSV
def writefiledata(x1, x2):
    save_path = "D:/file"
    os.makedirs(save_path, exist_ok=True)
    completeName = os.path.join(save_path, "kanannngilangfix.csv")
    with open(completeName, "a") as file:
        value1 = str(round(x1, 2))
        value2 = str(round(x2, 2))
        file.write(value1 + "," + value2 + "\n")

# Inialisasi Logging
Ts = 0.01 # Sampling Time [seconds]
N = 100
k = 1
x_len = N # Jumlah titik pada grafik
```

```

Tmin = -6
Tmax = 6
y_range = [Tmin, Tmax]
data = []

# Buat Figure untuk plotting
fig = plt.figure()
ax = fig.add_subplot(1, 1, 1)
xs1 = list(range(0, N))
ys1 = [0] * x_len
xs2 = list(range(0, N))
ys2 = [0] * x_len
ax.set_ylim(y_range)

# Garis kosong yang akan diperbarui
line1 = ax.plot(xs1, ys1, label='CH1')
line2 = ax.plot(xs2, ys2, label='CH2')

# Konfigurasi plot
plt.title('Real-Time Plot')
plt.xlabel('Time [s]')
plt.ylabel('Amplitude [V]')
plt.legend(loc='lower right', labelspace=0.25, handlelength=0.4,
handletextpad=0.7)
plt.grid()

# Fungsi Logging dan Update Grafik
def logging(i, ys1, ys2):
    global k
    value = readdaq()
    print(str(round(value[0], 1)) + "," + str(round(value[1], 1)))
    data.append(value)
    writefiledata(value[0], value[1]) # Simpan ke file
    time.sleep(Ts)

    k = k + 1

    ys1.append(value[0])
    ys2.append(value[1])

    ys1 = ys1[-x_len:]
    ys2 = ys2[-x_len:]

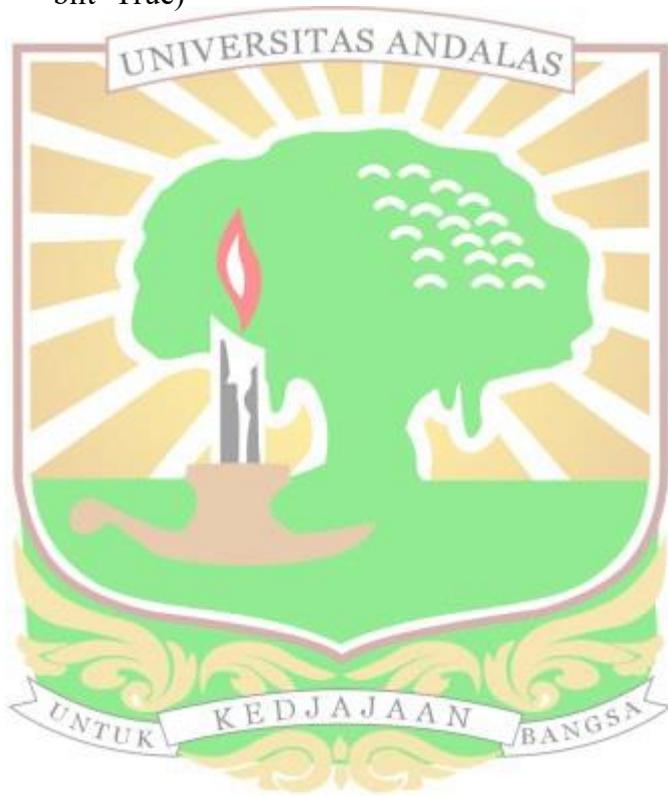
```



```
line.set_ydata(ys1)
line2.set_ydata(ys2)
return line, line2
```

```
# Jalankan animasi real-time
ani = animation.FuncAnimation(fig,
                              logging,
                              frames=len(ys1),
                              fargs=(ys1, ys2),
                              interval=1,
                              blit=True)
```

```
plt.show()
```



## Lampiran B.2 Program Perancangan Model Random Forest

```
from google.colab import files
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score, confusion_matrix,
ConfusionMatrixDisplay, classification_report
from sklearn.model_selection import train_test_split
import matplotlib.pyplot as plt
import pandas as pd
import time
import pickle
# Upload file
uploaded = files.upload()
filename = list(uploaded.keys())[0]
data = pd.read_excel(filename)
# Feature dan label
feature_cols = ['CH1_MAX', 'CH1_MIN', 'CH2_MAX', 'CH2_MIN']
X = data[feature_cols]
y = data['Kelas']
# Split data: 70% train, 30% test
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.30,
random_state=42, stratify=y)
# Training
start_time = time.time()
clf = RandomForestClassifier(n_estimators=100,max_depth=7, random_state=42)
clf.fit(X_train, y_train)
end_time = time.time()
training_time = end_time - start_time
print("\nWaktu Training:", training_time, "detik")
# Prediksi
y_train_pred = clf.predict(X_train)
y_test_pred = clf.predict(X_test)
# Akurasi global
train_accuracy = accuracy_score(y_train, y_train_pred)
test_accuracy = accuracy_score(y_test, y_test_pred)
print("\nAkurasi total data latih:", train_accuracy)
print("Akurasi total data uji:", test_accuracy)
# Akurasi per kelas
print("\n=== Laporan klasifikasi untuk Data Latih ===")
print(classification_report(y_train, y_train_pred))
print("\n=== Laporan klasifikasi untuk Data Uji ===")
print(classification_report(y_test, y_test_pred))
```

### Lampiran B.3 Program GUI dengan *Random Forest*

```
import nidaqmx
import time
import pickle
import pandas as pd
import serial
import threading
import tkinter as tk
from nidaqmx.constants import TerminalConfiguration

# === KONFIGURASI ===
arduino_port = 'COM9'
arduino_baudrate = 9600
ser = serial.Serial(arduino_port, arduino_baudrate)
Ts = 0.1 # Waktu sampling

# === LOAD MODEL ===
with open("D:/PROGRAM TA/RF_modell.pkl", 'rb') as model_file:
    random_forest_model = pickle.load(model_file)

# === VARIABEL GLOBAL ===
kondisi = "mati"
CH1_MAX = CH1_MIN = CH2_MAX = CH2_MIN = 0
thmin = -0.5
thplus = 0.5
k = 0
locked = False

# === DURASI LOCK PER PERINTAH ===
lock_durations = {
    "lirik atas": 3.0,
    "Lirik Atas": 3.0,
    "lirik kanan": 1.5,
    "lirik kiri": 1.5,
    "lirik bawah": 1.5,
    "Kedip Sadar": 0,
    "Kedip Tidak Sadar": 0,
}

# === FUNGSI LOCK ===
```



```

def set_lock(movement):
    global locked
    locked = True
    durasi = lock_durations.get(movement, 1.5)
    print(f"[LOCK] Arah '{movement}' dikunci selama {durasi} detik")
    threading.Timer(durasi, release_lock).start()

def release_lock():
    global locked
    locked = False
    print("[UNLOCK] Kunci dilepas")

# === FUNGSI BACA DAQ ===
def readdaq():
    with nidaqmx.Task() as task1, nidaqmx.Task() as task2:
        task1.ai_channels.add_ai_voltage_chan("Dev1/ai0",
terminal_config=TerminalConfiguration.RSE)
        task2.ai_channels.add_ai_voltage_chan("Dev1/ai1",
terminal_config=TerminalConfiguration.RSE)
        value1 = task1.read()
        value2 = task2.read()
        return value1, value2

# === FUNGSI KIRIM KE ARDUINO ===
def control_motor(movement_text):
    global kondisi
    if movement_text in ["diam", "diam2", "Kedip Tidak Sadar", "kanan selesai",
"kiri selesai", "bawah selesai", "stop"]:
        ser.write(b"")
    elif movement_text in ["Lirik Atas", "lirik atas", "maju"]:
        ser.write(b'maju')
    elif movement_text in ["lirik kanan", "kanan", "kanan terdeteksi"]:
        ser.write(b'kanan')
    elif movement_text in ["lirik kiri", "kiri", "kiri terdeteksi"]:
        ser.write(b'kiri')
    elif movement_text in ["lirik bawah", "mundur"]:
        ser.write(b'mundur')
    elif movement_text in ["Kedip Sadar", "on", "off"]:
        if kondisi == "mati":
            ser.write(b'on')
            kondisi = "hidup"
        else:

```

```

ser.write(b'off')
kondisi = "mati"

# === FUNGSI MANUAL DARI TOMBOL GUI ===
def kirim_perintah_manual(arah):
    global locked
    if not locked:
        movement_text.set(f"{arah}")
        control_motor(arah)
        set_lock(arah)
        writefiledata(k * Ts, 0, 0, arah) # nilai dummy untuk log manual

# === FUNGSI TULIS LOG ===
def writefiledata(t, x1, x2, movement_text):
    with open("tempdata.txt", "a") as file:
        file.write(f"{t:.2f}\t{x1:.3f}\t{x2:.3f}\t{movement_text}\n")

# === LOOP UTAMA (THREAD) ===
def main_loop():
    global k, CH1_MAX, CH1_MIN, CH2_MAX, CH2_MIN
    while True:
        if locked:
            time.sleep(Ts)
            continue

        k += 1
        value1, value2 = readdaq()
        print(f"Data ke- {k} | V1: {value1:.3f}, V2: {value2:.3f}", end=' | ')

# --- DETEKSI GERAKAN ---
if thmin < value1 < thplus and thmin < value2 < thplus:
    movement = "diam"
elif value1 > thplus and value2 < thmin:
    movement = "lirik kanan"
    set_lock(movement)
elif value1 < thmin and value2 > thplus:
    movement = "lirik kiri"
    set_lock(movement)
elif value1 > 1.5 and value2 > 1.5:
    movement = "lirik bawah"
    set_lock(movement)
elif value1 < thmin or value2 < thmin:

```

```

        set_lock(movement)
    else:
        CH1_MAX = max(CH1_MAX, value1)
        CH1_MIN = min(CH1_MIN, value1)
        CH2_MAX = max(CH2_MAX, value2)
        CH2_MIN = min(CH2_MIN, value2)

    input_data = pd.DataFrame([[CH1_MAX, CH1_MIN, CH2_MAX,
CH2_MIN]],
                             columns=["CH1_MAX", "CH1_MIN", "CH2_MAX",
"CH2_MIN"])
    try:
        prediction = random_forest_model.predict(input_data)[0]
    except ValueError as e:
        print(f"Model prediction error: {e}")
        prediction = -1

    if prediction == 1:
        movement = "Kedip Sadar"
        set_lock(movement)
    elif prediction == 2:
        movement = "Kedip Tidak Sadar"
        set_lock(movement)
    elif prediction == 3:
        movement = "Lirik Atas"
        set_lock(movement)
    else:
        movement = "unknown"

    CH1_MAX = CH1_MIN = CH2_MAX = CH2_MIN = 0
    else:
        movement = "noise"

    print(f"Status: {movement}")
    movement_text.set(f"Arah: {movement}")
    control_motor(movement)
    writefiledata(k * Ts, value1, value2, movement)
    time.sleep(Ts)

# === GUI TKINTER ===
root = tk.Tk()

```

```

root.title("Tampilan Arah Kursi Roda")
root.geometry("500x350")
movement_text = tk.StringVar(root)
movement_text.set("Menunggu sinyal...")

label = tk.Label(root, textvariable=movement_text, font=("Helvetica", 20),
fg="blue")
label.pack(pady=20)

# Frame untuk tombol arah
frame_buttons = tk.Frame(root)
frame_buttons.pack(pady=10)

tk.Button(frame_buttons, text="Maju", width=10, command=lambda:
 kirim_perintah_manual("maju")).grid(row=0, column=1, padx=5, pady=5)
tk.Button(frame_buttons, text="Kiri", width=10, command=lambda:
 kirim_perintah_manual("kiri")).grid(row=1, column=0, padx=5, pady=5)
tk.Button(frame_buttons, text="Stop", width=10, command=lambda:
 kirim_perintah_manual("stop")).grid(row=1, column=1, padx=5, pady=5)
tk.Button(frame_buttons, text="Kanan", width=10, command=lambda:
 kirim_perintah_manual("kanan")).grid(row=1, column=2, padx=5, pady=5)
tk.Button(frame_buttons, text="Mundur", width=10, command=lambda:
 kirim_perintah_manual("mundur")).grid(row=2, column=1, padx=5, pady=5)

# Tombol ON/OFF
frame_power = tk.Frame(root)
frame_power.pack(pady=10)
tk.Button(frame_power, text="ON/OFF", width=20, command=lambda:
 kirim_perintah_manual("on" if kondisi == "mati" else "off")).pack()

# Jalankan thread utama
main_thread = threading.Thread(target=main_loop, daemon=True)
main_thread.start()

# Jalankan GUI
root.mainloop()

# Tutup serial saat selesai
ser.close()

```

## Lampiran C Program Kendali Kursi Roda Pada Arduino IDE

```
String inByt;
int RPWM1 = 5;
int LPWM1 = 6;
int RPWM2 = 10;
int LPWM2 = 11;

bool sedangBergerak = false;
String kondisi = "mati";
String gerakanSekarang = "stop";

void setup() {
  pinMode(RPWM1, OUTPUT);
  pinMode(LPWM1, OUTPUT);
  pinMode(RPWM2, OUTPUT);
  pinMode(LPWM2, OUTPUT);

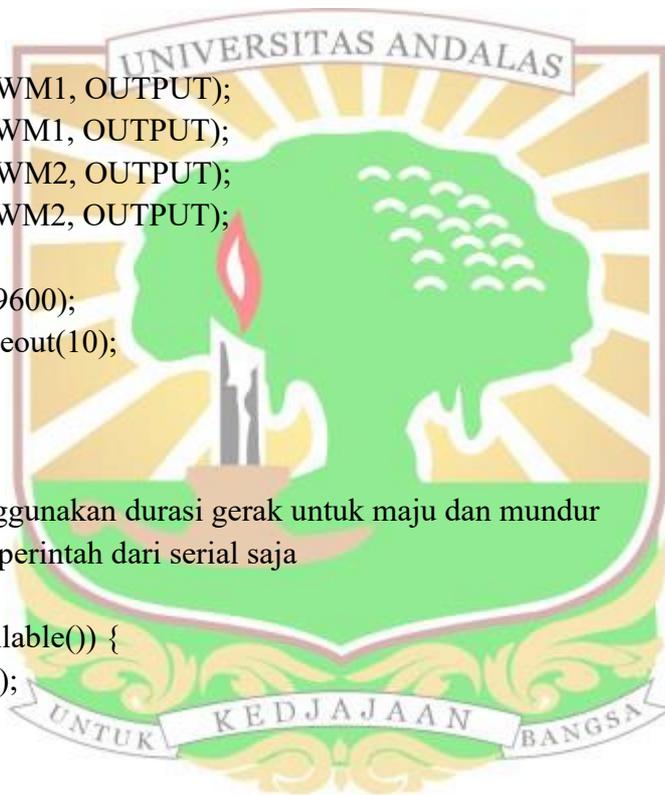
  Serial.begin(9600);
  Serial.setTimeout(10);
}

void loop() {
  // Tidak menggunakan durasi gerak untuk maju dan mundur
  // Hanya cek perintah dari serial saja

  if (Serial.available()) {
    serialEvent();
  }
}

void serialEvent() {
  inByt = Serial.readStringUntil('\n');
  inByt.trim(); // Menghilangkan spasi dan newline

  if (inByt == "on" && kondisi == "mati") {
    kondisi = "hidup";
    stopMotor();
    gerakanSekarang = "stop";
  }
  else if (inByt == "off" && kondisi == "hidup") {
    kondisi = "mati";
  }
}
```



```

stopMotor();
gerakanSekarang = "stop";
}
else if (kondisi == "hidup") {
if (inByt == "stop") {
stopMotor();
gerakanSekarang = "stop";
}
else if (inByt == "maju") {
maju();
gerakanSekarang = "maju";
}
else if (inByt == "mundur") {
mundur();
gerakanSekarang = "mundur";
}
else if (inByt == "kanan") {
kanan();
gerakanSekarang = "kanan";
}
else if (inByt == "kiri") {
kiri();
gerakanSekarang = "kiri";
}
}
}
}

```

```

void maju() {
analogWrite(LPWM1, 0);
analogWrite(RPWM1, 135);
analogWrite(LPWM2, 0);
analogWrite(RPWM2, 135);
}

```

```

void mundur() {
analogWrite(LPWM1, 135);
analogWrite(RPWM1, 0);
analogWrite(LPWM2, 135);
analogWrite(RPWM2, 0);
}

```

```

void kanan() {

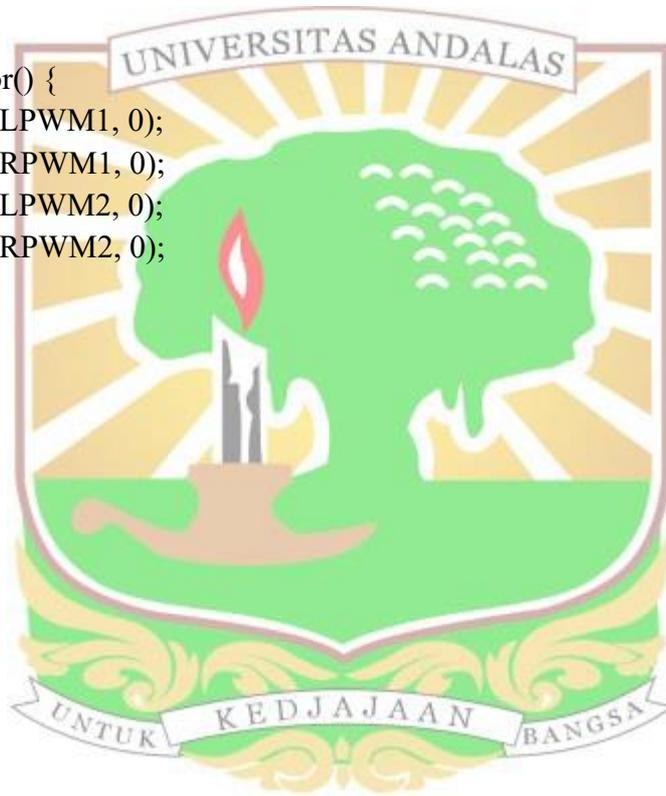
```



```
analogWrite(LPWM1, 135);  
analogWrite(RPWM1, 0);  
analogWrite(LPWM2, 0);  
analogWrite(RPWM2, 135);  
}
```

```
void kiri() {  
  analogWrite(LPWM1, 0);  
  analogWrite(RPWM1, 135);  
  analogWrite(LPWM2, 135);  
  analogWrite(RPWM2, 0);  
}
```

```
void stopMotor() {  
  analogWrite(LPWM1, 0);  
  analogWrite(RPWM1, 0);  
  analogWrite(LPWM2, 0);  
  analogWrite(RPWM2, 0);  
}
```



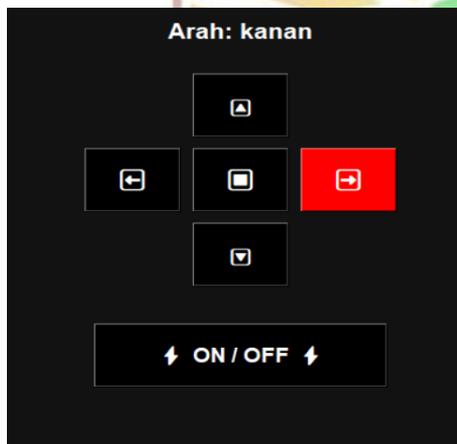
Lampiran D Gambar Tampilan GUI



Gambar D.1 Tampilan GUI Gerak Maju



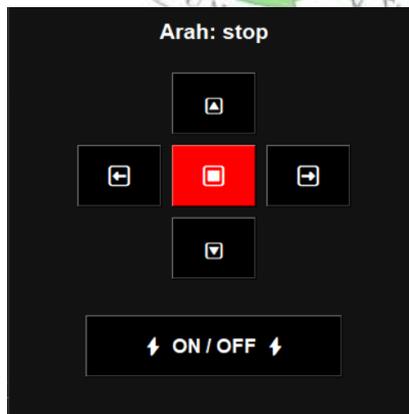
Gambar D.2 Tampilan GUI Gerak Mundur



Gambar D.3 Tampilan GUI Belok Kanan



Gambar D.4 Tampilan GUI Belok Kiri



Gambar D.5 Tampilan GUI Stop



Gambar D.6 Tampilan GUI ON/OFF

