

**SISTEM PEMILAH SAMPAH OTOMATIS
BERBASIS *IMAGE DETECTION*
DAN *IOT***



Disusun Oleh:

N a m a : Jefrian Arya Hernanda
NIM : 21523107

**PROGRAM STUDI INFORMATIKA – PROGRAM SARJANA
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS ISLAM INDONESIA**

2026

HALAMAN PENGESAHAN DOSEN PEMBIMBING

SISTEM PEMILAH SAMPAH OTOMATIS
BERBASIS *IMAGE DETECTION*
DAN *IOT*

TUGAS AKHIR



Yogyakarta, 27 April 2026

Pembimbing,

(Ir. Kurniawan Dwi Irianto, S.T., M.Sc.)

HALAMAN PENGESAHAN DOSEN PENGUJI

SISTEM PEMILAH SAMPAH OTOMATIS

BERBASIS *IMAGE DETECTION*

DAN *IOT*

TUGAS AKHIR

Telah dipertahankan di depan sidang penguji sebagai salah satu syarat untuk memperoleh gelar Sarjana Komputer dari Program Studi Informatika – Program Sarjana di Fakultas Teknologi Industri Universitas Islam Indonesia
Yogyakarta, 13 Mei 2026

Tim Penguji

Ir. Kurniawan Dwi Irianto, S.T., M.Sc.

Anggota 1

Dr. Yudi Prayudi, S.Si., M.Kom.

Anggota 2

Moh. Idris, S.Kom., M.Kom.

Mengetahui,

Ketua Program Studi Informatika – Program Sarjana
Fakultas Teknologi Industri
Universitas Islam Indonesia



(Dhomas Hatta Fudholi, S.T., M.Eng., Ph.D)

HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR

Yang bertanda tangan di bawah ini:

Nama : Jefrian Arya Hernanda

NIM : 21523107

Tugas akhir dengan judul:

SISTEM PEMILAH SAMPAH OTOMATIS BERBASIS *IMAGE DETECTION* DAN *IOT*

Menyatakan bahwa seluruh komponen dan isi dalam tugas akhir ini adalah hasil karya saya sendiri. Apabila di kemudian hari terbukti ada beberapa bagian dari karya ini adalah bukan hasil karya sendiri, tugas akhir yang diajukan sebagai hasil karya sendiri ini siap ditarik kembali dan siap menanggung risiko dan konsekuensi apapun.

Demikian surat pernyataan ini dibuat, semoga dapat dipergunakan sebagaimana mestinya.

Yogyakarta, 25 April 2026



(Jefrian Arya Hernanda)

HALAMAN PERSEMBAHAN

Segala puji dan syukur saya panjatkan ke hadirat Allah SWT atas segala rahmat, karunia, dan kemudahan yang diberikan, sehingga saya dapat menyelesaikan Tugas Akhir ini dengan baik. Tugas akhir ini saya persembahkan sepenuhnya kepada kedua orang tua saya, ayah dan mama tercinta, yang senantiasa memberikan doa, kasih sayang, dukungan, serta pengorbanan yang tiada henti dalam setiap langkah kehidupan saya. Terima kasih atas segala kepercayaan dan kekuatan yang selalu diberikan. Semoga dengan selesainya tugas akhir ini dapat menjadi bentuk sederhana dari rasa terima kasih dan kebanggaan untuk ayah dan mama.

HALAMAN MOTO

“Maka sesungguhnya bersama kesulitan ada kemudahan. Sesungguhnya bersama kesulitan ada kemudahan”

(QS. Al-Insyirah: 5-6)

“Keluhan demi keluhan, segera ku gilas perlahan”

Perunggu – Gemilang

“Remuk rasanya, tapi ada kalanya sebuah lagu menjadi sebuah pelukan.”

KATA PENGANTAR

Puji dan syukur penulis panjatkan ke hadirat Allah SWT yang telah melimpahkan rahmat, karunia, dan hidayah-Nya, sehingga penulis dapat menyelesaikan Tugas Akhir yang berjudul “Sistem Pemilah Sampah Otomatis Berbasis *Image Detection* dan *IoT*” serta menyusun laporan ini dengan baik.

Penyusunan laporan Tugas Akhir ini merupakan salah satu syarat untuk menyelesaikan pendidikan pada Program Studi Informatika, Universitas Islam Indonesia. Dalam proses penyusunan laporan ini, penulis menyadari bahwa banyak tantangan dan kesulitan yang dihadapi, terutama dalam proses pengembangan sistem, pelatihan model deteksi, serta integrasi antara perangkat keras dan perangkat lunak. Namun, berkat dukungan, doa, dan bantuan dari berbagai pihak, laporan ini dapat diselesaikan dengan baik.

Pada kesempatan ini, penulis ingin menyampaikan ucapan terima kasih yang sebesar-besarnya kepada:

1. Allah SWT atas segala kemudahan, kesehatan, dan kelancaran dalam setiap proses penyusunan Tugas Akhir ini.
2. Kedua orang tua saya, ayah dan mama tercinta, yang selalu memberikan doa, kasih sayang, serta dukungan tanpa henti.
3. Bapak Dr. Raden Teduh Dirghayu, S.T., M.Sc. selaku Ketua Jurusan Informatika Universitas Islam Indonesia.
4. Bapak Dhomas Hatta Fudholi, S.T., M.Eng., Ph.D. selaku Ketua Program Studi Informatika Universitas Islam Indonesia.
5. Bapak Ir. Kurniawan Dwi Irianto, S.T., M.Sc. selaku dosen pembimbing, yang telah memberikan arahan, bimbingan, serta masukan yang sangat berharga selama proses pengerjaan Tugas Akhir ini.
6. Seluruh dosen dan staf Program Studi Informatika Universitas Islam Indonesia, yang telah memberikan ilmu pengetahuan serta dukungan selama masa perkuliahan.
7. Diri saya sendiri, yang telah berusaha, bertahan, dan tidak menyerah dalam menyelesaikan Tugas Akhir ini hingga selesai.
8. Adik dan nenek saya tercinta yang senantiasa memberikan doa, kasih sayang, dukungan, serta semangat selama proses perkuliahan dan penyusunan Tugas Akhir ini.
9. Seorang perempuan (AK) yang saya anggap istimewa, yang telah memberikan dukungan, juga perhatiannya untuk saya menyelesaikan Tugas Akhir ini.

10. Teman-teman seperjuangan yang telah memberikan semangat, bantuan, serta kebersamaan selama proses penyusunan Tugas Akhir ini.
11. Staf kebersihan FTI UII yang telah bersedia membantu dan berpartisipasi dalam proses wawancara dan pengujian sistem di lingkungan nyata.

Penulis menyadari bahwa laporan ini masih memiliki kekurangan, baik dari segi penulisan maupun isi. Oleh karena itu, penulis sangat mengharapkan kritik dan saran yang membangun untuk perbaikan di masa yang akan datang.

Akhir kata, penulis berharap semoga laporan Tugas Akhir ini dapat memberikan manfaat bagi pembaca serta dapat menjadi referensi dalam pengembangan sistem pemilah sampah otomatis di masa mendatang.

Yogyakarta, 25 April 2026



(Jefrian Arya Hernanda)

SARI

Permasalahan pengelolaan sampah di Indonesia masih menjadi isu penting karena rendahnya kesadaran masyarakat dalam melakukan pemilahan sampah berdasarkan jenisnya, sehingga menyebabkan sampah tercampur dan menyulitkan proses pengolahan. Penelitian ini bertujuan untuk merancang dan mengimplementasikan sistem pemilah sampah otomatis berbasis image detection dan *Internet of Things* (IoT) menggunakan algoritma YOLOv5s untuk mendeteksi dan mengklasifikasikan sampah ke dalam tiga kategori, yaitu kertas dan tisu, botol plastik, serta kaleng. Model yang telah dilatih kemudian dikonversi ke format TensorFlow Lite (TFLite) agar dapat dijalankan pada Raspberry Pi 3B+ sebagai pengendali utama sistem. Metode penelitian yang digunakan adalah Waterfall yang meliputi tahap analisis kebutuhan, perancangan sistem, implementasi, dan pengujian. Sistem terdiri atas *webcam*, model deteksi objek, motor servo sebagai mekanisme pemilah, sensor ultrasonik untuk monitoring volume tempat sampah, LCD sebagai media informasi, serta Firebase untuk pengiriman data secara *real-time*. Hasil pelatihan model menunjukkan performa yang sangat baik dengan nilai mAP@0.50 sebesar 99% dan mAP@0.50–0.95 sebesar 88,5%. Berdasarkan pengujian object detection, sistem berhasil mendeteksi 14 dari 18 objek yang diuji dengan tingkat keberhasilan sebesar 77,78%. Selain itu, hasil *Hardware Acceptance Testing* (HAT) dan evaluasi kinerja sistem memperoleh nilai keseluruhan sebesar 85,82% yang termasuk dalam kategori sangat baik. Hasil penelitian menunjukkan bahwa sistem mampu melakukan pemilahan sampah secara otomatis serta menyediakan informasi kondisi tempat sampah secara *real-time*.

Kata kunci: pemilah sampah otomatis, YOLOv5s, *Internet of Things*, TensorFlow Lite, Raspberry Pi

GLOSARIUM

<i>Artificial Intelligence (AI)</i>	Teknologi yang memungkinkan sistem komputer untuk meniru kemampuan kecerdasan manusia dalam mengenali pola dan mengambil keputusan.
<i>Confusion Matrix</i>	Tabel evaluasi yang digunakan untuk mengetahui performa model klasifikasi dengan membandingkan hasil prediksi dan label sebenarnya.
<i>Dataset</i>	Kumpulan data berupa citra yang digunakan untuk proses pelatihan dan pengujian model.
<i>Deep Learning</i>	cabang dari machine learning yang menggunakan jaringan saraf tiruan untuk mempelajari representasi data secara otomatis.
Firestore	Platform layanan backend berbasis cloud yang digunakan untuk menyimpan, mengelola, dan mengirim data secara real-time melalui internet.
GPIO	Pin pada Raspberry Pi yang digunakan untuk menghubungkan dan mengontrol perangkat eksternal seperti sensor dan aktuator.
<i>Internet of Things (IoT)</i>	Konsep di mana perangkat fisik dapat terhubung dan berkomunikasi melalui jaringan internet.
<i>LabelImg</i>	Perangkat lunak yang digunakan untuk melakukan anotasi data citra pada proses pembuatan dataset.
LCD (<i>Liquid Crystal Display</i>)	Perangkat tampilan yang digunakan untuk menampilkan informasi hasil deteksi dan status sistem kepada pengguna.
<i>Mean Average Precision (mAP)</i>	Metrik evaluasi yang digunakan untuk mengukur performa model deteksi objek berdasarkan rata-rata nilai precision pada berbagai threshold.
<i>Model Training</i>	Proses melatih model deep learning menggunakan dataset agar dapat mengenali pola dan melakukan prediksi.

<i>Object Detection</i>	Metode dalam pengolahan citra yang digunakan untuk mendeteksi dan mengidentifikasi objek dalam sebuah gambar atau video.
<i>OpenCV</i>	Library open source yang digunakan untuk pengolahan citra dan video dalam berbagai aplikasi computer vision.
<i>Precision</i>	Metrik yang menunjukkan tingkat ketepatan model dalam mendeteksi objek yang benar dari seluruh prediksi yang dilakukan.
Raspberry Pi	Komputer mini berbasis single-board yang digunakan sebagai pengendali utama dalam sistem tertanam.
<i>Real-time</i>	Kemampuan sistem untuk memproses data dan memberikan hasil secara langsung tanpa jeda waktu yang signifikan.
<i>Recall</i>	Metrik yang menunjukkan kemampuan model dalam mendeteksi seluruh objek yang seharusnya terdeteksi.
Servo Motor	Aktuator untuk menggerakkan mekanisme pemilahan sampah berdasarkan hasil klasifikasi.
<i>TensorFlow Lite (TFLite)</i>	Versi ringan dari TensorFlow yang digunakan untuk menjalankan model machine learning pada perangkat dengan sumber daya terbatas.
Ultrasonic Sensor	Sensor yang digunakan untuk mengukur jarak dengan memanfaatkan gelombang ultrasonik, dalam penelitian ini digunakan untuk mengukur volume sampah.
Waterfall	Metode pengembangan perangkat lunak yang dilakukan secara bertahap dan berurutan mulai dari analisis kebutuhan hingga pengujian sistem.
YOLOv5s	Algoritma deteksi objek berbasis deep learning yang memiliki ukuran model ringan dan kecepatan tinggi, sehingga cocok untuk aplikasi real-time.

DAFTAR ISI

HALAMAN JUDUL	i
HALAMAN PENGESAHAN DOSEN PEMBIMBING	ii
HALAMAN PENGESAHAN DOSEN PENGUJI	iii
HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR.....	iv
HALAMAN PERSEMBAHAN.....	v
HALAMAN MOTO.....	vi
KATA PENGANTAR.....	vii
SARI	ix
GLOSARIUM	x
DAFTAR ISI	xii
DAFTAR TABEL	xiv
DAFTAR GAMBAR	xv
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	3
1.3 Batasan Masalah	3
1.4 Tujuan Penelitian	3
1.5 Manfaat Penelitian	4
1.6 Metodologi Penelitian	4
1.7 Sistematika Penulisan	5
BAB II TINJAUAN PUSTAKA	6
2.1 Sampah.....	6
2.2 Pemilahan Sampah.....	6
2.3 <i>Object Detection</i> dalam Pemilahan Sampah	7
2.4 <i>Internet of Things (IoT)</i>	8
2.5 Raspberry Pi 3B+	9
2.6 Open Source Computer Vision Library (OpenCV)	9
2.7 You Only Look Once (YOLO).....	10
2.8 TensorFlow	11
2.9 Google Colaboratory (Google Colab).....	12
2.10 Webcam.....	12
2.11 PCA9685 PWM Driver	13
2.12 Motor Servo (MG996R).....	13
2.13 Sensor Ultrasonik (HC-SR04)	14
2.14 Firebase	14
2.15 LabelImg	15
2.16 Library Adafruit	15
2.17 Library GPIO	16
2.18 <i>Black Box Testing</i>	16
2.19 Penelitian Terdahulu	17
BAB III METODOLOGI PENELITIAN	21

3.1	<i>Requirements Definition</i> (Analisis Kebutuhan)	22
3.1.1	Kebutuhan Fungsional.....	23
3.1.2	Kebutuhan Perangkat Keras	24
3.1.3	Kebutuhan Perangkat Lunak	24
3.2	Pengumpulan dan Pengolahan <i>Dataset</i>	25
3.2.1	Kelas <i>Dataset</i>	25
3.2.2	Pengmpulan <i>Dataset</i>	26
3.2.3	Anotasi <i>Dataset</i>	27
3.2.4	<i>Dataset Split</i>	29
3.3	<i>System and Software Design</i> (Perancangan Sistem dan Perangkat Lunak)	29
3.3.1	Diagram Blok Sistem	29
3.3.2	Diagram Blok <i>Object Detection</i>	30
3.3.3	<i>Flowchart</i> Sistem	31
3.3.4	<i>Flowchart</i> Proses Pengukuran Volume Bak Sampah.....	33
3.4	Perancangan Perangkat Keras	34
3.4.1	Rangkaian Sistem Perangkat Keras.....	34
3.4.2	Konfigurasi Pin dan Koneksi Antar Perangkat	35
3.5	Perancangan dan Pelatihan Model	36
3.6	<i>Implementation and Unit Testing</i> (Implementasi dan Pengujian unit)	40
3.7	<i>Integration and System Testing</i> (Integrasi dan Pengujian Sistem)	41
3.7.1	<i>Hardware Acceptance Testing</i> (HAT)	41
3.7.2	Evaluasi Kinerja dan Kelayakan Sistem	41
BAB IV	PEMBAHASAN	43
4.1	Hasil Training Model YOLOv5s	43
4.1.1	Analisis <i>Overfitting Model</i>	47
4.2	Implementasi Sistem	48
4.3	Pengujian Unit.....	58
4.3.1	Uji Fungsi <i>Object Detection</i>	59
4.3.2	Uji LCD 16x2 I2C	61
4.3.3	Uji Respon Servo.....	62
4.3.4	Uji Pengukuran Volume Bak Sampah.....	63
4.3.5	Uji Pengiriman Data (Firebase).....	64
4.4	Pengujian Sistem.....	65
4.5	Pembahasan dan Evaluasi	68
4.5.1	Evaluasi Model Deteksi YOLOv5s.....	68
4.5.2	Evaluasi Implementasi Sistem dan Pengujian Unit.....	69
4.5.3	Evaluasi Pengujian Sistem Secara Keseluruhan	70
4.5.4	Kelebihan dan Keterbatasan Sistem	71
BAB V	KESIMPULAN DAN SARAN	72
5.1	Kesimpulan	72
5.2	Saran.....	73
	DAFTAR PUSTAKA.....	74
	LAMPIRAN	77

DAFTAR TABEL

Tabel 3.1 Kebutuhan Perangkat Keras.....	24
Tabel 3.2 Kebutuhan Perangkat Lunak.....	24
Tabel 3.3 Kelas <i>Dataset</i>	25
Tabel 3.4 Koneksi Pin PCA9685 (Servo Driver).....	35
Tabel 3.5 Koneksi Pin Servo MG996R	35
Tabel 3.6 Koneksi Pin LCD 16x2 I2C.....	35
Tabel 3.7 Koneksi Pin Sensor Ultrasonik	35
Tabel 3.8 Koneksi Kamera (Webcam).....	36
Tabel 3.9 Instrumen Evaluasi	42
Tabel 4.1 Skenario Black Box Testing	58
Tabel 4.2 Pengujian Object Detection	60
Tabel 4.3 Pengujian Respon Servo	62
Tabel 4.4 Pengujian Volume Bak Sampah	63
Tabel 4.5 Pengujian Hardware Acceptance Testing	66
Tabel 4.6 Data Responden	67

DAFTAR GAMBAR

Gambar 2.1 Raspberry Pi 3 B+	9
Gambar 2.2 TensorFlow	11
Gambar 2.3 Google Colaboratory	12
Gambar 2.4 Webcam	12
Gambar 2.5 PCA9685	13
Gambar 2.6 Motor Servo MG996R	13
Gambar 2.7 Sensor Ultrasonik HC-SR04	14
Gambar 3.1 Tahapan Metode Waterfall Menurut Sommerville	21
Gambar 3.2 Pengumpulan Citra Dataset Kertas dan Tisu	26
Gambar 3.3 Pengumpulan Citra Dataset Botol Plastik	26
Gambar 3.4 Pengumpulan Citra Dataset Kaleng	27
Gambar 3.5 Jumlah Citra Keseluruhan	27
Gambar 3.6 Proses Anotasi Dataset Menggunakan tools labelImg	28
Gambar 3.7 Hasil Anotasi Gambar	28
Gambar 3.8 Dataset Split	29
Gambar 3.9 Diagram Blok Sistem	29
Gambar 3.10 Diagram Blok Object Detection	30
Gambar 3.11 Flowchart Sistem	32
Gambar 3.12 Flowchart Proses Pengukuran Volume Bak Sampah	33
Gambar 3.13 Rangkaian Perangkat Keras	34
Gambar 3.14 Perintah Environment Setup YOLOv5s	36
Gambar 3.15 Isi File Requirements Environment Setup YOLOv5s	38
Gambar 3.16 Struktur Folder Dataset	38
Gambar 3.17 Konfigurasi data.yaml	39
Gambar 3.18 Perintah Memulai Proses Pelatihan YOLOv5s	39
Gambar 3.19 Perintah Konversi YOLOv5s ke TensorFlow Lite	40
Gambar 4.1 Hasil Proses Training Model YOLOv5s	43
Gambar 4.2 <i>Confusion Matrix</i> Hasil Pengujian Model YOLOv5s	44
Gambar 4.3 Hasil <i>Precision-Recall Curve</i>	45
Gambar 4.4 Hasil <i>F1-Confidence Curve</i>	46
Gambar 4.5 <i>Training Results</i>	47
Gambar 4.6 Implementasi Rancangan Rangkaian Perangkat Keras	49

Gambar 4.7 Penempatan Perangkat Keras pada Kotak	49
Gambar 4.8 Kondisi Sistem Setengah Jadi	50
Gambar 4.9 Kondisi Sistem yang Telah Siap	51
Gambar 4.10 Fungsi <i>Library</i> Pendukung.....	52
Gambar 4.11 Fungsi <i>Image Detection</i>	53
Gambar 4.12 Fungsi Ukur Volume Bak Sampah	54
Gambar 4.13 Fungsi Penggerak Pemilah Sampah (Motor Servo).....	55
Gambar 4.14 Fungsi Informasi Hasil Deteksi.....	56
Gambar 4.15 Fungsi Inisialisasi Firebase	56
Gambar 4.16 Fungsi Program Utama	57
Gambar 4.17 Sampah yang Terdeteksi dan Akurasi-nya.....	59
Gambar 4.18 Pengujian LCD ketika Sistem Berhasil Mendeteksi Sampah	61
Gambar 4.19 Pengujian LCD ketika Tidak Ada Objek yang Terdeteksi Sistem.....	62
Gambar 4.20 Data yang Terkirim ke Firebase	64
Gambar 4.21 Lokasi Penempatan Sistem di <i>Hall</i> FTI UII.....	65
Gambar 4.22 Pengujian yang Berlangsung di <i>Hall</i> FTI UII.....	66
Gambar 4.23 FPS Model yang didapat	68

BAB I

PENDAHULUAN

1.1 Latar Belakang

Masalah sampah masih menjadi salah satu persoalan lingkungan paling krusial di Indonesia. Sampah merupakan sisa dari aktivitas manusia maupun alam yang dibuang karena dianggap tidak memiliki nilai guna (Dewi, 2021). Seiring bertambahnya jumlah penduduk dan meningkatnya aktivitas manusia, volume sampah yang dihasilkan pun terus meningkat setiap harinya. Sayangnya, peningkatan volume sampah ini tidak diiringi dengan kesadaran masyarakat untuk memilah dan membuang sampah dengan benar. Banyak masyarakat yang masih belum membiasakan diri memisahkan sampah sesuai jenisnya yang justru berujung pada penumpukan limbah dan pencemaran lingkungan.

Kebiasaan mencampur berbagai jenis sampah dalam satu tempat tidak hanya menyulitkan proses daur ulang, tetapi juga menyebabkan penumpukan sampah di tempat pembuangan sementara. Bila tidak ditangani dengan baik, kondisi ini dapat menimbulkan gangguan kesehatan dan pencemaran lingkungan (Rahman, 2021). Beberapa tempat biasanya telah menyediakan fasilitas tempat sampah berdasarkan kategori jenisnya, tetapi tingkat kesadaran dan partisipasi masyarakat dalam menggunakannya masih sangat minim (Sarifa et al., 2025). Selain itu, masalah kebersihan dan kenyamanan fasilitas tempat sampah juga menjadi penghambat, karena banyak tempat sampah yang dinilai tidak higienis, mengeluarkan bau tidak sedap, dan mengharuskan kontak langsung dengan tutup tempat sampah (Rahman, 2021).

Permasalahan tersebut tidak hanya terjadi di ruang publik, tetapi juga ditemukan di lingkungan pendidikan, termasuk kampus. Sebagai tempat pembelajaran dan pusat inovasi, kampus seharusnya menjadi contoh dalam menciptakan lingkungan yang bersih dan berkelanjutan. Namun, kenyataannya masih banyak kampus yang belum memiliki sistem pengelolaan sampah yang efektif, terutama dalam hal pemilahan sampah. Mahasiswa cenderung belum menjadikan kebiasaan memilah sampah sebagai bagian dari rutinitas harian mereka. Hal ini menunjukkan bahwa kesadaran akan pentingnya pemilahan sampah sejak dari sumber masih belum terbentuk dengan baik di lingkungan pendidikan.

Untuk itu, peran kampus dan mahasiswa menjadi sangat penting dalam mendorong perubahan perilaku menuju pengelolaan sampah yang lebih baik. Kampus yang mampu memberikan pengaruh positif kepada sivitas akademika dan masyarakat sekitar akan menciptakan ekosistem yang mendukung kepedulian terhadap lingkungan (Sasono, 2024). Mahasiswa pun berpotensi menjadi agen perubahan yang dapat membawa pengaruh baik, tidak hanya selama masa studi, tetapi juga setelah lulus dan berkontribusi langsung di masyarakat, termasuk dalam keluarga mereka sendiri (Sasono, 2024). Upaya strategis seperti pelibatan dalam kegiatan sosial, penyusunan program keberlanjutan yang spesifik, serta partisipasi dalam penilaian kampus berkelanjutan merupakan langkah konkret untuk membentuk budaya sadar lingkungan yang lebih kuat (Sasono, 2024).

Seiring berkembangnya teknologi digital, berbagai perangkat fisik kini dapat saling terhubung melalui internet dan bertukar data secara otomatis. Konsep ini dikenal sebagai *Internet of Things (IoT)*, yaitu sistem yang memungkinkan perangkat seperti mesin, sensor, dan alat elektronik saling berkomunikasi dan dikendalikan jarak jauh melalui jaringan internet. *IoT* memanfaatkan kombinasi perangkat keras, perangkat lunak, dan layanan untuk mengumpulkan dan berbagi data antar perangkat, sehingga mampu mendukung berbagai sistem otomatisasi seperti pengawasan lingkungan, pemantauan fasilitas, hingga sistem pemilah sampah (Nahdi & Dhika, 2021). *IoT* juga dikenal sebagai teknologi yang dapat mengurangi ketergantungan pada tenaga manusia dan memungkinkan kontrol perangkat dari jarak jauh tanpa perlu interaksi langsung (Nahdi & Dhika, 2021).

Melihat kompleksitas permasalahan pengelolaan sampah serta masih rendahnya kesadaran masyarakat dalam memilah sampah sejak dari sumbernya, dibutuhkan sebuah sistem yang tidak hanya berfungsi secara teknis, tetapi juga mampu membentuk perilaku baru yang lebih peduli terhadap lingkungan. Salah satu solusi yang dapat diterapkan adalah sistem pemilah sampah otomatis berbasis *image detection* dan *Internet of Things (IoT)*. Sistem ini mampu mengidentifikasi jenis sampah melalui kamera, kemudian secara otomatis mengarahkan sampah ke bak yang sesuai. Selain itu, teknologi *IoT* dimanfaatkan untuk memantau kondisi sistem secara jarak jauh, seperti informasi volume sampah pada setiap bak. Dengan adanya sistem ini, proses pemilahan sampah diharapkan dapat dilakukan secara lebih efektif, efisien, dan mendukung upaya pengelolaan lingkungan yang berkelanjutan.

1.2 Rumusan Masalah

Berdasarkan penjelasan latar belakang di atas, dapat dirumuskan beberapa masalah sebagai berikut:

- a. Bagaimana merancang sistem pemilah sampah otomatis berbasis *image detection* dan *Internet of Things (IoT)*?
- b. Bagaimana sistem *image detection* dapat digunakan untuk mengidentifikasi dan mengklasifikasikan jenis sampah secara otomatis?
- c. Bagaimana sistem dapat menampilkan informasi volume sampah secara *real-time*?

1.3 Batasan Masalah

Untuk menghindari pembahasan masalah yang melebar, diperlukan batasan masalah dan lingkup penelitian sebagai berikut:

- a. Mikrokontroler yang digunakan adalah Raspberry Pi 3 Model B+.
- b. Metode deteksi objek yang digunakan adalah YOLOv5s yang dikonversi ke TensorFlow Lite.
- c. Sistem mengidentifikasi jenis sampah kertas dan tisu, botol plastik, serta kaleng.
- d. *Dataset* yang digunakan merupakan dataset yang dianotasi secara mandiri oleh peneliti.
- e. Sampah yang dapat diidentifikasi merupakan sampah ringan.
- f. Sistem hanya bekerja pada kondisi pencahayaan yang cukup dan tidak diuji pada kondisi cahaya ekstrem.
- g. Tidak dapat mengidentifikasi sampah dengan jumlah banyak secara bersamaan dalam satu waktu.

1.4 Tujuan Penelitian

Berdasarkan rumusan masalah yang telah dijelaskan sebelumnya, penelitian ini memiliki beberapa tujuan yang ingin dicapai, yaitu:

- a. Merancang sistem pemilah sampah otomatis berbasis *image detection* dan *Internet of Things (IoT)*.
- b. Mengimplementasikan teknologi *image detection* untuk mengidentifikasi dan mengklasifikasikan jenis sampah secara otomatis.
- c. Mengintegrasikan sistem pemantauan *real-time* terhadap volume sampah menggunakan teknologi *IoT*.

1.5 Manfaat Penelitian

Melalui pengembangan sistem pemilah sampah otomatis berbasis *image detection* dan IoT, manfaat dari penelitian ini antara lain:

- a. Memudahkan proses pemilahan jenis sampah yang dilakukan secara otomatis.
- b. Pemantauan volume tempat sampah secara *real-time* sehingga meningkatkan efisiensi.
- c. Menerapkan peran teknologi untuk solusi permasalahan lingkungan.

1.6 Metodologi Penelitian

Metodologi penelitian yang digunakan dalam penelitian ini adalah waterfall. Metode ini dipilih agar proses pengembangan sistem dapat dilakukan secara sistematis, mulai dari identifikasi kebutuhan hingga pengujian sistem secara menyeluruh.

a. *Requirements Definition*

Tahap *requirements definition* merupakan tahap awal dalam pengembangan sistem yang bertujuan untuk menentukan kebutuhan sistem secara jelas.

b. *System and Software Design*

Tahap *system and software design* bertujuan untuk merancang sistem berdasarkan kebutuhan yang telah ditentukan sebelumnya. Perancangan meliputi arsitektur sistem, alur kerja, serta desain perangkat lunak dan perangkat keras yang digunakan.

c. *Implementation and Unit Testing*

Tahap *implementation and unit testing* merupakan tahap penerapan hasil perancangan ke dalam bentuk sistem yang nyata. Setelah implementasi, setiap bagian atau unit sistem diuji secara terpisah untuk memastikan bahwa masing-masing fungsi dapat berjalan dengan baik sesuai perancangan.

d. *Integration and System Testing*

Tahap *integration and system testing* dilakukan dengan menggabungkan seluruh unit sistem yang telah diuji sebelumnya menjadi satu kesatuan sistem. Setelah proses integrasi, dilakukan pengujian sistem secara menyeluruh untuk memastikan bahwa seluruh komponen dapat bekerja dan sistem mampu memenuhi kebutuhan pengguna sesuai dengan tujuan penelitian.

1.7 Sistematika Penulisan

Sistematika penulisan laporan tugas akhir ini disusun untuk memberikan gambaran yang jelas mengenai alur pembahasan pada setiap bab. Penulisan skripsi terdiri dari beberapa bab yang saling berkaitan dan disusun secara sistematis sebagai berikut:

BAB I PENDAHULUAN

Bab ini berisi latar belakang penelitian, rumusan masalah, batasan masalah, tujuan dan manfaat penelitian, metodologi penelitian, serta sistematika penulisan.

BAB II TINJAUAN PUSTAKA

Bab ini membahas teori-teori pendukung dan hasil penelitian terdahulu yang berkaitan dengan pengembangan sistem pemilah sampah otomatis dan pengolahan citra.

BAB III METODOLOGI PENELITIAN

Bab ini menjelaskan tahapan penelitian, analisis kebutuhan, pengumpulan dan pengolahan dataset, perancangan sistem, perancangan perangkat keras dan perangkat lunak, serta perancangan dan pelatihan model deteksi objek.

BAB IV HASIL DAN PEMBAHASAN

Bab ini memaparkan hasil implementasi sistem serta pengujian unit dan pengujian sistem secara keseluruhan, termasuk evaluasi kinerja model deteksi.

BAB V KESIMPULAN DAN SARAN

Bab ini berisi kesimpulan dari hasil penelitian serta saran untuk pengembangan sistem di masa mendatang.

BAB II

TINJAUAN PUSTAKA

2.1 Sampah

Sampah merupakan sisa dari aktivitas manusia yang sudah tidak digunakan lagi dan dianggap tidak memiliki nilai guna oleh pemiliknya. Sampah dapat berbentuk padat, cair, atau gas tergantung dari jenis aktivitas yang menghasilkan (Hasibuan, 2023). Seiring meningkatnya jumlah penduduk dan aktivitas konsumsi, volume sampah terus bertambah dan menjadi masalah serius di berbagai daerah (Sholihah, 2020). Masalah ini tidak hanya berkaitan dengan jumlah timbulan sampah, tetapi juga cara pengelolaannya yang belum optimal.

Menurut Hasibuan (2023), sampah dibedakan menjadi tiga kategori, yaitu:

- a. Sampah organik adalah jenis sampah yang berasal dari bahan alami seperti sisa makanan dan daun kering yang dapat terurai secara alami. Namun, jika tidak ditangani dengan benar, sampah organik dapat menimbulkan bau tidak sedap, serta menjadi sumber penyakit akibat proses pembusukan yang cepat.
- b. Sampah anorganik mencakup bahan seperti plastik, logam, dan kaca yang sulit terurai secara alami dan membutuhkan proses daur ulang atau pengolahan khusus.
- c. Sampah berbahaya meliputi limbah seperti baterai, cat, dan limbah medis yang mengandung zat beracun dan harus ditangani dengan cara yang sangat hati-hati.

Pengelolaan sampah yang buruk dapat menimbulkan berbagai dampak negatif terhadap lingkungan, seperti pencemaran udara, tanah, dan air akibat akumulasi limbah yang tidak terurai (Khoiriyah, 2021). Selain itu, sampah yang menumpuk di tempat umum dapat mengganggu kenyamanan dan merusak pemandangan. Kondisi ini dapat menyebabkan munculnya serangga tikus, atau hewan lain pembawa penyakit dan menurunkan kualitas hidup masyarakat (Utami et al., 2023). Pada lingkungan padat penduduk, penumpukan sampah juga dapat menimbulkan bau tidak sedap dan berisiko menyebarkan penyakit menular.

2.2 Pemilahan Sampah

Pemilahan sampah adalah proses memisahkan sampah berdasarkan jenis dan karakteristiknya sebelum dilakukan pengolahan lebih lanjut. Tujuan utama dari pemilahan ini adalah untuk memudahkan proses daur ulang, mengurangi volume sampah yang dibuang ke tempat pembuangan akhir (TPA), serta meminimalkan dampak negatif terhadap lingkungan.

Sayangnya, kesadaran masyarakat dalam memilah sampah masih tergolong rendah. Banyak masyarakat yang belum membiasakan diri untuk memisahkan sampah berdasarkan jenisnya, sehingga sampah yang seharusnya bisa didaur ulang menjadi tercampur dan sulit diolah kembali (Khoiriyah, 2021). Kebiasaan membuang sampah sembarangan dan mencampur berbagai jenis limbah juga menjadi kendala dalam pengelolaan sampah yang berkelanjutan (Khoiriyah, 2021).

Pemilahan sampah seharusnya dilakukan sejak dari sumbernya, yaitu rumah tangga, tempat kerja, sekolah, dan tempat umum lainnya. Di beberapa wilayah, pemilahan sudah mulai dilakukan dengan menggunakan tempat sampah yang diberi kode warna sesuai jenis sampah. Namun, penerapannya masih belum merata dan kesadaran masyarakat masih menjadi tantangan utama.

Ketiadaan sistem pemilahan yang efektif menyebabkan peningkatan jumlah sampah yang menumpuk di TPA. Dengan demikian, pemilahan sampah merupakan langkah awal yang krusial dalam sistem pengelolaan sampah. Penerapan teknologi seperti sistem pemilah sampah otomatis menjadi solusi yang relevan untuk mengatasi lemahnya praktik pemilahan secara manual.

2.3 Object Detection dalam Pemilahan Sampah

Object detection adalah salah satu teknik utama dalam bidang *computer vision* yang digunakan untuk menemukan serta mengenali keberadaan objek tertentu dalam sebuah gambar atau video. Teknologi ini memungkinkan sistem untuk tidak hanya mendeteksi posisi suatu objek, tetapi juga mengidentifikasinya secara otomatis berdasarkan kategori tertentu seperti wajah, kendaraan, atau benda lainnya (Fauzi et al., 2020).

Dalam konteks sistem pemilah sampah, *object detection* memegang peran penting untuk mengklasifikasikan berbagai jenis sampah berdasarkan citra visualnya. Proses ini membantu mempercepat dan mempermudah pemilahan antara berbagai jenis sampah tanpa campur tangan manusia secara langsung (Anggara et al., 2025).

Teknologi ini telah banyak dimanfaatkan dalam berbagai bidang, mulai dari pengawasan lalu lintas hingga sistem keamanan, karena kemampuannya dalam mengenali objek di lingkungan yang kompleks (Fauzi et al., 2020). Dalam implementasi pada sistem pemilah sampah, penggunaan *object detection* secara signifikan dapat meningkatkan efisiensi pengolahan sampah dengan hasil klasifikasi yang lebih akurat.

2.4 *Internet of Things (IoT)*

Internet of Things (IoT) adalah sebuah konsep teknologi di mana benda-benda fisik seperti peralatan rumah tangga, kendaraan, sensor, dan perangkat lainnya dapat saling terhubung dan berkomunikasi melalui jaringan internet. Melalui koneksi ini, objek-objek tersebut dapat saling bertukar data secara otomatis, tanpa memerlukan interaksi langsung dari manusia. Tujuan utama dari *IoT* adalah untuk mempermudah aktivitas manusia sehari-hari dengan sistem yang lebih efisien dan responsif terhadap lingkungan.

Istilah *IoT* pertama kali dikenalkan oleh Kevin Ashton pada tahun 1999 melalui pengembangan teknologi *Radio Frequency Identification (RFID)*, yang memungkinkan benda fisik memiliki identitas unik dalam jaringan (M. Fernando et al., 2022). Teknologi *RFID* inilah yang menjadi landasan awal terbentuknya ide bahwa setiap objek dapat memiliki alamat *Internet Protocol (IP)* untuk bisa saling berkomunikasi secara digital (M. Fernando et al., 2022).

IoT tidak terbatas hanya pada komputer dan *smartphone*, tetapi juga mencakup berbagai perangkat seperti kamera, sensor suhu, hingga perangkat elektronik rumah yang dapat dihubungkan dengan sistem kendali otomatis (Khairnar et al., 2021). Semua perangkat ini bisa dikendalikan dari jarak jauh selama keduanya memiliki koneksi internet yang stabil (Khairnar et al., 2021). Dengan memanfaatkan teknologi tertanam (*embedded*), perangkat-perangkat ini dapat menjalankan fungsinya secara otomatis berdasarkan perintah dari pusat kontrol, seperti Raspberry Pi dalam sistem otomasi (Khairnar et al., 2021).

Dalam proses kerjanya, *IoT* memanfaatkan pemrograman dan koneksi jaringan untuk memungkinkan perangkat mengambil keputusan sendiri berdasarkan data yang diterima. Setelah suatu benda diberi *IP address* dan terhubung ke internet, ia bisa mengakses informasi, mengolahnya, lalu mengeksekusi fungsi tertentu misalnya mengaktifkan motor, mengubah posisi, atau mengirim data ke sistem lain. Sistem ini berjalan secara otomatis tanpa harus dikendalikan manual oleh manusia (Selay et al., 2022).

IoT memberikan banyak manfaat dalam kehidupan sehari-hari, seperti memantau kondisi lingkungan, mengatur peralatan rumah secara otomatis, dan mengelola sistem industri dengan lebih efisien. Dalam konteks sistem pemilah sampah otomatis, *IoT* dapat digunakan untuk menghubungkan kamera pendeteksi sampah dengan mikrokontroler seperti Raspberry Pi yang kemudian mengontrol aktuator atau motor servo untuk memindahkan sampah ke tempat yang sesuai secara *real-time*.

2.5 Raspberry Pi 3B+



Gambar 2.1 Raspberry Pi 3 B+
Sumber: <https://core-electronics.com.au>

Raspberry Pi adalah sebuah *single-board computer* yang berfungsi sebagai perangkat komputasi berukuran kecil dengan konsumsi daya rendah, yang banyak digunakan dalam pengembangan sistem *embedded*, *Internet of Things (IoT)*, dan aplikasi berbasis komputasi lainnya. Pada pengembangan ini menggunakan model Raspberry Pi 3 model B+.

Tabel 2.1 Spesifikasi Raspberry Pi 3B+

Spesifikasi	Keterangan
CPU Processor	64-bit quad-core ARMv8 Cortex-A53
CPU Broadcom	BCM2837 B0
CPU Cores	4
CPU Clock Frequency	1400 MHz
RAM	1024 MB LPDDR2 SDRAM
GPU	400 MHz Video Core IV
Wi-Fi	2.4 & 5 GHz IEEE 802.11/bg/n/ac
Bluetooth	BLE 4.2
Storage	MicroSD card
Network Adaptor	100 MBps Ethernet
USB Port	4 USB-2.0
Power Supply	5V 2.5A

Sumber: (Papakyriakou & Barbounakis, 2023)

2.6 Open Source Computer Vision Library (OpenCV)

Open Source Computer Vision Library (OpenCV) adalah *library open source* yang dirancang khusus untuk mempermudah pemrograman terkait pengolahan citra digital (Wibysono et al., 2022). *library* ini awalnya dikembangkan oleh Intel dan ditujukan agar komputer memiliki kemampuan memproses visual yang menyerupai cara kerja penglihatan manusia (Zulfachmi et al., 2024).

OpenCV menyediakan berbagai algoritma dan fungsi yang dapat digunakan untuk memproses gambar maupun video, mulai dari tingkat dasar (*low-level API*) hingga tingkat

lanjut (*high-level API*) (Wibysono et al., 2022). Beberapa fitur yang tersedia di dalamnya meliputi pengenalan wajah, pelacakan wajah, deteksi objek, Kalman *filtering*, metode kecerdasan buatan (*Artificial Intelligence*), serta berbagai teknik pengolahan citra lainnya (Wibysono et al., 2022). Selain itu, OpenCV memiliki kemampuan untuk menangani proses *input* dan *output* gambar atau video, transformasi citra, *sampling*, hingga integrasi dengan modul *machine learning* (Zulfachmi et al., 2024).

Secara teknis, OpenCV ditulis menggunakan bahasa C/C++ untuk performa yang optimal, namun telah diperluas penggunaannya ke bahasa pemrograman populer lainnya seperti Python, Java, dan MATLAB (Zulfachmi et al., 2024). Keunggulan ini membuat OpenCV menjadi fleksibel dan mudah diintegrasikan ke berbagai *platform*, salah satunya Raspberry Pi.

Dalam praktiknya, OpenCV banyak dimanfaatkan untuk mempermudah proses pengolahan gambar, seperti mengubah warna, meningkatkan ketajaman, menyesuaikan posisi objek, dan memfilter *noise* (Zulfachmi et al., 2024). Dengan kombinasi kecepatan pemrosesan, kelengkapan fitur, serta kompatibilitas dengan berbagai *library*, OpenCV menjadi salah satu pilihan utama bagi pengembang dalam membangun sistem berbasis pengolahan citra.

2.7 You Only Look Once (YOLO)

You Only Look Once (YOLO) merupakan algoritma *deep learning* yang digunakan untuk melakukan deteksi dan klasifikasi objek secara *real-time*. YOLO bekerja dengan memproses citra dalam satu kali pengamatan (*single-stage detection*), di mana gambar dibagi ke dalam beberapa *grid* dan setiap *grid* bertugas memprediksi keberadaan serta kelas objek. Pendekatan ini memungkinkan proses deteksi berlangsung dengan cepat tanpa mengorbankan akurasi secara signifikan, sehingga sesuai untuk aplikasi yang membutuhkan respon waktu nyata, seperti sistem pemilahan sampah otomatis (Fajar et al., 2025).

Salah satu varian YOLO yang banyak digunakan pada perangkat dengan sumber daya terbatas adalah YOLOv5s. Varian ini dirancang dengan arsitektur yang lebih ringan dibandingkan YOLO versi standar, sehingga mampu memberikan performa deteksi yang efisien dengan kebutuhan komputasi yang lebih rendah (Santoso et al., 2025). Karakteristik tersebut menjadikan YOLOv5s sangat cocok diimplementasikan pada Raspberry Pi, khususnya untuk mendukung sistem pemilahan sampah berbasis *image detection*.

Pemilihan YOLOv5s didasarkan pada kebutuhan implementasi sistem pada Raspberry Pi 3B+ yang memiliki keterbatasan sumber daya komputasi, terutama pada kapasitas RAM dan

kemampuan pemrosesan CPU. YOLOv5s merupakan varian terkecil dari keluarga YOLOv5 yang memiliki jumlah parameter dan ukuran model lebih kecil dibandingkan varian lainnya, sehingga membutuhkan memori yang lebih rendah dan waktu inferensi yang lebih cepat.

Meskipun telah tersedia versi YOLO yang lebih baru seperti YOLOv8, YOLOv9, maupun YOLOv11, model-model tersebut umumnya memiliki kompleksitas arsitektur yang lebih tinggi dan membutuhkan sumber daya komputasi yang lebih besar. Pada perangkat Raspberry Pi 3B+, penggunaan model yang terlalu kompleks dapat menyebabkan peningkatan waktu inferensi dan penurunan performa *real-time*.

Selain itu, YOLOv5 memiliki dokumentasi yang matang, komunitas pengguna yang luas, serta dukungan ekspor model ke berbagai format, termasuk TensorFlow Lite (TFLite), yang diperlukan dalam penelitian ini untuk optimasi pada perangkat. Berdasarkan pertimbangan efisiensi komputasi, kemudahan implementasi, dan kemampuan mempertahankan akurasi yang tinggi, YOLOv5s dipilih sebagai model yang paling sesuai untuk sistem pemilah sampah otomatis yang dikembangkan.

2.8 TensorFlow



Gambar 2.2 TensorFlow
Sumber: <https://logowik.com>

TensorFlow adalah framework machine learning yang dikembangkan oleh Google untuk membuat dan melatih model kecerdasan buatan. Pada CNN terdapat beberapa lapisan, seperti Conv2D untuk mengambil ciri dari gambar, MaxPooling untuk mengurangi ukuran data agar proses lebih cepat, Flatten untuk mengubah data menjadi satu baris vektor, dan Dense untuk menentukan hasil klasifikasi. Saat proses pelatihan, model akan belajar dari data dengan cara memperbaiki kesalahan prediksi menggunakan metode optimasi seperti teknik *backpropagation*. (Rizqolima & Widhiantoro, 2025)

Selain digunakan di komputer, TensorFlow juga memiliki versi ringan yang disebut TensorFlow Lite (TFLite). TensorFlow Lite digunakan agar model dapat dijalankan pada perangkat dengan kemampuan terbatas seperti Raspberry Pi atau *smartphone*. Model yang sudah dilatih di TensorFlow dapat diubah ke format TFLite supaya ukurannya lebih kecil dan proses prediksinya lebih cepat.

2.9 Google Colaboratory (Google Colab)



Gambar 2.3 Google Colaboratory
Sumber: <https://developers.google.com/colab>

Google Colaboratory (Google Colab) merupakan *platform* berbasis *cloud* yang disediakan oleh Google untuk menjalankan kode Python secara daring tanpa perlu melakukan instalasi perangkat lunak tambahan pada komputer lokal. *Platform* ini sangat mendukung pengembangan dan pelatihan model machine learning karena menyediakan akses ke GPU dan TPU secara gratis dalam batas tertentu.

Pada penelitian ini, Google Colab digunakan sebagai lingkungan pelatihan model YOLOv5s karena mampu mempercepat proses *training* dibandingkan jika dijalankan pada komputer dengan spesifikasi standar. Selain itu, Google Colab dapat terintegrasi dengan Google Drive, sehingga memudahkan penyimpanan *dataset*, file konfigurasi, serta hasil pelatihan model.

2.10 Webcam



Gambar 2.4 Webcam
Sumber: <https://www.logitech.com>

Webcam merupakan perangkat *input* visual yang digunakan untuk menangkap gambar atau video secara *real-time*. Dalam sistem pemilah sampah otomatis, *webcam* berperan penting sebagai alat akuisisi citra, yaitu menangkap gambar objek sampah yang akan diklasifikasikan. Gambar hasil tangkapan *webcam* kemudian diproses oleh sistem deteksi citra, seperti *object detection*, untuk mengidentifikasi jenis sampah berdasarkan bentuk, warna, dan karakteristik visual lainnya.

Penggunaan *webcam* dalam sistem ini memiliki beberapa kelebihan. Salah satunya adalah kemudahan integrasi dengan perangkat seperti Raspberry Pi atau komputer mini lainnya. *Webcam* juga mendukung pengambilan gambar secara langsung (*real-time*), sehingga proses pemilahan dapat berlangsung secara otomatis tanpa perlu intervensi manual.

2.11 PCA9685 PWM Driver



Gambar 2.5 PCA9685

Sumber: <https://www.az-delivery.de>

PCA9685 merupakan PWM driver yang umum digunakan untuk mengendalikan motor servo karena mampu menyediakan hingga 16 channel keluaran PWM secara independen. Driver ini menghasilkan sinyal *Pulse Width Modulation* dengan rentang frekuensi yang dapat diatur, yaitu sekitar 24 Hz hingga 1526 Hz, serta memiliki resolusi 12-bit yang memungkinkan pengaturan lebar pulsa secara presisi (Suciningtyas et al., 2024). PCA9685 berkomunikasi dengan sistem utama melalui antarmuka I2C, sehingga hanya memerlukan sedikit pin pada mikrokontroler atau *single board computer*.

2.12 Motor Servo (MG996R)



Gambar 2.6 Motor Servo MG996R

Sumber: <https://www.kiwi-electronics.com>

Motor servo merupakan aktuator yang bekerja dengan sistem umpan balik tertutup (*closed-loop control*), sehingga posisi sudut putaran motor dapat dikendalikan secara presisi. Motor ini terdiri dari motor DC, rangkaian roda gigi (*gear*), potensiometer, serta rangkaian kontrol internal. Pengaturan sudut putar servo dilakukan menggunakan sinyal *Pulse Width*

Modulation (PWM), di mana *pulse width* menentukan posisi sudut poros servo (Al-Rafi et al., 2025).

Motor servo MG996R memiliki torsi yang cukup besar, yaitu sekitar 9,40 kg/cm pada tegangan 4,8 V dan dapat mencapai sekitar 11,00 kg/cm pada tegangan 6,0 V, dengan kecepatan pergerakan sekitar 0,19 detik untuk setiap sudut 60° (Slamat, 2021). Selain itu, servo ini menggunakan roda gigi berbahan logam (*metal gear*) yang membuatnya lebih tahan terhadap beban mekanik dan siklus kerja berulang, sehingga cocok digunakan pada mekanisme buka-tutup (*trap door*) dalam sistem pemilah sampah otomatis (Slamat, 2021).

2.13 Sensor Ultrasonik (HC-SR04)



Gambar 2.7 Sensor Ultrasonik HC-SR04

Sumber: <https://www.arduinoindonesia.id>

Sensor ultrasonik merupakan sensor yang bekerja dengan memanfaatkan pantulan gelombang suara berfrekuensi tinggi untuk mendeteksi keberadaan objek di depannya. Sensor ini beroperasi pada frekuensi di atas batas pendengaran manusia, yaitu lebih dari 20 kHz. Secara umum, sensor ultrasonik terdiri dari dua bagian utama, yaitu unit pemancar (*transmitter*) dan unit penerima (*receiver*). Unit pemancar berfungsi untuk mengirimkan gelombang ultrasonik ke arah objek, sedangkan unit penerima bertugas menangkap gelombang pantul yang kembali setelah mengenai permukaan objek. (Kristanti et al., 2022)

Waktu tempuh antara pengiriman dan penerimaan gelombang ultrasonik kemudian digunakan oleh sistem untuk menghitung jarak objek terhadap sensor. Semakin dekat objek, maka waktu pantul yang diterima akan semakin singkat, dan sebaliknya. Pada penelitian ini, sensor ultrasonik dimanfaatkan untuk mengukur volume sampah di dalam masing-masing bak sampah.

2.14 Firebase

Firebase merupakan layanan *Backend as a Service* (BaaS) yang dikembangkan oleh Google untuk membantu dan mempercepat proses pengembangan aplikasi. Firebase

menyediakan berbagai layanan yang dapat digunakan oleh pengembang, seperti penyimpanan data, autentikasi pengguna, dan sinkronisasi data secara *real-time*. Dengan adanya layanan ini, pengembang tidak perlu membangun dan mengelola backend secara mandiri sehingga proses pengembangan aplikasi menjadi lebih efisien (Mahendra & Winardi, 2023).

Salah satu fitur utama Firebase adalah *Firebase Realtime Database*, yaitu basis data berbasis cloud yang memungkinkan data disimpan dan disinkronkan secara langsung (*real-time*) antar perangkat yang terhubung. Melalui fitur ini, perubahan data yang terjadi pada satu perangkat dapat segera diperbarui dan diterima oleh perangkat lain tanpa memerlukan proses penyegaran (*refresh*) secara manual (Mahendra & Winardi, 2023).

Dalam penerapan *Internet of Things* (IoT), *Firebase Realtime Database* dapat dimanfaatkan sebagai media penyimpanan dan pertukaran data antar perangkat yang terhubung ke internet. Kemampuan sinkronisasi data secara *real-time* menjadikan Firebase sebagai solusi yang efektif untuk mendukung proses pemantauan dan pengendalian sistem IoT dari jarak jauh.

2.15 LabelImg

LabelImg merupakan perangkat lunak *open-source* yang digunakan untuk melakukan anotasi gambar pada proses pembuatan dataset *object detection*. Melalui LabelImg, pengguna dapat membuat *bounding box* pada objek yang terdapat di dalam gambar serta memberikan label sesuai dengan kelas objek yang telah ditentukan. Hasil anotasi kemudian disimpan dalam format yang dapat digunakan pada proses pelatihan model deteksi objek, seperti format YOLO. Proses anotasi ini bertujuan untuk memberikan informasi posisi dan kelas objek sehingga model YOLOv5s dapat mempelajari karakteristik setiap jenis sampah selama proses pelatihan.

2.16 Library Adafruit

Library Adafruit merupakan kumpulan pustaka (*library*) untuk memudahkan komunikasi antara mikrokontroler atau komputer mini dengan berbagai perangkat keras seperti sensor, modul tampilan, dan aktuator. *Library* ini menyediakan fungsi-fungsi yang telah dikembangkan sehingga pengguna tidak perlu membuat kode komunikasi perangkat keras dari awal. Penggunaan *library Adafruit* dapat mempercepat proses pengembangan sistem karena mendukung berbagai protokol komunikasi seperti I2C, SPI, dan UART.

Dalam penelitian ini, *library Adafruit* digunakan untuk mempermudah pengendalian modul LCD 16x2 I2C yang berfungsi menampilkan hasil deteksi sampah dan informasi

kapasitas tempat sampah. Dengan adanya *library* ini, proses pengiriman data ke LCD dapat dilakukan dengan lebih sederhana dan efisien.

2.17 Library GPIO

GPIO (*General Purpose Input/Output*) merupakan antarmuka pada Raspberry Pi yang memungkinkan perangkat untuk berkomunikasi dengan komponen elektronik eksternal melalui pin *input* dan *output*. Untuk mengakses pin GPIO, Raspberry Pi menyediakan *library* yang dapat digunakan dalam bahasa pemrograman Python, salah satunya adalah RPi.GPIO. *Library* ini memungkinkan pengguna untuk mengontrol kondisi pin digital, membaca data dari sensor, serta mengendalikan aktuator seperti relay dan servo motor.

Dalam penelitian ini, *library* GPIO digunakan untuk menghubungkan Raspberry Pi dengan berbagai perangkat keras seperti sensor ultrasonik dan servo motor. Melalui *library* GPIO, Raspberry Pi dapat membaca data jarak dari sensor ultrasonik serta mengirimkan sinyal kendali kepada servo motor untuk mengarahkan sampah ke tempat yang sesuai berdasarkan hasil klasifikasi yang dilakukan oleh model YOLOv5s. Dengan demikian, *library* GPIO berperan penting dalam proses integrasi antara perangkat lunak dan perangkat keras pada sistem pemilah sampah otomatis.

2.18 Black Box Testing

Black box testing atau yang juga dikenal sebagai behavioral testing merupakan metode pengujian sistem yang dilakukan dengan mengamati hubungan antara *input* dan *output* tanpa mengetahui struktur kode program di dalamnya (Nurfauziah & Jamaliyah, 2022). Pengujian ini biasanya dilakukan pada tahap akhir pengembangan untuk memastikan bahwa seluruh fitur telah berjalan sesuai dengan kebutuhan dan spesifikasi yang telah dirancang.

Black box testing memiliki peran penting dalam memvalidasi fungsi keseluruhan sistem agar dapat dipastikan bekerja dengan baik sebelum digunakan oleh pengguna. Penguji hanya berinteraksi melalui antarmuka sistem dengan memberikan berbagai variasi *input*, kemudian mengamati apakah *output* yang dihasilkan sudah sesuai. Keberhasilan pengujian dapat dilihat dari kesesuaian hasil sistem dengan skenario pengujian, spesifikasi kebutuhan, serta rancangan awal yang telah dibuat.

Pengujian *black box* bersifat dinamis, yang ditunjukkan oleh beberapa karakteristik antara lain (Nurfauziah & Jamaliyah, 2022):

1. Pengujian dilakukan dengan menjalankan sistem secara langsung untuk melihat respons terhadap *input* yang diberikan.
2. Fokus pengujian berada pada fungsi sistem, bukan pada struktur kode atau logika internal program.
3. Sistem diuji menggunakan *input* yang valid maupun tidak valid untuk memastikan sistem mampu menangani berbagai kemungkinan kondisi.
4. Hasil pengujian dievaluasi berdasarkan kesesuaian antara *output* yang diharapkan dan output yang dihasilkan sistem.

Dalam praktiknya, pengujian ini dilakukan dengan membuat sejumlah kasus uji yang membandingkan kondisi benar dan salah, sehingga dapat diketahui apakah sistem telah berfungsi sesuai dengan tujuan pengembangannya.

2.19 Penelitian Terdahulu

Penelitian terkait pemilahan sampah otomatis berbasis *image detection* terus berkembang seiring meningkatnya pemanfaatan algoritma deep learning, khususnya YOLO, pada perangkat berdaya komputasi rendah seperti Raspberry Pi. Berbagai studi telah mengkaji sistem klasifikasi sampah secara *real-time* dengan pendekatan dan hasil yang beragam, baik dari sisi metode, akurasi, maupun integrasi sistem. Oleh karena itu, kajian terhadap penelitian terdahulu diperlukan sebagai landasan dalam merancang dan mengembangkan sistem pemilah sampah otomatis pada penelitian ini.

Penelitian oleh Hendri & Utaminigrum (2022) berjudul "Rancang Bangun Sistem Pengklasifikasi Jenis Sampah Organik dan Anorganik Menggunakan Metode You Only Look Once versi 3 Berbasis Raspberry Pi" mengembangkan sistem klasifikasi sampah menggunakan algoritma YOLOv3 dan Raspberry Pi 4. Sistem ini diuji sebanyak 15 kali untuk masing-masing jenis sampah, menghasilkan akurasi rata-rata 92% untuk sampah organik dan 93% untuk sampah anorganik. Waktu komputasi yang dibutuhkan rata-rata adalah 15,1 detik untuk sampah organik dan 12,5 detik untuk sampah anorganik. Meski akurasinya tinggi, sistem ini mengalami kendala saat mendeteksi objek dengan posisi tidak jelas di kamera, serta keterbatasan Raspberry Pi dalam memproses model berat seperti YOLOv3 secara cepat.

Selanjutnya Fernando & Utaminigrum (2024) pada penelitian berjudul "Rancang Bangun Sistem Klasifikasi Sampah Menggunakan YOLOv8 Berbasis Raspberry Pi 4" membahas penerapan beberapa varian algoritma YOLOv8 untuk sistem klasifikasi sampah. Hasil penelitian menunjukkan bahwa setiap varian YOLOv8 memiliki perbedaan performa

baik dari sisi akurasi maupun waktu komputasi. Berdasarkan pengujian sebanyak 16 kali, model YOLOv8n menghasilkan performa terbaik dengan nilai akurasi, presisi, dan recall yang tinggi, serta rata-rata akurasi sistem mencapai 94%. Selain itu, waktu komputasi YOLOv8n sekitar 0,69 detik, sehingga lebih sesuai untuk diterapkan pada sistem klasifikasi sampah secara real-time berbasis Raspberry Pi 4.

Penelitian oleh Santoso & Gamar (2025) dengan judul “Deteksi Sampah Botol Plastik di Perairan Menggunakan YOLOv4-Tiny” melakukan penelitian terkait deteksi sampah botol plastik secara *real-time* menggunakan algoritma YOLOv4-Tiny yang diimplementasikan pada Raspberry Pi 5. Model dilatih menggunakan dataset hasil augmentasi dengan resolusi input 320×320 piksel. Hasil penelitian menunjukkan bahwa YOLOv4-Tiny mampu bekerja secara real-time dengan performa deteksi yang baik pada perangkat komputasi terbatas. Namun, penelitian ini masih berfokus pada satu jenis objek dan belum mengintegrasikan sistem pemilahan sampah secara fisik.

Pada penelitian berikutnya, Fajar et al. (2025) dalam penelitian berjudul “Rancang Bangun Pemilahan Sampah Plastik Otomatis Menggunakan YOLO pada Raspberry Pi” mengembangkan sistem pemilah sampah plastik otomatis yang dilengkapi kamera, conveyor, dan motor servo. Sistem ini memanfaatkan algoritma YOLOv5 dan pustaka OpenCV untuk mendeteksi serta mengklasifikasikan sampah plastik secara *real-time*. Dataset pelatihan diperoleh dari Roboflow dan proses pelatihan dilakukan menggunakan Google Colab. Hasil pengujian menunjukkan bahwa sistem telah mampu melakukan pemilahan sampah plastik, namun akurasi yang diperoleh masih relatif rendah, yaitu sebesar 47,5%, sehingga diperlukan peningkatan kualitas dataset dan optimasi model untuk meningkatkan kinerja sistem.

Penelitian lain oleh Adlim et al. (2025) yang berjudul “Implementasi Object Detection Klasifikasi Sampah Organik dan Anorganik Menggunakan Raspberry Pi dengan Algoritma YOLO” mengembangkan sistem klasifikasi sampah otomatis dengan memanfaatkan model YOLOv8n untuk membedakan sampah organik dan anorganik secara *real-time*. Model dipilih karena ringan dan efisien sehingga dapat dijalankan pada Raspberry Pi 5. Hasil pelatihan menunjukkan performa yang baik dengan nilai precision sebesar 0,93, recall sebesar 0,852, dan mAP50 sebesar 0,929. Model kemudian dikonversi ke format NCNN dan mampu berjalan dengan waktu inferensi kurang dari 100 milidetik per gambar serta kecepatan rata-rata 6,5 FPS. Sistem juga berhasil mengintegrasikan deteksi objek dengan kendali motor servo untuk

melakukan pemisahan sampah secara otomatis, serta dilengkapi logika penundaan 2 detik untuk mengurangi kesalahan deteksi akibat *noise*.

Berdasarkan penelitian-penelitian terdahulu, algoritma YOLO telah banyak diterapkan untuk sistem klasifikasi dan pemilahan sampah dengan tingkat akurasi yang cukup baik. Namun, masih terdapat beberapa keterbatasan, seperti penggunaan model yang relatif berat, fokus penelitian yang hanya pada satu jenis sampah, belum adanya integrasi monitoring berbasis IoT, serta implementasi yang umumnya menggunakan perangkat dengan spesifikasi yang lebih tinggi seperti Raspberry Pi 4 dan Raspberry Pi 5.

Posisi penelitian ini berada pada pengembangan sistem pemilah sampah otomatis berbasis image detection yang mengintegrasikan algoritma YOLOv5s, TensorFlow Lite, Raspberry Pi 3B+, dan Internet of Things (IoT). Berbeda dengan penelitian sebelumnya, penelitian ini berfokus pada optimalisasi model agar dapat berjalan secara efisien pada Raspberry Pi 3B+ yang memiliki sumber daya komputasi lebih terbatas. Raspberry Pi 3B+ dipilih karena memiliki spesifikasi yang cukup untuk menjalankan sistem sekaligus mewakili perangkat edge dengan keterbatasan memori dan kemampuan pemrosesan.

Dengan demikian, penelitian ini dapat membuktikan bahwa model YOLOv5s yang telah dikonversi ke TensorFlow Lite tetap mampu melakukan deteksi dan klasifikasi sampah secara *real-time* tanpa memerlukan perangkat dengan spesifikasi tinggi. Selain itu, sistem yang dikembangkan tidak hanya melakukan klasifikasi sampah, tetapi juga mengintegrasikan servo motor untuk pemilahan otomatis, sensor ultrasonik untuk pemantauan kapasitas tempat sampah, serta Firebase sebagai media monitoring berbasis IoT. Oleh karena itu, kontribusi penelitian ini terletak pada pengembangan sistem pemilah sampah otomatis yang akurat, efisien, ekonomis, dan terintegrasi untuk mendukung pengelolaan sampah secara lebih cerdas.

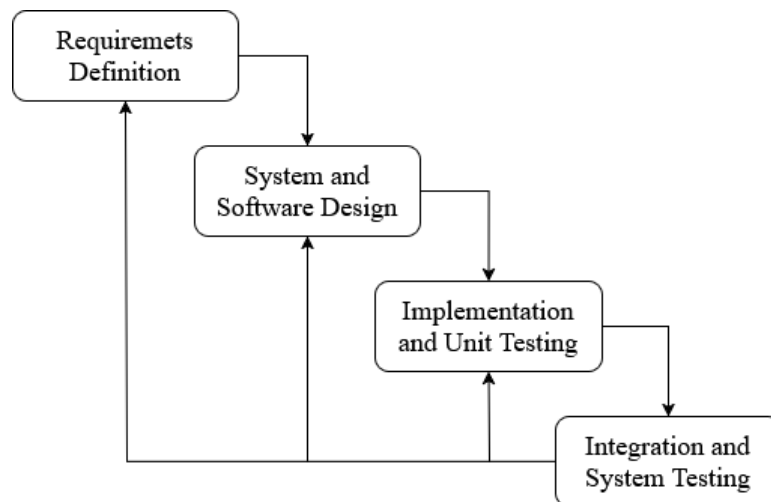
Tabel 2.2 Tabel Penelitian Terdahulu

Judul Penelitian	Perangkat	Metode	Objek Sampah	Hasil Utama	Keterbatasan
Sistem Klasifikasi Sampah Organik dan Anorganik Menggunakan YOLOv3 Berbasis Raspberry Pi 4 (Hendri & Utaminigrum, 2022)	Raspberry Pi 4	YOLOv3	Organik dan anorganik	Sistem mampu mengklasifikasikan sampah organik dan anorganik dengan tingkat akurasi yang tinggi menggunakan <i>dataset</i> berjumlah besar	Waktu komputasi relatif lama sehingga kurang optimal untuk penerapan <i>real-time</i> pada perangkat dengan sumber daya terbatas
Rancang Bangun Sistem Klasifikasi Sampah Menggunakan YOLOv8 Berbasis Raspberry Pi 4 (Fernando & Utaminigrum, 2024)	Raspberry Pi 4	YOLOv8	Organik, anorganik, B3, residu	Model YOLOv8n menghasilkan akurasi deteksi yang tinggi untuk empat kelas sampah dan terintegrasi dengan motor servo	Performa sistem masih dipengaruhi oleh variasi bentuk dan jenis sampah yang beragam
Deteksi Sampah Botol Plastik di Perairan Menggunakan YOLOv4-Tiny (Santoso & Gamar, 2025)	Raspberry Pi 5	YOLOv4-Tiny	Botol plastik	Deteksi berjalan secara <i>real-time</i> dengan performa baik	Fokus hanya pada satu jenis objek dan belum terintegrasi dengan mekanisme pemilahan fisik
Rancang Bangun Pemilahan Sampah Plastik Otomatis Menggunakan YOLO pada Raspberry Pi (Fajar et al., 2025)	Raspberry Pi 4	YOLOv5	Sampah plastik	Sistem pemilah sampah plastik otomatis dapat berfungsi sesuai tujuan dengan dukungan motor servo	Tingkat akurasi masih relatif rendah
Implementasi Object Detection Klasifikasi Sampah Organik dan Anorganik Menggunakan Raspberry Pi dengan Algoritma YOLO (Adlim et al., 2025)	Raspberry Pi 5	YOLOv8n	Organik dan Anorganik	Sistem berhasil mendeteksi dan mengklasifikasikan sampah organik dan anorganik secara <i>real-time</i>	Kecepatan 6,5 FPS masih tergolong sedang dan dapat menurun jika beban sistem meningkat

BAB III

METODOLOGI PENELITIAN

Penelitian ini menggunakan metode waterfall sebagai pendekatan pengembangan sistem. Menurut (Sommerville, 2016:47-48), metode waterfall menggambarkan proses pengembangan perangkat lunak sebagai serangkaian tahapan yang dilakukan secara berurutan, di mana keluaran dari setiap tahap menjadi masukan bagi tahap berikutnya. Tahapan tersebut meliputi analisis kebutuhan, perancangan sistem dan perangkat lunak, implementasi serta pengujian unit, serta integrasi dan pengujian sistem.



Gambar 3.1 Tahapan Metode Waterfall Menurut Sommerville

Metode ini bersifat *plan-driven* karena seluruh aktivitas pengembangan direncanakan dan dijadwalkan sejak awal. Metode waterfall menekankan pentingnya dokumentasi yang jelas dan lengkap pada setiap tahap pengembangan, karena hasil dari tiap fase menjadi dasar bagi proses pada tahap berikutnya. Menurut Ramadhan et al. (2023) metode waterfall memiliki kelebihan dan kekurangannya.

Kelebihan metode waterfall:

1. Proses pengembangannya jelas dan terstruktur, karena setiap tahap sudah ditentukan sejak awal. Alurnya tetap, mudah diterapkan, dan langkah-langkahnya tersusun rapi.
2. Cocok digunakan pada proyek yang kebutuhan sistemnya sudah jelas sejak awal, sehingga risiko terjadinya kesalahan atau perubahan besar selama pengembangan lebih kecil.

3. Hasil sistem yang dibuat dengan metode ini umumnya memiliki kualitas yang baik, karena setiap tahap harus diselesaikan dengan matang sebelum lanjut ke tahap berikutnya.
4. Dokumentasi pengembangan sangat lengkap dan tertata, karena setiap fase diwajibkan terselesaikan terlebih dahulu sebelum melanjutkan proses ke tahap selanjutnya.

Kekurangan metode waterfall:

1. Proyek jarang berjalan secara berurutan seperti yang direncanakan, sehingga jika ada perubahan di tengah jalan, pengembang perlu mengulang pekerjaan sebelumnya, dan hal ini sering menimbulkan masalah baru.
2. Setiap tahap dalam metode ini kurang fleksibel, karena keputusan yang diambil di awal harus diikuti sampai akhir, sehingga sulit untuk menyesuaikan perubahan yang muncul selama pengembangan.
3. Perubahan kebutuhan dari pengguna sulit diterapkan, karena metode ini mengharuskan semua kebutuhan ditetapkan sejak awal. Jika ada perubahan di tengah pengerjaan, penyesuaiannya akan cukup sulit.
4. Pengguna harus menunggu hingga seluruh tahapan selesai sebelum bisa melihat hasil akhir, karena proses pengerjaan dilakukan tahap demi tahap dan tidak bisa dilanjutkan ke tahap berikutnya sebelum tahap sebelumnya benar-benar selesai.
5. Perubahan yang terjadi di pertengahan proyek dapat membingungkan tim pengembang, karena mereka harus menyesuaikan kembali pekerjaan yang sudah dilakukan.
6. Ada kemungkinan tim mengalami waktu menganggur, misalnya ketika satu bagian harus menunggu bagian lain menyelesaikan tugasnya terlebih dahulu sebelum bisa melanjutkan pekerjaan.

3.1 *Requirements Definition (Analisis Kebutuhan)*

Tahap analisis kebutuhan merupakan langkah awal dalam metode waterfall yang bertujuan untuk mengidentifikasi dan mendefinisikan seluruh kebutuhan yang harus dipenuhi oleh sistem. Pada tahap ini, kebutuhan sistem dikumpulkan dan dianalisis secara menyeluruh agar sistem yang dikembangkan sesuai dengan tujuan dan kebutuhan pengguna. Dokumen kebutuhan yang dihasilkan pada tahap ini berperan sebagai acuan utama selama siklus pengembangan sistem, sehingga ketidakjelasan atau kesalahan dalam perumusan kebutuhan dapat berdampak pada munculnya permasalahan di tahap perancangan dan implementasi selanjutnya (Swasotomo, 2025).

3.1.1 Kebutuhan Fungsional

Kebutuhan fungsional menjelaskan fungsi utama yang harus dimiliki sistem agar mampu menjalankan proses pemilahan sampah secara otomatis sesuai dengan tujuan penelitian.

- a. Sistem dapat mengambil citra sampah melalui kamera (*webcam*). *Webcam* berfungsi untuk menangkap citra sampah yang digunakan sebagai *input* pada proses deteksi dan klasifikasi jenis sampah.
- b. Sistem ini mampu mengklasifikasikan tiga jenis sampah, yaitu kertas dan tisu, botol plastik, serta kaleng menggunakan model YOLOv5s yang dikonversi ke TensorFlow Lite. TensorFlow Lite digunakan untuk melakukan deteksi dan klasifikasi sampah secara *real-time* berdasarkan citra yang diambil oleh *webcam*.
- c. Sistem dapat menggerakkan motor servo untuk mengarahkan sampah ke bak yang sesuai dengan hasil deteksi. Motor servo berfungsi sebagai aktuator untuk mengarahkan atau memindahkan sampah ke bak yang sesuai berdasarkan hasil klasifikasi yang dilakukan oleh sistem.
- d. Sistem mampu mengukur volume sampah pada setiap bak menggunakan sensor ultrasonik. Sensor ultrasonik digunakan untuk mengukur jarak antara sensor dan permukaan sampah di dalam bak, sehingga volume atau tingkat kepenruhan bak dapat diketahui. Informasi ini digunakan untuk memantau kondisi bak sampah.
- e. Sistem dapat mengirimkan data jenis dan volume sampah ke Firebase sebagai media pemantauan jarak jauh. Data hasil klasifikasi jenis sampah dan hasil pengukuran volume bak dikirimkan ke Firebase melalui koneksi internet. Pengiriman data ini memungkinkan pemantauan kondisi tempat sampah secara *real-time* dari jarak jauh.
- f. Sistem dapat menampilkan informasi hasil deteksi dan volume sampah pada LCD berbasis I2C. LCD digunakan sebagai media tampilan untuk memberikan informasi secara langsung kepada pengguna mengenai jenis sampah yang terdeteksi serta kondisi volume masing-masing bak sampah.

3.1.2 Kebutuhan Perangkat Keras

Kebutuhan perangkat keras mencakup spesifikasi perangkat keras yang digunakan dalam pengembangan dan implementasi sistem, meliputi jenis perangkat, fungsi, serta peran masing-masing komponen dalam sistem secara keseluruhan.

Tabel 3.1 Kebutuhan Perangkat Keras

No	Nama Perangkat	Jumlah	Fungsi
1	Raspberry Pi 3B+	1	Sebagai pusat kendali sistem untuk menjalankan proses deteksi objek, pengolahan data sensor, pengendalian aktuator, serta komunikasi IoT
2	Webcam (USB)	1	Mengambil citra sampah sebagai input sistem <i>image detection</i>
3	Motor Servo MG996R	2	Mengarahkan sampah ke tempat penampungan sesuai hasil klasifikasi
4	PCA9685 16-Channel PWM Driver Servo	1	Menghasilkan sinyal PWM yang stabil untuk mengendalikan 2 motor servo
5	LCD 16×2 I2C	1	Menampilkan informasi sistem seperti status alat dan hasil deteksi
6	Power Supply 5V 5A	1	Menyediakan daya untuk Raspberry Pi, servo, dan perangkat pendukung lainnya
7	Sensor Ultrasonik HC-SR04	3	Mengukur volume sampah pada setiap tempat sampah berdasarkan jenis sampah yang telah diklasifikasikan
8	Kabel Jumper	-	Koneksi antar perangkat

3.1.3 Kebutuhan Perangkat Lunak

Kebutuhan ini mencakup spesifikasi perangkat lunak yang digunakan dalam pengembangan dan implementasi sistem, termasuk sistem operasi, bahasa pemrograman, pustaka pendukung, serta platform yang berperan dalam menjalankan sistem secara keseluruhan.

Tabel 3.2 Kebutuhan Perangkat Lunak

No	Kebutuhan Perangkat Lunak	Fungsi
1	Raspberry Pi OS	Platform utama untuk menjalankan seluruh proses sistem, mulai dari pengolahan citra, pengendalian perangkat keras, hingga komunikasi data ke Firebase
2	Bahasa Pemrograman Python	Bahasa utama dalam pengembangan sistem karena memiliki banyak library pendukung,

		khususnya untuk pengolahan citra dan integrasi perangkat keras
3	OpenCV	Pengolahan citra dan pengambilan gambar dari kamera (<i>webcam</i>)
4	YOLOv5s	Model deteksi objek yang digunakan untuk melatih dan mengenali jenis sampah
5	TensorFlow Lite	Menjalankan model hasil konversi (.tflite) pada Raspberry Pi untuk proses deteksi secara <i>real-time</i>
6	Library Adafruit (ServoKit dan Blinka)	Kontrol motor servo melalui modul PWM PCA9685 serta mengatur komunikasi I2C antara Raspberry Pi dan perangkat tambahan seperti LCD
7	Library GPIO (RPi.GPIO atau pigpio)	Akses pin <i>input</i> dan <i>output</i> pada Raspberry Pi, khususnya untuk membaca sensor ultrasonik
8	Firestore <i>Realtime Database</i>	Platform <i>IoT</i> untuk menyimpan dan menampilkan data hasil klasifikasi jenis sampah dan volume bak secara <i>real-time</i>
9	LabelImg	Anotasi <i>dataset</i>
10	Google Colab	melakukan proses pelatihan (<i>training</i>) model YOLOv5s

3.2 Pengumpulan dan Pengolahan *Dataset*

Pengumpulan dan pengolahan *dataset* merupakan tahap awal yang sangat penting dalam pengembangan sistem pemilah sampah otomatis. *Dataset* yang digunakan berfungsi sebagai data latih untuk melatih model agar mampu mengenali dan membedakan jenis sampah dengan tepat.

3.2.1 Kelas *Dataset*

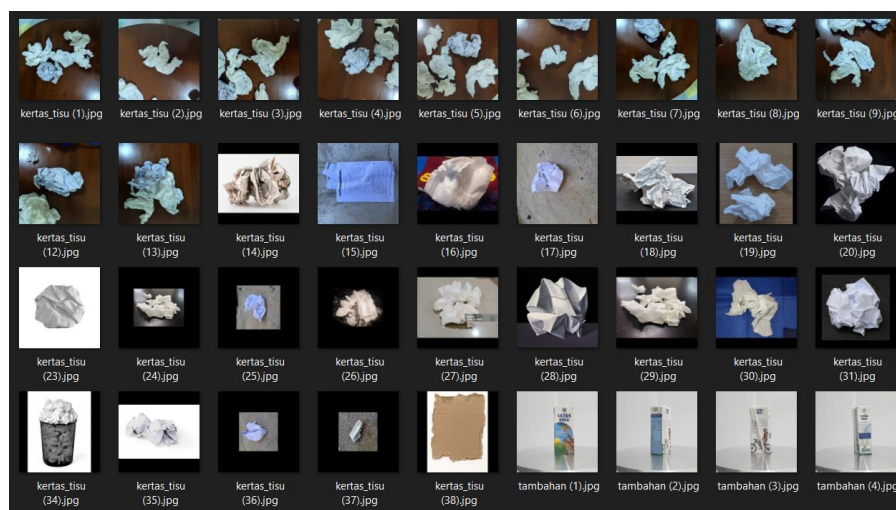
Kelas *dataset* pada penelitian ini ditentukan berdasarkan jenis sampah yang menjadi objek utama dalam proses deteksi dan pemilahan otomatis. Pada penelitian ini, digunakan 3 kelas yang akan menjadi klasifikasi jenis sampah.

Tabel 3.3 Kelas *Dataset*

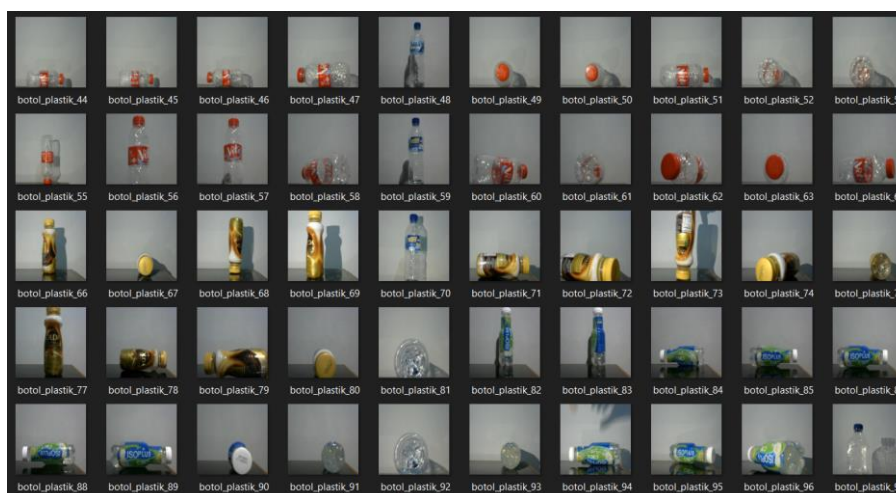
Kelas <i>Dataset</i> Sampah
Kertas dan Tisu
Botol Plastik
Kaleng

Kelas-kelas pada Tabel 3.3 dipilih karena merupakan jenis sampah yang umum ditemukan di lingkungan kampus (*Hall* FTI UII) dan memiliki karakteristik visual yang berbeda satu sama lain. Kertas dan tisu umumnya memiliki tekstur yang lebih lunak serta bentuk yang tidak beraturan, terutama setelah digunakan atau diremas. Botol plastik memiliki bentuk yang cenderung silinder dengan permukaan transparan atau berwarna, sedangkan kaleng memiliki permukaan logam yang lebih kaku dan bersifat reflektif. Perbedaan karakteristik visual tersebut menjadi dasar bagi model YOLOv5s dalam mempelajari dan membedakan setiap kategori sampah selama proses pelatihan dan pengujian.

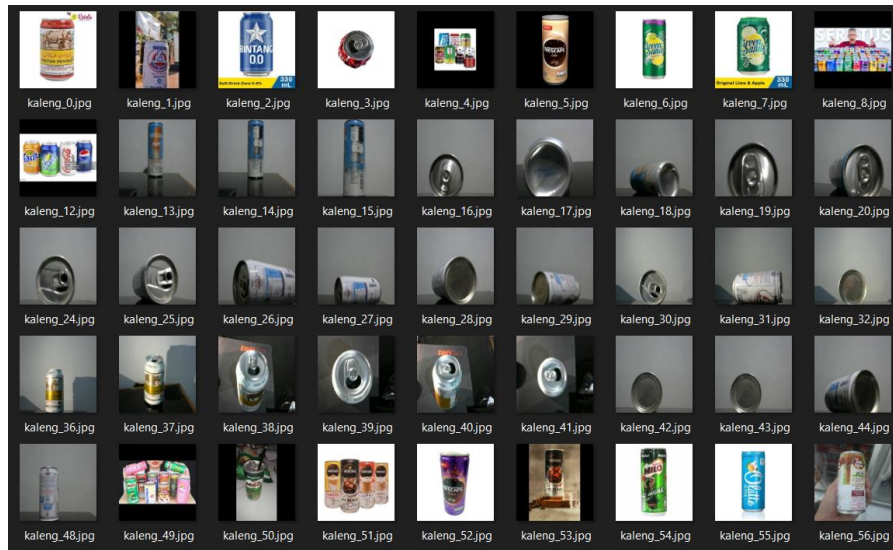
3.2.2 Pengumpulan *Dataset*



Gambar 3.2 Pengumpulan Citra *Dataset* Kertas dan Tisu

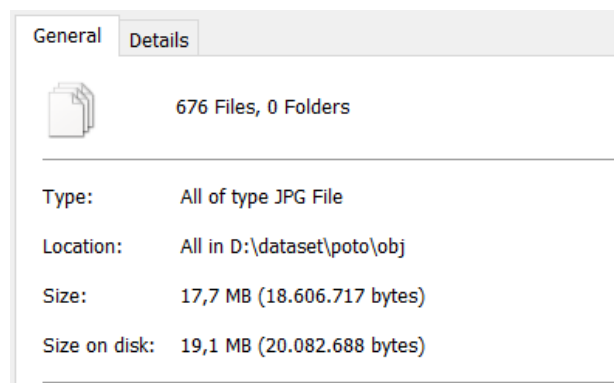


Gambar 3.3 Pengumpulan Citra *Dataset* Botol Plastik



Gambar 3.4 Pengumpulan Citra *Dataset* Kaleng

Tahap awal dalam proses pengembangan sistem dimulai dengan pengumpulan *dataset* yang akan digunakan untuk melatih model. *Dataset* berperan sebagai sumber utama informasi bagi model agar dapat mengenali dan mengklasifikasikan objek sesuai tujuan sistem. Pada penelitian ini, dataset diperoleh dari hasil dokumentasi langsung melalui *webcam*, kamera *smartphone* serta beberapa sumber pendukung lain (internet). Gambar yang tidak relevan, buram, atau terlalu gelap dihapus agar hanya data berkualitas baik yang digunakan.



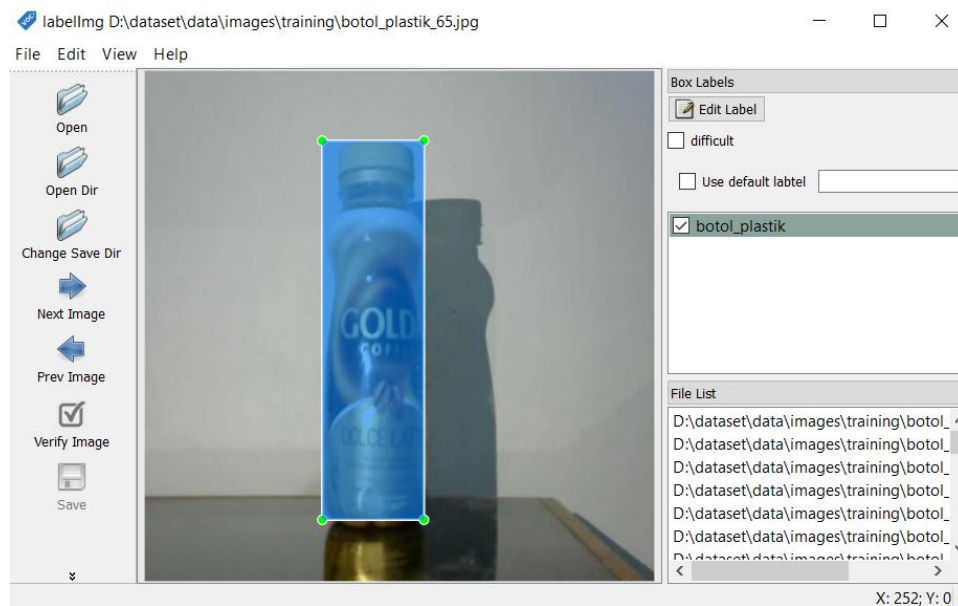
Gambar 3.5 Jumlah Citra Keseluruhan

Jumlah data yang digunakan dalam penelitian ini berjumlah 676 data, yang terdiri dari 229 data untuk kategori kertas dan tisu, 247 data untuk botol plastik, serta 200 data untuk kaleng.

3.2.3 Anotasi *Dataset*

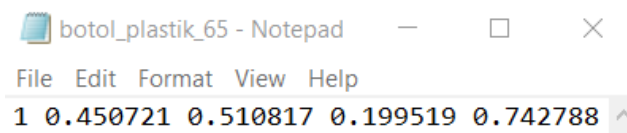
Proses anotasi *dataset* merupakan tahap awal yang penting sebelum proses pelatihan model deteksi objek dilakukan. Proses anotasi pada gambar (*dataset*) dilakukan menggunakan

tools labelImg. Setiap objek yang ingin diidentifikasi diberi kotak pembatas (*bounding box*) yang mengelilingi objek tersebut.



Gambar 3.6 Proses Anotasi Dataset Menggunakan tools labelImg

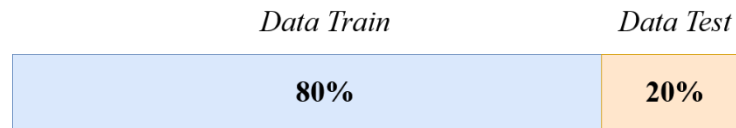
Setiap *bounding box* yang dibuat kemudian diberi label sesuai dengan kategori objek. Hasil anotasi ini disimpan dalam format .txt yang memuat informasi penting, antara lain koordinat posisi objek pada gambar, nama kelas, serta detail ukuran gambar. Dengan adanya *bounding box*, sistem dapat lebih fokus pada area objek tanpa terganggu oleh latar belakang gambar.



Gambar 3.7 Hasil Anotasi Gambar

Terlihat pada Gambar 3.7 bahwa *file* anotasi disimpan dalam format teks (.txt) dengan struktur lima nilai numerik dalam satu baris. Nilai pertama menunjukkan id kelas objek yang terdeteksi, sedangkan empat nilai berikutnya merepresentasikan koordinat *bounding box* dalam format ter-normalisasi, yaitu nilai titik tengah sumbu x (x_center), titik tengah sumbu y (y_center), lebar (*width*), dan tinggi (*height*) objek terhadap ukuran gambar.

3.2.4 Dataset Split



Gambar 3.8 Dataset Split

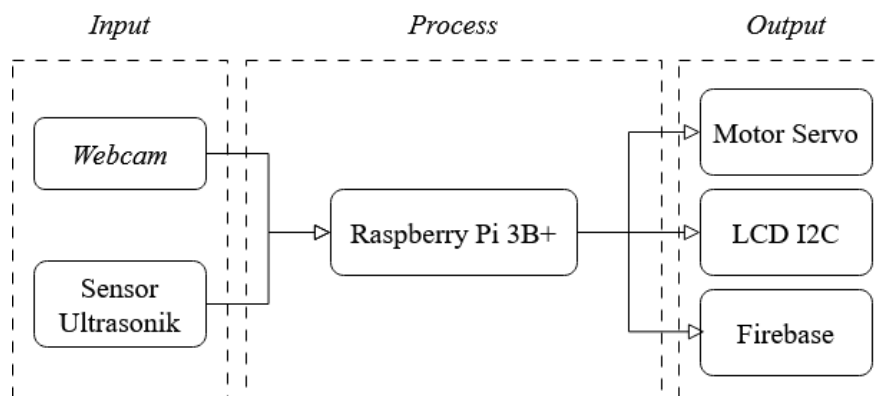
Dataset yang digunakan dalam penelitian ini dibagi menjadi dua bagian, yaitu *data train* dan *data test*. Sebanyak 80% dari total *dataset* digunakan sebagai data latih untuk melatih model agar mampu mengenali dan mengklasifikasikan jenis sampah dengan baik. Sementara itu, 20% sisanya digunakan sebagai *data test* yang berfungsi untuk mengevaluasi kinerja model selama proses pelatihan berlangsung. Pembagian *dataset* ini bertujuan untuk memastikan bahwa model yang dihasilkan mampu bekerja dengan baik pada data yang belum pernah dilihat sebelumnya.

3.3 System and Software Design (Perancangan Sistem dan Perangkat Lunak)

Bagian ini menjelaskan proses perancangan sistem dan perangkat lunak yang digunakan dalam penelitian. Perancangan dilakukan untuk menggambarkan alur kerja sistem, hubungan antar komponen, serta mekanisme pengolahan data mulai dari *input* hingga *output*.

3.3.1 Diagram Blok Sistem

Diagram blok sistem digunakan untuk menggambarkan hubungan antar komponen dalam sistem pemilah sampah otomatis. Sistem terdiri dari tiga bagian utama, yaitu *input*, *process*, dan *output*, yang saling terintegrasi dan bekerja secara berurutan. Setiap bagian pada diagram blok memiliki peran masing-masing dalam mendukung proses pemilahan sampah secara otomatis berbasis *image detection* dan *IoT*.



Gambar 3.9 Diagram Blok Sistem

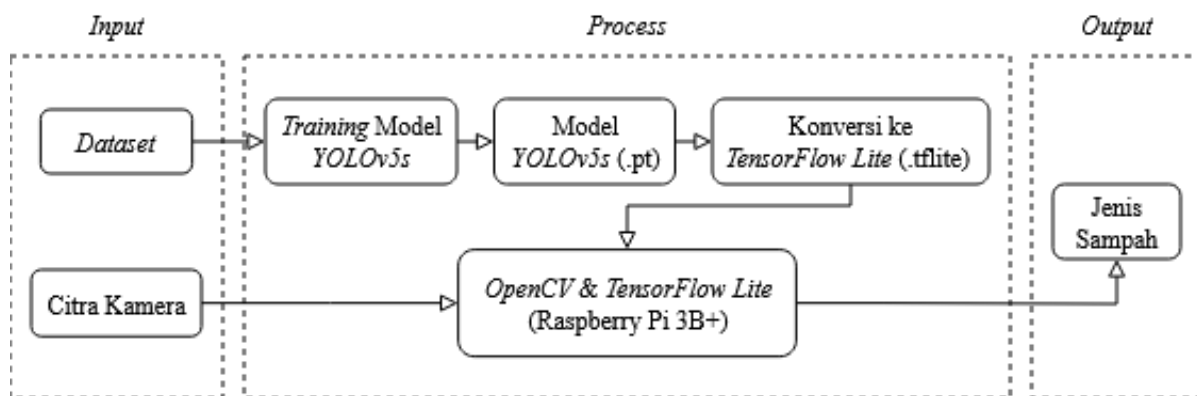
Pada bagian *input*, sistem menerima data dari beberapa perangkat sensor. *Webcam* berfungsi sebagai perangkat masukan visual yang digunakan untuk mengambil citra sampah yang masuk ke dalam sistem. Citra tersebut selanjutnya digunakan sebagai data utama dalam proses deteksi dan klasifikasi jenis sampah berbasis *object detection*. Selain itu, sensor ultrasonik digunakan untuk mengukur jarak permukaan sampah di dalam bak, sehingga dapat diketahui tingkat volume atau kapasitas tempat sampah.

Seluruh data dari perangkat *input* kemudian diproses oleh Raspberry Pi 3B+ yang berperan sebagai pusat kendali sistem. Raspberry Pi bertugas menjalankan pengolahan citra dan *object detection* menggunakan model YOLOv5s yang dikonversi ke TensorFlow Lite untuk mengklasifikasikan jenis sampah. Selain itu, Raspberry Pi juga mengolah data jarak dari sensor ultrasonik untuk menghitung volume sampah pada masing-masing bak serta mengendalikan perangkat keluaran sesuai dengan hasil pemrosesan.

Pada bagian *output*, sistem menghasilkan beberapa bentuk keluaran. Motor servo digunakan untuk mengarahkan sampah ke bak yang sesuai berdasarkan hasil klasifikasi. LCD berbasis I2C berfungsi untuk menampilkan informasi sistem, seperti jenis sampah yang terdeteksi dan kondisi volume tempat sampah. Selain itu, data hasil deteksi dan pengukuran volume dikirimkan ke Firebase sebagai pemantauan berbasis *Internet of Things (IoT)*, sehingga informasi dapat diakses secara *real-time*.

3.3.2 Diagram Blok *Object Detection*

Diagram blok *object detection* digunakan untuk menjelaskan proses pendeteksian dan pengenalan jenis sampah menggunakan model YOLOv5s yang dikonversi ke TensorFlow Lite. Diagram ini menggambarkan alur kerja sistem mulai dari tahap pelatihan model hingga proses klasifikasi jenis sampah saat sistem dijalankan.



Gambar 3.10 Diagram Blok *Object Detection*

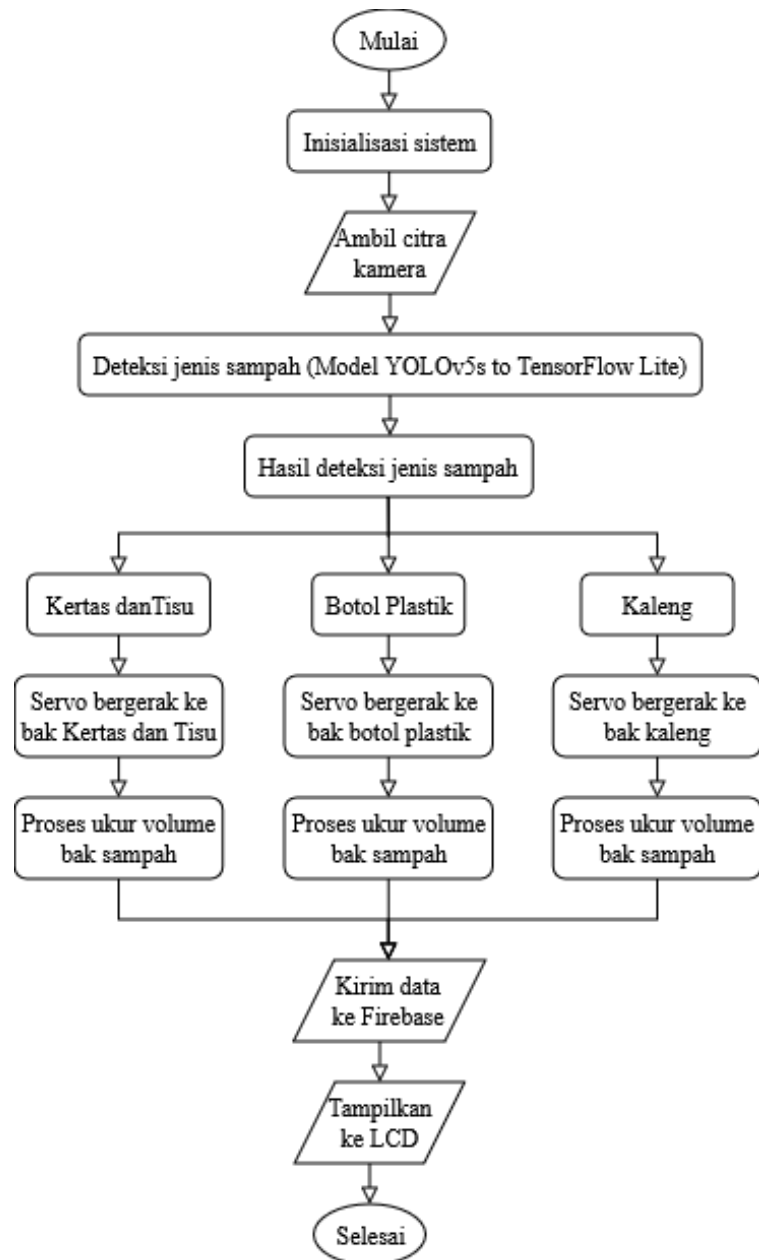
Proses *object detection* diawali dengan penyediaan *dataset* sampah yang berisi citra dari beberapa jenis sampah, yaitu kertas dan tisu, botol plastik, serta kaleng. *Dataset* ini digunakan pada tahap *training YOLOv5s* untuk melatih model agar mampu mengenali dan membedakan setiap jenis sampah berdasarkan ciri visualnya. Hasil dari proses pelatihan ini berupa model *YOLOv5s* dalam format .pt yang telah memiliki bobot (*weight*) hasil pelatihan. Model tersebut kemudian dikonversi ke dalam format *TensorFlow Lite* (.tflite) agar dapat dijalankan pada perangkat dengan sumber daya terbatas.

Model yang telah dikonversi selanjutnya diimplementasikan pada Raspberry Pi 3B+ sebagai perangkat pemrosesan utama. Pada saat sistem berjalan, citra sampah yang diambil oleh *webcam* dikirimkan ke Raspberry Pi untuk diproses menggunakan *OpenCV* dan *TensorFlow Lite*. *OpenCV* berfungsi untuk membantu proses pengolahan citra, sedangkan *TensorFlow Lite* digunakan untuk melakukan deteksi objek dan klasifikasi jenis sampah secara *real-time*.

Hasil dari proses deteksi tersebut berupa informasi jenis sampah yang teridentifikasi oleh sistem. Informasi ini kemudian digunakan sebagai dasar pengambilan keputusan dalam proses pemilahan sampah, yaitu dengan mengatur pergerakan motor servo agar sampah diarahkan ke bak yang sesuai dengan kategori yang telah ditentukan.

3.3.3 Flowchart Sistem

Flowchart sistem digunakan untuk menggambarkan tahapan proses kerja sistem pemilahan sampah otomatis secara keseluruhan. Diagram ini menunjukkan urutan proses mulai dari sistem dijalankan sampai proses pemilahan dan penyajian informasi selesai dilakukan.

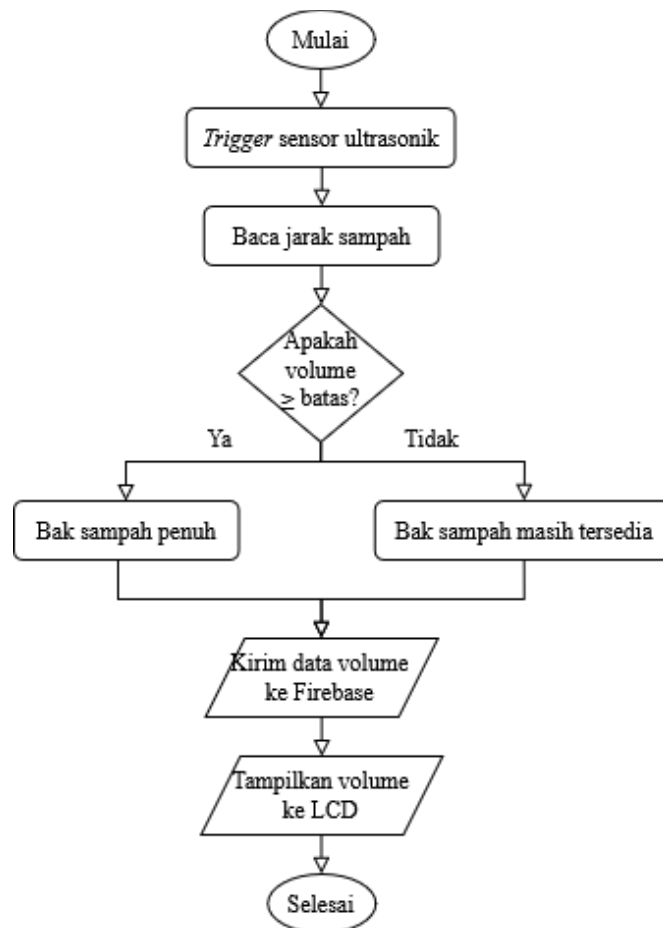
Gambar 3.11 *Flowchart Sistem*

Proses dimulai dengan inisialisasi sistem, di mana seluruh perangkat keras dan perangkat lunak dipersiapkan agar siap digunakan. Selanjutnya sistem akan mengambil citra menggunakan *webcam* dan memprosesnya menggunakan model YOLOv5s yang dikonversi ke TensorFlow Lite untuk menentukan jenis sampah. Berdasarkan hasil deteksi, sistem mengelompokkan sampah ke dalam kategori kertas dan tisu, botol plastik, atau kaleng.

Setelah jenis sampah diketahui, motor servo akan bergerak mengarahkan sampah ke bak yang sesuai. Selanjutnya, sistem melakukan pengukuran volume bak sampah untuk mengetahui tingkat keterisian. Data hasil pengukuran kemudian dikirim ke Firebase dan ditampilkan pada LCD I2C sebagai informasi bagi pengguna.

3.3.4 Flowchart Proses Pengukuran Volume Bak Sampah

Flowchart proses pengukuran volume bak sampah digunakan untuk menjelaskan tahapan sistem dalam memantau tingkat keterisian setiap bak sampah secara otomatis. Proses ini memanfaatkan sensor ultrasonik untuk mengukur jarak antara sensor dan permukaan sampah di dalam bak.



Gambar 3.12 *Flowchart* Proses Pengukuran Volume Bak Sampah

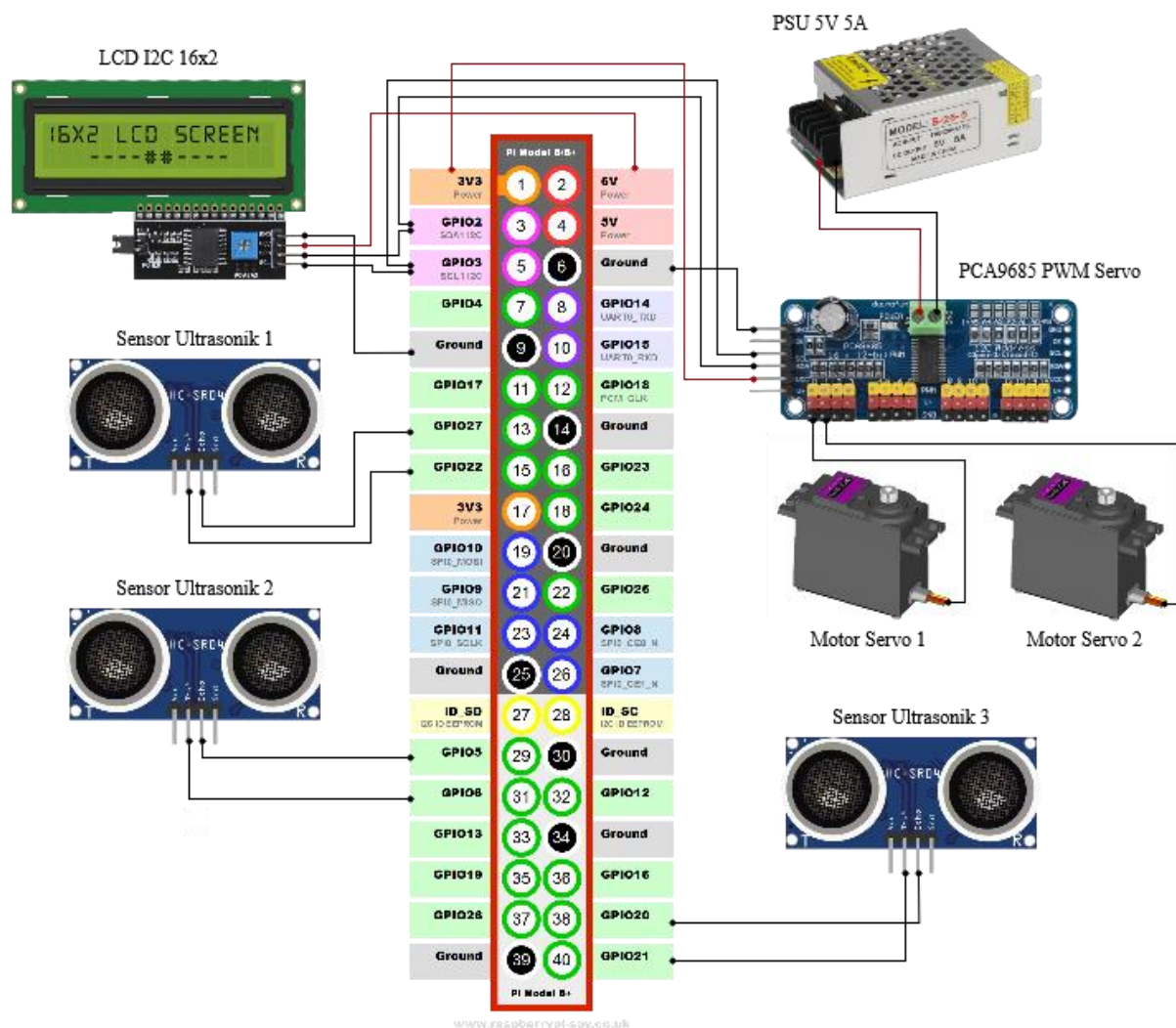
Proses diawali dengan sistem melakukan *trigger* pada sensor ultrasonik untuk mengirimkan gelombang ultrasonik. Gelombang yang dipantulkan oleh permukaan sampah kemudian diterima kembali oleh sensor, sehingga sistem dapat membaca jarak sampah di dalam bak. Hasil pembacaan jarak tersebut selanjutnya dibandingkan dengan nilai batas yang telah ditentukan. Apabila nilai volume telah melewati batas, maka bak sampah dianggap penuh. Sebaliknya, jika masih berada di bawah batas, maka bak sampah dinyatakan masih tersedia.

Informasi volume bak sampah kemudian dikirim ke Firebase sebagai data pemantauan jarak jauh dan ditampilkan pada LCD I2C untuk memberikan informasi secara langsung kepada pengguna. Setelah seluruh proses selesai, sistem kembali ke kondisi awal untuk melakukan pengukuran berikutnya.

3.4 Perancangan Perangkat Keras

Bagian ini menjelaskan perancangan perangkat keras yang digunakan dalam sistem pemilah sampah otomatis. Pembahasan meliputi pemilihan komponen, fungsi masing-masing perangkat, serta cara penghubungannya agar seluruh sistem dapat bekerja secara terintegrasi.

3.4.1 Rangkaian Sistem Perangkat Keras



Gambar 3.13 Rangkaian Perangkat Keras

Pada Gambar 3.13 menunjukkan rangkaian sistem perangkat keras yang akan dikembangkan. Rangkaian tersebut menunjukkan hubungan antar komponen, seperti Raspberry Pi, sensor, dan perangkat pendukung lainnya, sehingga dapat terlihat alur koneksi sistem secara keseluruhan.

3.4.2 Konfigurasi Pin dan Koneksi Antar Perangkat

Bagian ini menjelaskan konfigurasi pin antar perangkat yang digunakan dalam sistem pemilah sampah otomatis. Setiap komponen perangkat keras, dihubungkan ke pin tertentu pada Raspberry Pi sesuai dengan fungsi dan kebutuhan sistem. Konfigurasi pin ini bertujuan untuk memberikan gambaran yang jelas mengenai hubungan antar perangkat, sehingga memudahkan proses perakitan dan pemrograman sistem.

a. Koneksi Pin PCA9685 (Servo Driver)

Tabel 3.4 Koneksi Pin PCA9685 (Servo Driver)

No	Perangkat	Pin Sensor	Pin Raspberry Pi	Keterangan
1	PCA9685	VCC	3.3V (Pin 1)	Logika I2C
2		GND	GND (Pin 6)	Ground
3		SDA	GPIO2 / SDA (Pin 3)	Data I2C
4		SCL	GPIO3 / SCL (Pin 5)	Clock I2C
5		V+	PSU 5V Eksternal	Daya Eksternal

b. Koneksi Pin Servo MG996R

Tabel 3.5 Koneksi Pin Servo MG996R

No	Perangkat	Channel PCA 9685
1	Servo 1	Channel 0
2	Servo 2	Channel 1

c. Koneksi Pin LCD 16x2 I2C

Tabel 3.6 Koneksi Pin LCD 16x2 I2C

No	Perangkat	Pin Sensor	Pin Raspberry Pi	Keterangan
1	LCD 16x2 I2C	VCC	5V (Pin 2)	Logika I2C
2		GND	GND (Pin 9)	Ground
3		SDA	GPIO2 / SDA (Pin 3)	Data I2C
4		SCL	GPIO3 / SCL (Pin 5)	Clock I2C

d. Koneksi Pin Sensor Ultrasonik

Tabel 3.7 Koneksi Pin Sensor Ultrasonik

No	Perangkat	Pin Sensor	Pin Raspberry Pi	Keterangan
1	Ultrasonik 1	VCC	Channel 5 (PCA9685)	Catu Daya
2		GND	Channel 5 (PCA9685)	Ground
3		TRIG	GPIO 22 (Pin 13)	Trigger
4		ECHO	GPIO 27 (Pin 15)	Echo
5		VCC	Channel 6 (PCA9685)	Catu Daya

6	Ultrasonik 2	GND	Channel 6 (PCA9685)	Ground
7		TRIG	GPIO 6 (Pin 31)	Trigger
8		ECHO	GPIO 5 (Pin 29)	Echo
9	Ultrasonik 3	VCC	Channel 7 (PCA9685)	Catu Daya
10		GND	Channel 7 (PCA9685)	Ground
11		TRIG	GPIO 21 (Pin 40)	Trigger
12		ECHO	GPIO 20 (Pin 38)	Echo

e. Koneksi Kamera (Webcam)

Tabel 3.8 Koneksi Kamera (Webcam)

Perangkat	Koneksi	Keterangan
Webcam	USB Raspberry Pi	Pengambilan Citra

3.5 Perancangan dan Pelatihan Model

Pada tahap ini, model YOLOv5s disiapkan dan dikonfigurasi agar mampu mengenali jenis sampah yang telah ditentukan. Proses ini meliputi persiapan lingkungan pelatihan, penyusunan struktur *dataset*, pembuatan file konfigurasi, serta pelaksanaan training menggunakan *dataset* yang telah dianotasi.

Model dilatih untuk mendeteksi dan mengklasifikasikan tiga jenis sampah, yaitu kertas dan tisu, botol plastik, serta kaleng. Setelah proses pelatihan selesai, model terbaik yang dihasilkan kemudian dikonversi ke format TensorFlow Lite agar dapat diimplementasikan pada Raspberry Pi sebagai perangkat pemrosesan utama.

3.5.1 Proses *Training Model YOLOv5s*

Proses training model dilakukan menggunakan Google Colab, karena menyediakan lingkungan komputasi dengan dukungan GPU yang memadai untuk mempercepat proses *training model*. Tahap pelatihan model diawali dengan menyiapkan lingkungan pengembangan (*environment setup*).

```
!git clone https://github.com/ultralytics/yolov5
!cd yolov5
!pip install -r requirements.txt
```

Gambar 3.14 Perintah *Environment Setup YOLOv5s*

Repository YOLOv5 diunduh dari GitHub, kemudian seluruh kebutuhan pustaka (*dependencies*) diinstal menggunakan perintah `pip install -r requirements.txt`. Proses pelatihan

dilakukan pada komputer dengan spesifikasi yang memadai agar *training* dapat berjalan lebih optimal dan stabil.

```
# YOLOv5 requirements
# Usage: pip install -r requirements.txt

# Base -----
matplotlib>=3.2.2
numpy>=1.18.5
opencv-python>=4.1.1
Pillow>=7.1.2
PyYAML>=5.3.1
requests>=2.23.0
scipy>=1.4.1
torch>=1.7.0 # see https://pytorch.org/get-started/locally/ (recommended)
torchvision>=0.8.1
tqdm>=4.64.0
# protobuf<=3.20.1 # https://github.com/ultralytics/yolov5/issues/8012

# Logging -----
tensorboard>=2.4.1
# wandb
# clearml

# Plotting -----
pandas>=1.1.4
seaborn>=0.11.0

# Export -----
# coremltools>=6.0 # CoreML export
# onnx>=1.9.0 # ONNX export
# onnx-simplifier>=0.4.1 # ONNX simplifier
# nvidia-pyindex # TensorRT export
# nvidia-tensorrt # TensorRT export
# scikit-learn<=1.1.2 # CoreML quantization
# tensorflow>=2.4.1 # TF exports (-cpu, -aarch64, -macos)
# tensorflowjs>=3.9.0 # TF.js export
# opencvino-dev # OpenVINO export

# Deploy -----
# tritonclient[all]~=2.24.0

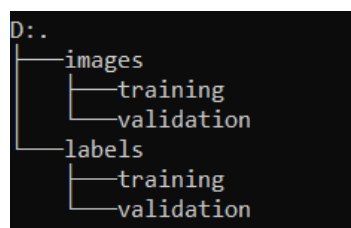
# Extras -----
ipython # interactive notebook
psutil # system utilization
thop>=0.1.1 # FLOPs computation
# mss # screenshots
# albumentations>=1.0.3
# pycocotools>=2.0 # COCO mAP
# roboflow
```

Gambar 3.15 Isi *File Requirements Environment Setup YOLOv5s*

File `requirements.txt` Gambar 3.15 berisi daftar library yang dibutuhkan oleh framework YOLOv5 agar dapat berjalan dengan baik. Library utama yang digunakan antara lain PyTorch (`torch` dan `torchvision`) sebagai framework deep learning untuk proses pelatihan dan inferensi model, OpenCV (`opencv-python`) untuk pengolahan citra dan video, serta NumPy dan SciPy untuk komputasi numerik. Selain itu, terdapat Matplotlib, Pandas, dan Seaborn yang digunakan untuk visualisasi dan analisis hasil pelatihan model. Library PyYAML digunakan untuk membaca file konfigurasi, Requests untuk komunikasi data melalui internet, dan TQDM untuk menampilkan progres proses pelatihan. Pada bagian logging, digunakan TensorBoard untuk memantau performa model selama training. Sementara itu, library tambahan seperti IPython, Psutil, dan THOP digunakan untuk mendukung proses pengembangan, pemantauan penggunaan sistem, serta perhitungan kompleksitas model.

Setelah *dependencies* selesai diinstal, dilakukan persiapan *dataset*. *Dataset* yang digunakan terdiri dari citra sampah yang telah diberi anotasi. Setiap citra memiliki file label dengan format `.txt` yang berisi informasi kelas dan koordinat *bounding box* sesuai dengan format YOLO. File citra dan file label harus memiliki nama yang sama agar dapat dipasangkan secara otomatis oleh sistem.

Struktur folder *dataset* disusun menjadi dua bagian utama, yaitu data pelatihan (training) dan data validasi (*validation*). Masing-masing bagian memiliki dua subfolder, yaitu folder *images* untuk menyimpan file gambar dan folder *labels* untuk menyimpan file anotasi. Secara umum, struktur folder yang digunakan adalah sebagai berikut:

Gambar 3.16 Struktur Folder *Dataset*

Selanjutnya dibuat file konfigurasi dataset dalam format YAML (`data.yaml`). File ini berisi informasi mengenai lokasi dataset, jumlah kelas yang digunakan, serta nama masing-masing kelas. Konfigurasi tersebut diperlukan agar sistem training dapat membaca dataset sesuai dengan struktur yang telah ditentukan.

```

train: ../data/images/training/
val: ../data/images/validation/
  
```

```
# number of classes
nc: 3

# class names
names: [kertas_dan_tisu, 'botol_plastik', 'kaleng']
```

Gambar 3. 17 Konfigurasi data.yaml

Proses pelatihan dilakukan dengan menjalankan skrip train.py menggunakan bobot awal (*pretrained weights*) YOLOv5s sehingga memanfaatkan metode *transfer learning*. Parameter pelatihan yang digunakan meliputi ukuran citra 416×416 piksel, *batch size* 16, dan jumlah *epoch* 100. Parameter tersebut dapat disesuaikan dengan kemampuan perangkat yang digunakan.

```
python train.py --img 416 --batch 16 --epochs 100 --data data.yaml --
weights yolov5s.pt
```

Gambar 3.18 Perintah Memulai Proses Pelatihan YOLOv5s

Penjelasan:

a. *img* 416

Ukuran citra menentukan resolusi gambar yang akan diproses oleh model. Semakin besar ukuran citra, semakin detail informasi yang diperoleh, tetapi membutuhkan komputasi lebih tinggi. Ukuran 416×416 merupakan ukuran umum pada YOLO karena memberikan keseimbangan antara akurasi dan kecepatan.

b. *Batch Size* (16)

Batch size adalah jumlah data yang diproses dalam satu kali iterasi sebelum dilakukan pembaruan bobot model. Nilai 16 berarti dalam satu iterasi model memproses 16 gambar sekaligus sebelum memperbarui parameter.

c. *Epoch* (100)

Epoch adalah jumlah perulangan penuh seluruh dataset dalam proses pelatihan. Semakin banyak *epoch*, model semakin belajar pola data, tetapi terlalu banyak epoch dapat menyebabkan *overfitting*.

Setelah *training* selesai, sistem secara otomatis menyimpan bobot terbaik dengan nama best.pt. Model ini yang kemudian digunakan pada tahap implementasi dan selanjutnya dikonversi ke format TensorFlow Lite agar dapat dijalankan pada Raspberry Pi sebagai perangkat pemrosesan utama.

3.5.2 Konversi Model ke TensorFlow Lite

Setelah proses *training* selesai dan diperoleh model terbaik dalam format best.pt, tahap selanjutnya adalah melakukan konversi model ke format TensorFlow Lite (.tflite). Proses konversi ini dilakukan karena model YOLOv5s hasil *training* masih berbasis PyTorch dan belum dapat dijalankan secara langsung pada Raspberry Pi 3B+.

```
python export.py --weights best.pt --img 416 --include tflite
```

Gambar 3.19 Perintah Konversi YOLOv5s ke TensorFlow Lite

Tahap konversi yaitu mengubah model dari format PyTorch (.pt) menjadi TensorFlow Lite (.tflite). Format TensorFlow Lite dipilih karena lebih ringan dan dioptimalkan untuk perangkat dengan sumber daya terbatas. Model dalam format .tflite selanjutnya digunakan pada tahap implementasi sistem untuk melakukan deteksi objek secara *real-time*.

3.6 Implementation and Unit Testing (Implementasi dan Pengujian unit)

Pada tahap ini dilakukan implementasi sistem pemilah sampah otomatis ke dalam bentuk nyata, yang meliputi integrasi perangkat keras dan perangkat lunak. Setiap komponen diuji secara terpisah untuk memastikan fungsinya berjalan sesuai dengan perancangan. Pengujian dilakukan terhadap sensor untuk mendeteksi keberadaan objek, kamera untuk menangkap citra sampah, model deteksi untuk mengklasifikasikan jenis sampah, serta motor servo sebagai aktuator pemisah. Proses ini bertujuan untuk memastikan bahwa masing-masing bagian sistem dapat bekerja dengan baik sebelum dilakukan pengujian secara menyeluruh.

Metode pengujian yang digunakan adalah *black box testing*, yaitu dengan memberikan input tertentu dan mengamati output yang dihasilkan tanpa meninjau kode program secara langsung. Sebagai contoh, ketika objek diletakkan di depan kamera, sistem harus mampu mendeteksi dan mengklasifikasikan sampah sesuai kategori yang telah dilatih, kemudian menggerakkan servo ke arah yang benar. Selain itu, dilakukan pengujian terhadap kondisi input yang berbeda, seperti variasi posisi objek dan pencahayaan, untuk memastikan sistem tetap memberikan respons yang sesuai. Dengan pendekatan ini, dapat diketahui apakah seluruh fungsi sistem telah berjalan sesuai spesifikasi dan tujuan perancangan.

3.7 *Integration and System Testing (Integrasi dan Pengujian Sistem)*

Setelah seluruh komponen sistem berhasil diuji secara terpisah, tahap berikutnya adalah mengintegrasikan seluruh bagian menjadi satu sistem yang utuh. Pada tahap ini dilakukan pengujian menyeluruh untuk memastikan bahwa setiap komponen, mulai dari proses deteksi sampah oleh kamera, pengklasifikasian objek oleh model, penggerakan motor servo sebagai aktuator, pengukuran volume sampah, hingga pengiriman dan penampilan data, dapat saling berkomunikasi dan bekerja secara sinkron. Pengujian ini bertujuan untuk menilai apakah alur kerja sistem telah berjalan sesuai dengan perancangan serta mampu memberikan respons yang cepat dan tepat dalam kondisi penggunaan nyata.

3.7.1 *Hardware Acceptance Testing (HAT)*

Hardware Acceptance Testing (HAT) merupakan metode pengujian yang dilakukan untuk memastikan bahwa seluruh komponen perangkat keras dan perangkat lunak yang telah diintegrasikan pada sistem dapat bekerja sesuai dengan kebutuhan yang telah ditentukan. Pengujian ini dilakukan setelah seluruh komponen sistem selesai dirancang dan diimplementasikan. Pada penelitian ini, HAT digunakan untuk menguji fungsi sistem secara menyeluruh mulai dari proses deteksi objek menggunakan webcam dan model YOLOv5s, pergerakan motor servo untuk proses pemilahan sampah, pembacaan volume tempat sampah menggunakan sensor ultrasonik, penampilan informasi pada LCD 16x2 I2C, hingga pengiriman data ke Firebase.

Pengujian dilakukan dengan memberikan beberapa skenario penggunaan yang mewakili kondisi operasional sistem. Setiap skenario dievaluasi berdasarkan kesesuaian antara hasil yang diharapkan dengan hasil aktual yang diperoleh saat sistem dijalankan. Sistem dinyatakan berhasil apabila seluruh fungsi dapat berjalan sesuai dengan kebutuhan yang telah ditetapkan.

3.7.2 *Evaluasi Kinerja dan Kelayakan Sistem*

Evaluasi ini dilakukan untuk mengetahui tingkat keberhasilan sistem dari aspek fungsional, kinerja, keandalan, dan kelayakan penerapan. Penilaian dilakukan oleh 10 responden yang telah mencoba sistem secara langsung. Responden memberikan penilaian menggunakan skala Likert 1–5, dimana nilai 1 menunjukkan sangat tidak setuju dan nilai 5 menunjukkan sangat setuju.

Aspek yang digunakan mengacu pada karakteristik kualitas sistem menurut ISO/IEC 25010 yang disesuaikan dengan tujuan penelitian, yaitu *Functional Suitability*, *Performance Efficiency*, *Reliability*, dan *Feasibility*.

Tabel 3.9 Instrumen Evaluasi

No	Pernyataan
Aspek Functional Suitability	
1	Sistem mampu mendeteksi jenis sampah dengan benar.
2	Sistem mampu mengklasifikasikan sampah sesuai kategori yang ditentukan.
3	Sistem mampu mengarahkan sampah ke tempat yang sesuai melalui motor servo.
4	Sistem mampu menampilkan informasi volume tempat sampah dengan benar.
Aspek Performance Efficiency	
5	Sistem memberikan respon deteksi dalam waktu yang cepat.
6	Pergerakan motor servo berlangsung dengan cepat setelah objek terdeteksi.
Aspek Reliability	
7	Sistem dapat beroperasi dengan stabil selama proses pengujian.
8	Hasil deteksi yang diberikan sistem konsisten pada objek yang sama.
9	Sistem dapat bekerja tanpa mengalami gangguan atau kegagalan fungsi yang signifikan.
Aspek Feasibility	
10	Sistem layak diterapkan di lingkungan kampus.
11	Sistem dapat membantu meningkatkan proses pemilahan sampah.

Hasil penilaian kemudian dihitung menggunakan rumus:

$$Persentase = \frac{\text{Skor yang diperoleh}}{\text{Skor maksimum}} \times 100\%$$

Dengan:

- Skor maksimum = jumlah responden $\times$ jumlah pertanyaan $\times$ 5
- Skor diperoleh = total seluruh jawaban responden

BAB IV

HASIL DAN PEMBAHASAN

4.1 Hasil Training Model YOLOv5s

Class	Images	Instances	P	R	mAP50	mAP50-95: 100%	12/12 [00:05<00:00, 2.30it/s]
all	376	496	0.986	0.978	0.99	0.885	
kertas_dan_tisu	376	198	0.974	0.963	0.982	0.876	
bottle_plastik	376	172	0.994	0.996	0.995	0.886	
kaleng	376	126	0.988	0.976	0.993	0.895	

Results saved to yolov5train/yolov5/runs/train/exp

Gambar 4.1 Hasil Proses Training Model YOLOv5s

Setelah proses pelatihan model YOLOv5s selesai, diperoleh hasil evaluasi performa deteksi objek berdasarkan data validasi. Berdasarkan Gambar 4.1, nilai *mean Average Precision* (mAP@0.50) mencapai 0,990 atau 99,0%, yang menunjukkan bahwa model memiliki tingkat akurasi yang sangat tinggi dalam mendeteksi objek sampah. Selain itu, nilai mAP@0.50–0.95 sebesar 0,885 atau 88,5% menunjukkan bahwa performa model tetap stabil meskipun diuji pada berbagai tingkat ambang *IoU*.

Secara keseluruhan, nilai *precision* (P) sebesar 0,986 menunjukkan bahwa hampir seluruh objek yang terdeteksi oleh sistem merupakan deteksi yang benar. Sementara itu, nilai *recall* (R) sebesar 0,978 menandakan bahwa sebagian besar objek pada *dataset* berhasil dikenali oleh model. Hal ini menunjukkan keseimbangan yang sangat baik antara ketepatan dan kelengkapan deteksi.

Berdasarkan masing-masing kelas, diperoleh hasil sebagai berikut:

- a. Kelas kertas_dan_tisu memperoleh nilai *precision* 0,974, *recall* 0,963, dan mAP@0.50 sebesar 0,982. Hal ini menunjukkan bahwa model mampu mendeteksi objek kertas dan tisu dengan tingkat akurasi yang sangat baik meskipun masih terdapat sedikit kesalahan deteksi.
- b. Kelas botol_plastik memiliki *precision* 0,994, *recall* 0,996, dan mAP@0.50 sebesar 0,995. Hasil ini menunjukkan bahwa objek botol plastik hampir seluruhnya dapat dideteksi dengan sangat akurat dan konsisten.
- c. Kelas kaleng memperoleh *precision* 0,988, *recall* 0,976, dan mAP@0.50 sebesar 0,993. Nilai ini menandakan bahwa performa deteksi untuk kelas kaleng juga sangat baik dengan tingkat kesalahan yang rendah.

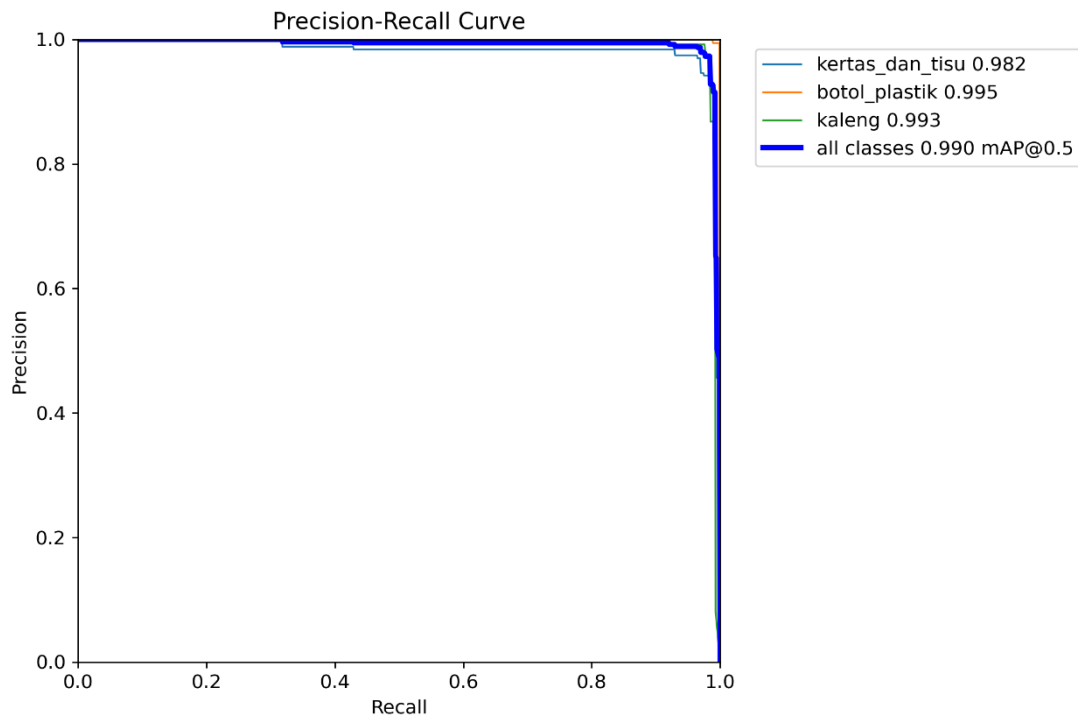


Gambar 4.2 *Confusion Matrix* Hasil Pengujian Model YOLOv5s

Gambar 4.2 menunjukkan *confusion matrix* dari hasil pengujian model YOLOv5s. *Confusion matrix* digunakan untuk melihat seberapa baik model dalam mengenali setiap jenis sampah serta mengetahui apakah masih terjadi kesalahan klasifikasi antar kelas.

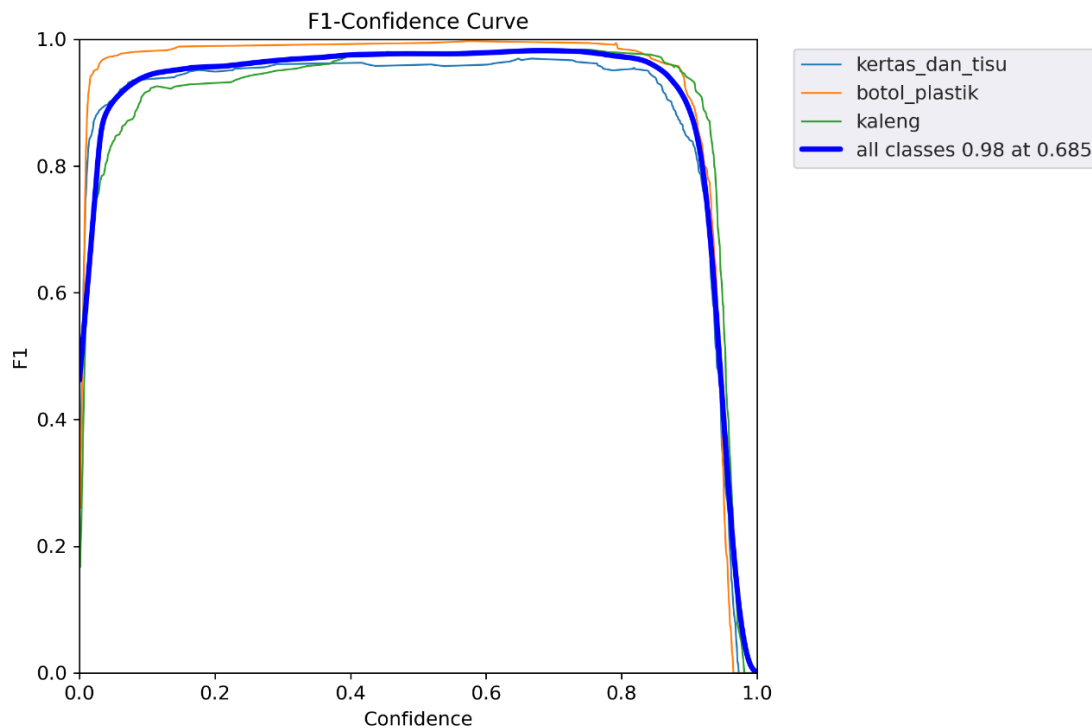
Nilai yang berada pada garis diagonal menunjukkan jumlah prediksi yang benar. Pada kelas kertas_dan_tisu, model memperoleh nilai sebesar 0,98 yang berarti 98% data berhasil dikenali dengan tepat. Pada kelas botol_plastik, diperoleh nilai 1,00 atau 100%, yang menunjukkan bahwa seluruh data pada kelas ini dapat dideteksi dengan benar tanpa kesalahan. Sedangkan pada kelas kaleng, nilai sebesar 0,98 menunjukkan bahwa hampir seluruh objek kaleng juga berhasil diklasifikasikan dengan tepat.

Sementara itu, nilai yang berada di luar diagonal menunjukkan kesalahan prediksi. Terlihat bahwa hanya terdapat nilai kecil seperti 0,02 pada beberapa bagian, yang berarti kesalahan klasifikasi yang terjadi sangat sedikit. Pada bagian *background*, terdapat beberapa nilai yang menunjukkan bahwa sebagian kecil latar belakang masih terdeteksi sebagai objek, namun jumlahnya tidak signifikan dan tidak terlalu memengaruhi kinerja model secara keseluruhan.



Gambar 4.3 Hasil *Precision-Recall Curve*

Pada Gambar 4.3 menunjukkan hubungan antara *precision* dan *recall* dalam proses deteksi objek menggunakan model YOLOv5s. *Precision* menunjukkan seberapa tepat hasil deteksi yang dilakukan model, sedangkan *recall* menunjukkan seberapa banyak objek yang berhasil dikenali dari seluruh objek yang ada pada data uji. Nilai *Average Precision (AP)* untuk masing-masing kelas terlihat sangat baik, yaitu 0,982 untuk kertas_dan_tisu, 0,995 untuk botol_plastik, dan 0,993 untuk kaleng. Selain itu, nilai *mean Average Precision (mAP@0.5)* untuk seluruh kelas mencapai 0,990 atau 99%.



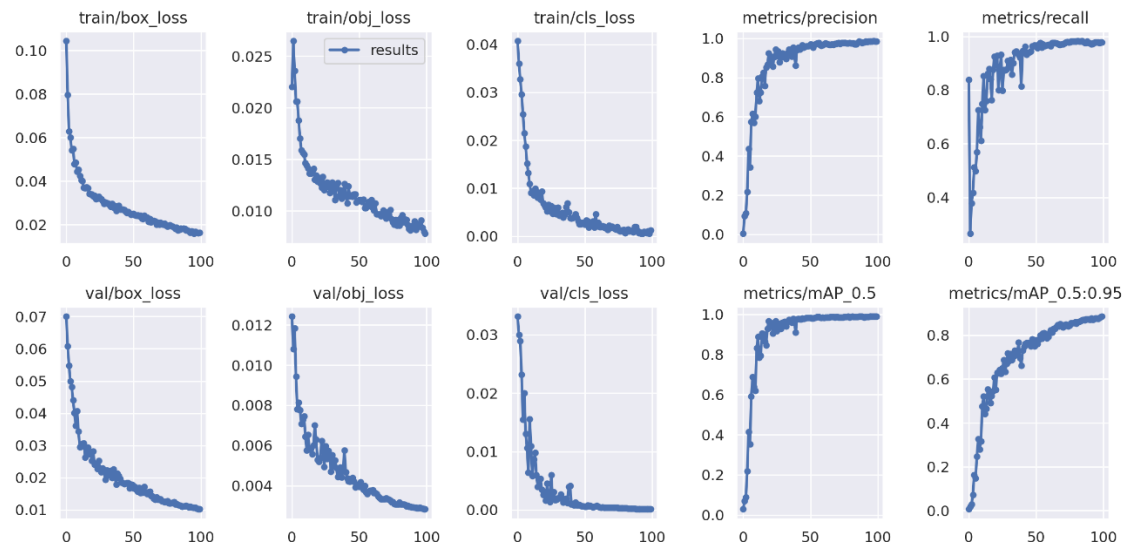
Gambar 4.4 Hasil *F1-Confidence Curve*

Pada Gambar 4.4 menunjukkan hubungan antara nilai *confidence threshold* dengan nilai F1-score. *Confidence* adalah batas keyakinan model dalam menentukan apakah suatu objek benar-benar terdeteksi atau tidak. Sedangkan F1-score merupakan nilai gabungan dari *precision* dan *recall* yang digunakan untuk melihat keseimbangan antara ketepatan dan kelengkapan deteksi. Dari grafik terlihat bahwa nilai F1 meningkat dengan cepat pada *confidence* rendah, kemudian stabil pada nilai yang tinggi.

Nilai F1 tertinggi untuk seluruh kelas berada di sekitar *confidence* 0,685 dengan nilai F1 sebesar 0,98. Hal ini menunjukkan bahwa model bekerja paling optimal ketika *confidence* diatur pada kisaran tersebut. Jika *confidence* terlalu tinggi, nilai F1 menurun karena model menjadi terlalu selektif sehingga beberapa objek tidak terdeteksi.

Secara umum, hasil training menunjukkan bahwa model YOLOv5s memiliki performa yang sangat baik dalam mendeteksi dan mengklasifikasikan objek sampah. Nilai akurasi yang tinggi menunjukkan bahwa sebagian besar objek berhasil dikenali dengan tepat dan hanya terdapat sedikit kesalahan dalam proses deteksi. Model juga mampu menjaga keseimbangan antara ketepatan hasil deteksi dan kemampuan menemukan seluruh objek pada data uji.

4.1.1 Analisis Overfitting Model



Gambar 4.5 Training Results

Selain mengevaluasi nilai precision, recall, dan mAP, dilakukan juga analisis overfitting untuk mengetahui apakah model hanya menghafal data pelatihan atau mampu melakukan generalisasi dengan baik terhadap data baru. Analisis ini dilakukan dengan mengamati grafik training loss dan validation loss yang dihasilkan selama proses pelatihan model YOLOv5s.

Berdasarkan grafik hasil pelatihan, nilai train box loss mengalami penurunan dari sekitar 0,105 pada awal pelatihan menjadi sekitar 0,017 pada akhir pelatihan. Pada saat yang sama, nilai validation box loss juga menurun dari sekitar 0,070 menjadi sekitar 0,010. Penurunan yang terjadi pada kedua kurva tersebut menunjukkan bahwa model semakin mampu mengenali posisi objek dengan baik baik pada data pelatihan maupun data validasi. Selain itu, nilai train object loss dan validation object loss juga menunjukkan tren penurunan yang stabil hingga akhir *epoch*, yang mengindikasikan bahwa model semakin baik dalam membedakan objek dan latar belakang.

Hal yang sama juga terlihat pada train classification loss dan validation classification loss yang terus menurun hingga mendekati nilai nol. Penurunan ini menunjukkan bahwa kemampuan model dalam membedakan setiap kelas sampah, yaitu kertas dan tisu, botol plastik, serta kaleng semakin meningkat selama proses pelatihan. Tidak ditemukan adanya peningkatan yang signifikan pada validation loss di akhir proses pelatihan. Kondisi ini berbeda dengan karakteristik overfitting, dimana biasanya training loss terus menurun sementara validation loss mulai meningkat karena model terlalu menyesuaikan diri terhadap data pelatihan

Selain analisis *loss*, evaluasi juga dilakukan melalui metrik performa model. Grafik precision menunjukkan peningkatan hingga mencapai nilai sekitar 0,986, sedangkan recall mencapai nilai sekitar 0,978. Nilai $mAP@0.5$ terus meningkat hingga mencapai 0,99 dan $mAP@0.5:0.95$ mencapai 0,885. Peningkatan yang konsisten pada metrik-metrik tersebut menunjukkan bahwa model tidak hanya mampu mengenali data pelatihan dengan baik, tetapi juga mampu mempertahankan performa yang tinggi pada data validasi.

Hasil ini diperkuat oleh confusion matrix yang menunjukkan bahwa sebagian besar objek berhasil diklasifikasikan dengan benar. Nilai diagonal pada setiap kelas mendekati 1, yaitu sebesar 0,98 untuk kelas kertas dan tisu, 1,00 untuk kelas botol plastik, dan 0,98 untuk kelas kaleng. Nilai tersebut menunjukkan bahwa kesalahan klasifikasi yang terjadi relatif kecil dan model mampu membedakan karakteristik masing-masing kelas dengan baik.

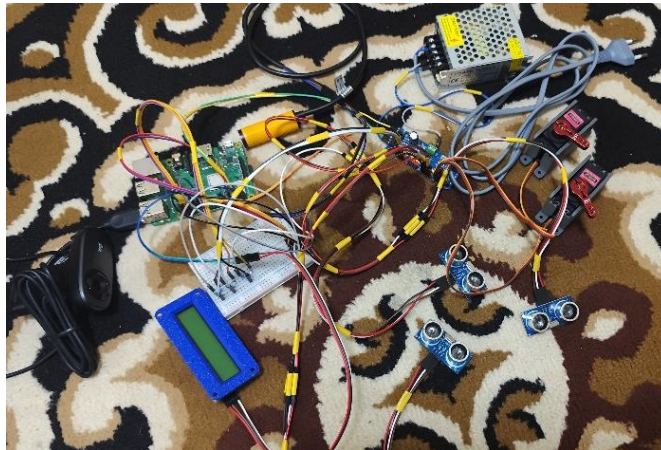
Berdasarkan hasil analisis training loss, validation loss, precision, recall, mAP , dan confusion matrix, dapat disimpulkan bahwa model YOLOv5s pada penelitian ini tidak menunjukkan indikasi terjadinya overfitting. Model mampu mempelajari pola data secara efektif serta memiliki kemampuan generalisasi yang baik terhadap data yang belum pernah dilihat sebelumnya, sehingga layak digunakan pada sistem pemilah sampah otomatis yang dikembangkan.

4.2 Implementasi Sistem

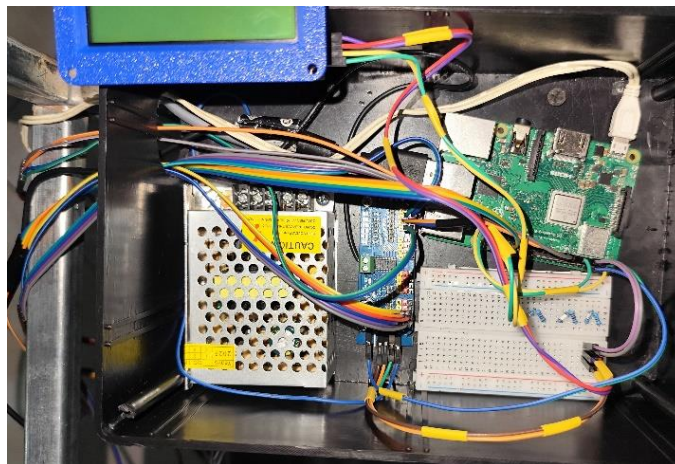
Implementasi sistem menjelaskan hasil penerapan dari seluruh proses perancangan yang telah dibahas pada Bab III. Tahapan ini dilakukan dengan merealisasikan rancangan perangkat keras dan perangkat lunak ke dalam sistem yang dapat berfungsi secara nyata. Seluruh komponen sistem diintegrasikan dan dijalankan sesuai dengan desain yang telah ditentukan untuk memastikan sistem bekerja sebagaimana yang diharapkan.

4.2.1 Implementasi Perangkat Keras

Implementasi perangkat keras dilakukan berdasarkan rancangan rangkaian dan konfigurasi pin yang telah dijelaskan pada Bab III. Seluruh komponen yang diperlukan telah dipasang dan dihubungkan sesuai dengan perancangan yang telah dibuat. Hasil implementasi perangkat keras ditunjukkan pada Gambar 4.6 dan Gambar 4.7.



Gambar 4.6 Implementasi Rancangan Rangkaian Perangkat Keras



Gambar 4.7 Penempatan Perangkat Keras pada Kotak

Pada tahap ini, Raspberry Pi 3B+ berfungsi sebagai pusat pengendali utama yang terhubung dengan kamera melalui port USB untuk proses pengambilan citra. Kamera diposisikan pada bagian atas area masuk sampah untuk menangkap citra objek sebelum dilakukan proses klasifikasi. Hasil deteksi kemudian diproses untuk menentukan arah pemilahan yang sesuai.



Gambar 4.8 Kondisi Sistem Setengah Jadi

Motor servo MG996R dikendalikan melalui modul PCA9685 yang menggunakan komunikasi I2C. Penggunaan modul ini bertujuan untuk menghasilkan sinyal PWM yang stabil sekaligus meminimalkan penggunaan GPIO pada Raspberry Pi. Pergerakan servo dirancang untuk mengarahkan sampah ke salah satu dari tiga bak penampungan sesuai hasil klasifikasi. Selain itu, LCD berbasis I2C dipasang pada bagian luar sistem untuk menampilkan informasi jenis sampah yang terdeteksi serta status operasional sistem secara *real-time*.



Gambar 4.9 Kondisi Sistem yang Telah Siap

Untuk memantau kapasitas masing-masing bak sampah, sistem dilengkapi dengan tiga sensor ultrasonik HC-SR04 yang dipasang pada bagian atas setiap bak sampah. Sensor ini berfungsi untuk mengukur jarak antara permukaan sampah dan sensor, sehingga sistem dapat memperkirakan tingkat kepenruhan setiap bak. Data yang diperoleh dari sensor ultrasonik kemudian diproses oleh Raspberry Pi dan dapat ditampilkan pada LCD maupun dikirimkan ke sistem monitoring (Firebase).

4.2.2 Implementasi Perangkat Lunak

Pada tahap ini, seluruh rancangan perangkat lunak yang telah dijelaskan pada bab sebelumnya diimplementasikan secara nyata. Perangkat lunak dikembangkan menggunakan bahasa pemrograman Python dan dijalankan pada Raspberry Pi sebagai pengendali utama sistem. *Software* ini berfungsi untuk menerima input dari kamera, melakukan proses deteksi objek, mengolah hasil deteksi, serta mengirimkan perintah ke aktuator dan sensor yang terhubung.

a. *Library* Pendukung

```
import cv2
import numpy as np
import time
from threading import Thread, Lock
from datetime import datetime
```

```

from RPLCD.i2c import CharLCD
from adafruit_servokit import ServoKit
import RPi.GPIO as GPIO

import firebase_admin
from firebase_admin import credentials, db

```

Gambar 4.10 Fungsi *Library* Pendukung

Library pendukung digunakan untuk membantu proses pengembangan sistem, baik dalam pengolahan data maupun pengendalian perangkat keras. Pada penelitian ini, OpenCV (cv2) digunakan untuk pengolahan citra dari kamera, sedangkan NumPy (numpy) digunakan untuk perhitungan numerik. *Library time* dan *datetime* digunakan untuk pengaturan waktu.

Untuk menjalankan beberapa proses secara bersamaan, digunakan *threading* (*Thread* dan *Lock*). Pengendalian perangkat keras dilakukan menggunakan RPLCD untuk LCD, Adafruit ServoKit untuk servo motor, dan RPi.GPIO untuk sensor ultrasonik. Selain itu, Firebase Admin digunakan untuk mengirim data ke *database* secara *real-time*.

b. *Image Detection* (TensorFlow Lite)

```

MODEL_PATH = "best-model.tflite"
LABEL_PATH = "labels.txt"

CONF_THRES = 0.5
MIN_CONF = 0.55

with open(LABEL_PATH) as f:
    class_names = [c.strip() for c in f.readlines()]

from tflite_runtime.interpreter import Interpreter

interpreter = Interpreter(model_path=MODEL_PATH)
interpreter.allocate_tensors()

input_details = interpreter.get_input_details()
output_details = interpreter.get_output_details()

IMG_H = input_details[0]['shape'][1]
IMG_W = input_details[0]['shape'][2]

def preprocess(frame):
    img = cv2.resize(frame, (IMG_W, IMG_H))
    img = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
    img = img.astype(np.float32) / 255.0
    return np.expand_dims(img, axis=0)

def detect(frame):
    input_data = preprocess(frame)
    interpreter.set_tensor(input_details[0]['index'], input_data)

```

```

interpreter.invoke()

output = interpreter.get_tensor(output_details[0]['index'])[0]

results = []
for d in output:
    if d[4] < CONF_THRES:
        continue
    cls = np.argmax(d[5:])
    conf = float(d[4]) * float(d[5 + cls])
    if conf >= MIN_CONF:
        results.append((conf, class_names[cls]))

return results

```

Gambar 4.11 Fungsi *Image Detection*

Modul ini adalah bagian utama yang berfungsi untuk mengenali jenis sampah secara otomatis menggunakan kamera. Sistem memanfaatkan model TensorFlow Lite (TFLite) yang sebelumnya telah dilatih menggunakan *dataset* citra sampah. Pada tahap awal, sistem memuat file model (.tflite) dan file label yang berisi nama kelas. Setelah model berhasil dimuat, sistem mengambil informasi terkait ukuran *input* gambar yang dibutuhkan oleh model.

Ketika kamera menangkap gambar, data tersebut tidak langsung diproses, tetapi terlebih dahulu melalui tahap *preprocessing*. Proses ini meliputi:

1. Mengubah ukuran gambar sesuai input model
2. Mengubah format warna dari BGR ke RGB
3. Menormalisasi nilai piksel agar sesuai dengan standar model

Setelah *preprocessing*, gambar dimasukkan ke dalam model untuk dilakukan proses *inference* (prediksi). Model akan menghasilkan output berupa beberapa kandidat deteksi yang memiliki nilai kepercayaan (*confidence*). Sistem kemudian melakukan penyaringan dengan menggunakan batas minimum *confidence*, sehingga hanya hasil dengan tingkat keyakinan tinggi yang digunakan. Hasil akhir modul ini adalah label jenis sampah yang terdeteksi, yang nantinya akan digunakan oleh modul lain seperti servo dan LCD.

c. Sensor Ultrasonik (Ukur Volume Bak Sampah)

```

GPIO.setmode(GPIO.BCM)
GPIO.setwarnings(False)

ULTRASONIC = {
    "kertas_dan_tisu": {"trig": 21, "echo": 20, "max": 30},
    "botol_plastik": {"trig": 22, "echo": 27, "max": 30},
    "kaleng": {"trig": 6, "echo": 5, "max": 30}
}

```

```

for u in ULTRASONIC.values():
    GPIO.setup(u["trig"], GPIO.OUT)
    GPIO.setup(u["echo"], GPIO.IN)
    GPIO.output(u["trig"], False)

def measure_distance(trig, echo):
    GPIO.output(trig, True)
    time.sleep(0.00001)
    GPIO.output(trig, False)

    while GPIO.input(echo) == 0:
        pulse_start = time.time()

    while GPIO.input(echo) == 1:
        pulse_end = time.time()

    distance = (pulse_end - pulse_start) * 34300 / 2
    return distance

def calculate_volume(distance, max_height):
    volume = max(0, min(100, int((1 - distance / max_height) * 100)))

    if volume < 30:
        status = "kosong"
    elif volume < 70:
        status = "sedang"
    else:
        status = "penuh"

    return volume, status

```

Gambar 4.12 Fungsi Ukur Volume Bak Sampah

Modul ini digunakan untuk mengukur volume sampah yang ada di dalam bak sampah menggunakan sensor ultrasonik. Sensor bekerja dengan prinsip gelombang suara, yaitu mengirimkan sinyal ultrasonik dan menerima pantulannya dari permukaan sampah. Proses kerja dimulai dengan mengaktifkan pin trigger untuk mengirim sinyal, kemudian sistem menunggu pantulan diterima oleh pin echo. Waktu yang dibutuhkan sinyal untuk kembali digunakan untuk menghitung jarak antara sensor dan permukaan sampah. Selain itu, sistem juga mengelompokkan kondisi volume menjadi tiga kategori:

1. Kosong (< 30%)
2. Sedang (30%–70%)
3. Penuh (> 70%)

Output dari modul ini berupa nilai volume (%) dan status kondisi tempat sampah, yang kemudian digunakan untuk ditampilkan ke LCD dan dikirim ke Firebase.

d. Servo (Penggerak Pemilah)

```

from adafruit_servokit import ServoKit

kit = ServoKit(channels=16)

SERVO_MAIN = 0
SERVO_SECOND = 1

SERVO_CENTER = 40
SERVO_CENTER_SECOND = 80
SERVO_LEFT = 0
SERVO_RIGHT = 130
SERVO_RIGHT_2 = 140

kit.servo[SERVO_MAIN].angle = SERVO_CENTER
kit.servo[SERVO_SECOND].angle = SERVO_CENTER_SECOND

def move_servo(label):
    if label == "kertas_dan_tisu":
        kit.servo[SERVO_MAIN].angle = SERVO_RIGHT
    else:
        kit.servo[SERVO_MAIN].angle = SERVO_LEFT

    time.sleep(1)

    if label == "botol_plastik":
        kit.servo[SERVO_SECOND].angle = SERVO_RIGHT_2
    elif label == "kaleng":
        kit.servo[SERVO_SECOND].angle = SERVO_LEFT

    time.sleep(1)

    kit.servo[SERVO_MAIN].angle = SERVO_CENTER
    kit.servo[SERVO_SECOND].angle = SERVO_CENTER_SECOND

```

Gambar 4.13 Fungsi Penggerak Pemilah Sampah (Motor Servo)

Modul ini berfungsi sebagai mekanisme pemilah sampah secara fisik. Servo motor digunakan untuk menggerakkan sampah agar masuk ke bak yang sesuai berdasarkan hasil deteksi. Sistem menggunakan dua servo, servo utama berfungsi untuk menentukan arah awal (sampah kertas dan tisu atau non-kertas dan tisu). Servo kedua berfungsi untuk melakukan pemilahan lanjutan (plastik atau kaleng). Setelah proses pemilahan selesai, servo akan kembali ke posisi awal (*default*) agar siap untuk proses berikutnya. Penggunaan delay pada pergerakan servo bertujuan agar gerakan lebih stabil dan tidak terlalu cepat.

e. LCD 16x2 I2C (Informasi Hasil Deteksi)

```
from RPLCD.i2c import CharLCD

lcd = CharLCD('PCF8574', 0x27, port=1, cols=16, rows=2)

def display_lcd(label, volume, status):
    lcd.clear()
    lcd.write_string(label[:16])
    lcd.cursor_pos = (1, 0)
    lcd.write_string(f"{volume}% {status}")

def lcd_idle():
    lcd.clear()
    lcd.write_string("Tidak ada objek")
```

Gambar 4.14 Fungsi Informasi Hasil Deteksi

Modul LCD digunakan sebagai media informasi langsung kepada pengguna. LCD akan menampilkan hasil deteksi sistem sehingga pengguna dapat mengetahui jenis sampah yang sedang diproses. Ada dua baris informasi yang ditampilkan, baris pertama menampilkan jenis sampah. Baris kedua menampilkan volume tempat sampah dan statusnya (misalnya: 75% penuh). Selain itu, sistem juga memiliki kondisi *idle*, yaitu ketika tidak ada objek yang terdeteksi oleh kamera. Pada kondisi ini, LCD akan menampilkan pesan seperti “Tidak ada objek” agar pengguna mengetahui bahwa sistem sedang *standby*.

f. Inisialisasi Firebase

```
cred = credentials.Certificate("firebase-key-fix.json")
firebase_admin.initialize_app(cred, {
    "databaseURL": "https://smart-trash-257c1-default-
rtadb.firebaseio.com/"
})

fb_ref = db.reference("raspberry")

def send_to_firebase(data):
    fb_ref.update(data)
```

Gambar 4.15 Fungsi Inisialisasi Firebase

Modul ini digunakan untuk mengirim data ke *database* secara *online (real-time)* menggunakan Firebase. Data yang dikirim meliputi:

1. Jenis sampah
2. Volume tempat sampah
3. Status (kosong, sedang, penuh)
4. Waktu pembacaan

Sistem terlebih dahulu melakukan inisialisasi koneksi ke Firebase menggunakan file kredensial. Setelah terhubung, data akan dikirim menggunakan fungsi update ke *database*. Keunggulan penggunaan Firebase adalah memungkinkan pengguna atau pengelola untuk memantau kondisi tempat sampah dari jarak jauh, tanpa harus melihat langsung ke lokasi.

g. Program Utama

```
while True:
    frame = camera.read()
    detections = detect(frame)

    if detections:
        label = detections[0][1]

        move_servo(label)

        sensor = ULTRASONIC[label]
        distance = measure_distance(sensor["trig"], sensor["echo"])
        volume, status = calculate_volume(distance, sensor["max"])

        display_lcd(label, volume, status)

        send_to_firebase({
            "jenis": label,
            "volume": volume,
            "status": status
        })

    else:
        lcd_idle()
```

Gambar 4.16 Fungsi Program Utama

Modul ini merupakan bagian inti yang mengintegrasikan seluruh modul dalam sistem. Program utama berjalan secara terus-menerus (*loop*) dan mengatur alur kerja sistem secara keseluruhan. Alur kerja sistem adalah sebagai berikut:

1. Kamera menangkap gambar secara *real-time*
2. Gambar dikirim ke modul image detection untuk menentukan jenis sampah
3. Jika objek terdeteksi:
 - a) Servo bergerak untuk memilah sampah
 - b) Sensor ultrasonik membaca volume tempat sampah
 - c) LCD menampilkan hasil deteksi dan volume
 - d) Data dikirim ke Firebase

4. Jika tidak ada objek:
 - a) Sistem berada dalam kondisi *standby*
 - b) LCD menampilkan pesan “Tidak ada objek”

Semua proses ini berjalan secara otomatis tanpa intervensi pengguna, sehingga sistem mampu melakukan pemilahan sampah secara *real-time* dan efisien.

4.3 Pengujian Unit

Pengujian unit dilakukan untuk memastikan bahwa setiap komponen sistem pemilah sampah otomatis dapat bekerja dengan baik secara mandiri sebelum diintegrasikan menjadi satu kesatuan sistem. Pada tahap ini, masing-masing bagian diuji secara terpisah, mulai dari webcam untuk memastikan kemampuan menangkap gambar dengan jelas, model deteksi untuk menguji proses klasifikasi objek, motor servo untuk memastikan pergerakan sesuai perintah, sensor volume untuk membaca kapasitas tempat sampah, hingga modul pengiriman data untuk memastikan koneksi ke Firebase berjalan dengan baik.

Pengujian *Black Box* dilakukan untuk memastikan bahwa setiap fungsi pada sistem pemilah sampah otomatis dapat berjalan sesuai dengan. Pengujian ini berfokus pada hubungan antara *input* yang diberikan dan *output* yang dihasilkan oleh sistem. Pengujian dilakukan dengan memberikan berbagai jenis objek sampah sebagai *input* untuk melihat apakah sistem mampu mendeteksi dan mengklasifikasikan objek dengan benar, serta menggerakkan motor servo sesuai kategori yang teridentifikasi. Selain itu, dilakukan juga pengujian terhadap respons sensor, pengiriman data (Firebase), dan tampilan informasi pada sistem.

Tabel 4.1 Skenario *Black Box Testing*

No	Fungsi yang Diuji	Skenario Pengujian	Hasil yang Diharapkan
1	Object Detection	Objek kertas dan tisu diletakkan di depan kamera	Sistem mendeteksi objek sebagai kertas dan tisu
2	Object Detection	Objek botol plastik diletakkan di depan kamera	Sistem mendeteksi objek sebagai botol plastik
3	Object Detection	Objek kaleng diletakkan di depan kamera	Sistem mendeteksi objek sebagai kaleng
4	LCD 16x2 I2C	Sistem berhasil mendeteksi objek	LCD menampilkan jenis sampah dan informasi volume
5	LCD 16x2 I2C	Tidak ada objek di depan kamera	LCD menampilkan pesan "Tidak ada objek"
6	Motor Servo	Kertas dan tisu terdeteksi	Servo bergerak ke jalur kertas dan tisu

7	Motor Servo	Botol plastik terdeteksi	Servo bergerak ke jalur botol plastik
8	Motor Servo	Kaleng terdeteksi	Servo bergerak ke jalur kaleng
9	Sensor Ultrasonik	Volume sampah bertambah	Nilai volume dan kategori berubah sesuai kondisi aktual
10	Firestore	Sistem berhasil mendeteksi objek	Data deteksi dan volume terkirim ke Firestore

4.3.1 Uji Fungsi *Object Detection*

Uji fungsi *object detection* dilakukan untuk memastikan bahwa sistem mampu mendeteksi dan mengklasifikasikan objek sampah sesuai dengan kategori yang telah dilatih pada model. Pengujian ini dilakukan dengan meletakkan berbagai jenis sampah di depan *webcam*, kemudian mengamati apakah sistem dapat mengenali objek tersebut dan menampilkan label klasifikasi yang benar. Proses ini bertujuan untuk mengetahui tingkat ketepatan deteksi model dalam kondisi penggunaan nyata.



Gambar 4.17 Sampah yang Terdeteksi dan Akurasi-nya

Dalam pengujian ini, beberapa objek dari masing-masing kategori diuji secara bergantian untuk melihat konsistensi hasil deteksi. Hasil yang diamati meliputi kesesuaian antara objek yang diberikan sebagai input dengan label yang ditampilkan oleh sistem. Pengujian ini bertujuan untuk mengetahui kemampuan model dalam mengenali dan mengklasifikasikan objek sampah sesuai dengan kategori yang telah ditentukan. Apabila label yang ditampilkan sesuai dengan jenis sampah yang diuji, maka fungsi *object detection* dinyatakan berhasil. Hasil pengujian disajikan dalam Tabel 4.2 untuk mempermudah analisis dan evaluasi kinerja sistem.

Tabel 4.2 Pengujian *Object Detection*

No	Input Objek	Hasil Deteksi	Akurasi	Status
1	Kotak susu Frisian Flag strawberry	Kertas dan tisu	0.96	Berhasil
2	Gumpalan tisu berwarna coklat	Kertas dan tisu	0.94	Berhasil
3	Gumpalan kertas hvs	Kertas dan tisu	0.94	Berhasil
4	Gumpalan tisu berwarna putih	Kertas dan tisu	0.78	Berhasil
5	Kotak minuman hydro coco	Kertas dan tisu	0.95	Berhasil
6	Kardus coklat kecil	Tidak terdeteksi	-	Gagal
7	Botol plastik le minerale	Botol plastik	0.91	Berhasil
8	Botol plastik Point Coffee	Botol plastik	0.95	Berhasil
9	Botol kaca saos ABC	Botol plastik	0.95	Gagal
10	Botol kaca bekas obat	Botol plastik	0.90	Gagal
11	Teh botol sosro	Botol plastik	0.87	Berhasil
12	Minuman yakult	Botol plastik	0.81	Berhasil
13	Botol plastik kopi Golda	Botol plastik	0.94	Berhasil
14	Minuman kaleng Bear Brand	Kaleng	0.95	Berhasil
15	Minuman kaleng Milo	Kaleng	0.96	Berhasil
16	Minuman kaleng cap kaki tiga	Kaleng	0.93	Berhasil
17	Lifecat kaleng makanan kucing	Kaleng	0.90	Berhasil
18	Minuman Teh Gelas	Kaleng	0.84	Gagal

Berdasarkan hasil pengujian pada Tabel 4.2, sistem object detection yang menggunakan model YOLOv5s mampu mendeteksi dan mengklasifikasikan sebagian besar objek sampah sesuai dengan kategori yang telah ditentukan, yaitu kertas dan tisu, botol plastik, serta kaleng. Dari total 18 objek yang diuji, sebanyak 14 objek berhasil dideteksi dengan benar, sedangkan 4 objek mengalami kegagalan deteksi atau salah klasifikasi. Dengan demikian, tingkat keberhasilan sistem mencapai 77,78%.

Pada kategori kertas dan tisu, sebagian besar objek berhasil dideteksi dengan nilai confidence antara 0,78 hingga 0,96. Namun, objek berupa kardus coklat kecil tidak berhasil terdeteksi. Hal ini kemungkinan disebabkan oleh karakteristik visual kardus yang kurang terwakili dalam dataset pelatihan, sehingga model kesulitan mengenali objek tersebut sebagai bagian dari kategori kertas dan tisu.

Pada kategori botol plastik, sebagian besar objek berhasil dideteksi dengan baik. Akan tetapi, botol kaca saos ABC dan botol kaca bekas obat terklasifikasi sebagai botol plastik. Kesalahan ini terjadi karena bentuk objek yang menyerupai botol plastik, sementara model hanya dilatih menggunakan tiga kelas sampah tanpa kelas khusus untuk botol kaca. Kondisi

tersebut menyebabkan model mengelompokkan objek ke kelas yang memiliki kemiripan bentuk paling tinggi.

Pada kategori kaleng, sebagian besar objek berhasil dikenali dengan nilai confidence yang cukup tinggi, yaitu antara 0,90 hingga 0,96. Namun, objek minuman Teh Gelas tidak berhasil diklasifikasikan dengan benar karena memiliki bentuk kemasan gelas plastik yang berbeda dengan karakteristik objek kaleng yang digunakan saat pelatihan model.

Secara keseluruhan, hasil pengujian menunjukkan bahwa model YOLOv5s memiliki kemampuan yang baik dalam mendeteksi objek yang memiliki karakteristik serupa dengan dataset pelatihan. Namun, performa model masih dapat ditingkatkan dengan menambahkan variasi data latih yang lebih beragam, terutama untuk objek yang memiliki bentuk atau karakteristik visual yang mirip antar kategori. Penambahan dataset yang lebih representatif diharapkan dapat meningkatkan kemampuan generalisasi model sehingga kesalahan klasifikasi dapat diminimalkan.

4.3.2 Uji LCD 16x2 I2C

Pengujian LCD 16x2 I2C dilakukan untuk memastikan bahwa LCD dapat menampilkan informasi dengan baik sesuai dengan kondisi sistem. LCD digunakan sebagai media untuk memberikan informasi kepada pengguna mengenai hasil deteksi jenis sampah serta kondisi volume tempat sampah.



Gambar 4.18 Pengujian LCD ketika Sistem Berhasil Mendeteksi Sampah

Pada pengujian ini, sistem dijalankan dan beberapa objek sampah diuji. Ketika sistem berhasil mendeteksi jenis sampah, LCD akan menampilkan nama jenis sampah pada baris pertama. Selanjutnya, pada baris kedua ditampilkan informasi volume dalam bentuk persentase serta status kondisi tempat sampah, seperti kosong, sedang, atau penuh.



Gambar 4.19 Pengujian LCD ketika Tidak Ada Objek yang Terdeteksi Sistem

Selain itu, ketika tidak ada objek yang terdeteksi oleh sistem, LCD akan menampilkan pesan “Tidak ada objek” sebagai tanda bahwa sistem sedang dalam kondisi *standby*. Berdasarkan hasil pengujian, LCD dapat menampilkan informasi dengan jelas dan sesuai dengan kondisi sistem secara *real-time*.

4.3.3 Uji Respon Servo

Uji respon servo dilakukan untuk memastikan bahwa motor servo dapat bergerak sesuai dengan perintah yang diberikan oleh sistem setelah proses deteksi dan klasifikasi objek selesai dilakukan. Pengujian ini bertujuan untuk mengetahui apakah setiap kategori sampah yang terdeteksi menghasilkan gerakan servo yang tepat, baik dari segi arah pergerakan maupun posisi sudut yang telah ditentukan dalam perancangan sistem.

Pada tahap ini, beberapa objek sampah dari masing-masing kategori diuji secara bergantian. Setelah sistem berhasil mengklasifikasikan objek, diamati apakah servo bergerak ke arah yang sesuai, serta dapat kembali ke posisi awal setelah proses pemilahan selesai. Selain itu, dilakukan juga pengamatan terhadap kestabilan gerakan servo untuk memastikan tidak terjadi getaran berlebihan atau pergerakan yang tidak konsisten. Hasil pengujian disajikan dalam bentuk Tabel 4.3.

Tabel 4.3 Pengujian Respon Servo

No	Input Deteksi	Perintah Sistem	Gerakan Servo	Status
1	Kertas dan tisu	Servo 1 aktif	Bergerak ke kiri	Berhasil
2	Kertas dan tisu	Servo 1 aktif	Bergerak ke kiri	Berhasil
3	Botol plastik	Servo 1 & 2 aktif (bergerak bergantian)	Servo 1 bergerak ke kanan, lalu servo 2 bergerak ke kiri	Berhasil
4	Botol plastik	Servo 1 & 2 aktif (bergerak bergantian)	Servo 1 bergerak ke kanan, lalu servo 2 bergerak ke kiri	Berhasil
5	Kaleng	Servo 1 & 2 aktif	Servo 1 bergerak ke kanan, lalu servo 2 bergerak ke kanan	Berhasil

		(bergerak bergantian)		
6	Kaleng	Servo 1 & 2 aktif (bergerak bergantian)	Servo 1 bergerak ke kanan, lalu servo 2 bergerak ke kanan	Berhasil

Berdasarkan hasil pengujian yang ditunjukkan pada Tabel 4.2, dapat diketahui bahwa motor servo mampu merespon perintah sistem dengan baik sesuai dengan hasil deteksi objek. Pada kategori kertas dan tisu, servo 1 bergerak ke arah kiri secara konsisten sesuai dengan perancangan sistem. Sementara itu, pada kategori botol plastik dan kaleng, servo 1 dan servo 2 bekerja secara bergantian untuk mengarahkan sampah ke bak yang sesuai, di mana pergerakan masing-masing servo juga berjalan dengan tepat. Selain itu, tidak ditemukan adanya keterlambatan respon maupun pergerakan yang tidak stabil selama pengujian berlangsung. Dengan demikian, dapat disimpulkan bahwa sistem penggerak servo telah berfungsi dengan baik dan mampu menjalankan proses pemilahan sampah secara otomatis sesuai dengan kategori yang terdeteksi.

4.3.4 Uji Pengukuran Volume Bak Sampah

Uji pengukuran volume bak sampah dilakukan untuk memastikan bahwa sensor ultrasonik yang digunakan mampu membaca tingkat ketinggian atau kapasitas sampah di dalam bak dengan akurat. Pengujian ini bertujuan untuk mengetahui apakah sistem dapat mendeteksi perubahan volume sampah dan menampilkan informasi yang sesuai, baik dalam bentuk persentase maupun kategori seperti kosong, setengah penuh (sedang), atau penuh.

Pada tahap ini, pengujian dilakukan dengan menambahkan sampah secara bertahap ke dalam wadah, kemudian mencatat hasil pembacaan sensor yang ditampilkan oleh sistem. Hasil tersebut dibandingkan dengan kondisi nyata di dalam tempat sampah untuk melihat tingkat kesesuaian pengukuran. Hasil pengujian disajikan dalam bentuk Tabel 4.4.

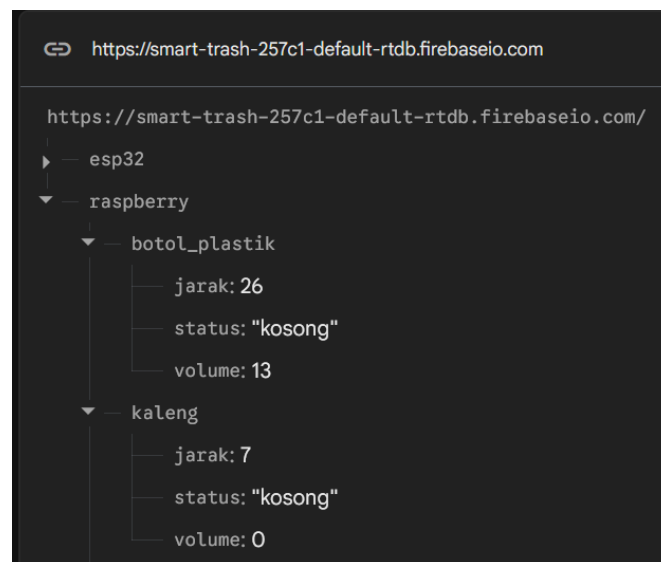
Tabel 4.4 Pengujian Volume Bak Sampah

No	Kondisi Aktual Bak Sampah (cm)	Jarak Terbaca Sensor (cm)	Persentase Volume (%)	Kategori Volume	Akurasi (%)
1	5	5.2	17.33	Kosong	96
2	10	9.6	32.00	Kosong	96
3	15	16.2	54.00	Sedang	92
4	20	19.1	63.67	Sedang	95.50
5	25	26.8	89.33	Penuh	92.80
6	30	28.6	95.33	Penuh	95.33

Berdasarkan hasil pada Tabel 4.4, sensor ultrasonik mampu mengukur volume sampah dengan cukup baik, dengan nilai akurasi berkisar antara 92% hingga 96%. Persentase volume yang dihasilkan juga menunjukkan peningkatan yang sesuai dengan bertambahnya jumlah sampah di dalam bak, sehingga dapat dikatakan bahwa sistem mampu mendeteksi perubahan volume dengan cukup akurat. Selain itu, kategori volume seperti kosong, sedang, dan penuh sudah sesuai dengan kondisi sebenarnya. Meskipun demikian, masih terdapat sedikit selisih pada beberapa pengukuran, terutama pada jarak yang lebih jauh. Hal ini kemungkinan dipengaruhi oleh posisi sensor, permukaan objek, serta pantulan gelombang ultrasonik. Secara keseluruhan, sensor ultrasonik sudah cukup andal untuk digunakan dalam memantau volume sampah secara *real-time*.

4.3.5 Uji Pengiriman Data (Firebase)

Uji pengiriman data dilakukan untuk memastikan bahwa sistem mampu mengirimkan data hasil deteksi dan pengukuran volume ke Firebase secara tepat dan *real-time*. Pengujian ini bertujuan untuk mengetahui apakah data yang dikirim oleh Raspberry Pi dapat diterima, disimpan, dan ditampilkan dengan benar pada *database* atau *dashboard monitoring*. Data yang diuji meliputi kategori sampah yang terdeteksi, waktu deteksi, serta informasi volume tempat sampah.



Gambar 4.20 Data yang Terkirim ke Firebase

Pengujian dilakukan dengan menjalankan sistem seperti pada kondisi normal, kemudian mencocokkan data yang muncul di Firebase dengan hasil deteksi yang terjadi secara langsung.

Jika data yang dikirim sesuai dengan kondisi sebenarnya dan tidak mengalami kegagalan pengiriman, maka sistem dinyatakan berhasil dalam proses komunikasi data.

4.4 Pengujian Sistem

Pengujian sistem dilakukan untuk memastikan bahwa seluruh komponen yang telah dirancang dan diimplementasikan dapat bekerja dengan baik secara terintegrasi. Pada tahap ini, sistem diuji secara menyeluruh mulai dari proses pendeteksian objek menggunakan webcam, proses klasifikasi oleh model, pergerakan motor servo sebagai pemilah sampah, hingga pengiriman dan penampilan data. Pengujian sistem ini dilaksanakan di *Hall FTI UII*, yang merupakan lingkungan nyata dengan aktivitas pengguna yang beragam.



Gambar 4.21 Lokasi Penempatan Sistem di *Hall FTI UII*

Tujuan dari pengujian ini adalah untuk mengetahui tingkat keberhasilan sistem dalam menjalankan fungsinya secara otomatis dan *real-time*. Selain itu, pengujian juga bertujuan untuk mengidentifikasi kemungkinan kesalahan atau kendala yang muncul selama sistem beroperasi. Dengan dilakukan pengujian sistem secara menyeluruh di lingkungan nyata, dapat dipastikan bahwa alat pemilah sampah otomatis ini bekerja sesuai dengan tujuan penelitian dan siap digunakan dalam kondisi sebenarnya.



Gambar 4.22 Pengujian yang Berlangsung di *Hall FTI UII*

Selama pengujian berlangsung di Hall FTI UII, sistem telah diuji oleh sekitar 20–30 pengguna dengan berbagai jenis sampah yang termasuk dalam tiga kategori yang telah ditentukan, yaitu kertas dan tisu, botol plastik, serta kaleng. Berdasarkan hasil pengujian, sebagian besar percobaan menunjukkan bahwa sistem mampu mendeteksi, mengklasifikasikan, dan mengarahkan sampah ke tempat yang sesuai dengan kategori yang terdeteksi. Namun, terdapat sekitar 5 percobaan yang mengalami kegagalan deteksi. Kegagalan tersebut disebabkan oleh beberapa faktor, seperti kondisi objek yang kurang jelas, posisi objek yang tidak sesuai dengan area tangkap kamera, serta kondisi pencahayaan yang kurang optimal.

Untuk memperjelas hasil pengujian, skenario *Hardware Acceptance Testing* yang dilakukan ditunjukkan pada Tabel 4.5.

Tabel 4.5 Pengujian *Hardware Acceptance Testing*

No	Skenario Pengujian	Hasil yang Diharapkan	Hasil Aktual	Status
1	Kertas dan tisu dimasukkan	Sistem mendeteksi dan mengarahkan ke bak kertas	Sesuai	Berhasil
2	Botol plastik dimasukkan	Sistem mendeteksi dan mengarahkan ke bak plastik	Sesuai	Berhasil
3	Kaleng dimasukkan	Sistem mendeteksi dan mengarahkan ke bak kaleng	Sesuai	Berhasil
4	Volume bak bertambah	Persentase volume berubah pada LCD dan Firebase	Sesuai	Berhasil
5	Tidak ada objek terdeteksi	LCD menampilkan kondisi standby	Sesuai	Berhasil
6	Data hasil deteksi dikirim	Data tersimpan pada Firebase	Sesuai	Berhasil

Berdasarkan hasil *Hardware Acceptance Testing*, seluruh fungsi utama sistem dapat berjalan sesuai dengan kebutuhan yang telah ditentukan. Sistem mampu melakukan deteksi objek, menggerakkan motor servo sesuai kategori sampah, menampilkan informasi pada LCD, memantau volume tempat sampah, serta mengirimkan data ke Firebase secara real-time. Selain itu, tanggapan pengguna selama pengujian menunjukkan bahwa sistem mudah digunakan dan dapat membantu proses pemilahan sampah secara otomatis. Oleh karena itu, sistem dapat dinyatakan memenuhi kebutuhan fungsional dan siap digunakan dalam kondisi operasional nyata.

Evaluasi kinerja dan kelayakan sistem dilakukan dengan melibatkan 10 responden yang telah mencoba menggunakan sistem pemilah sampah otomatis secara langsung. Setelah melakukan pengujian, responden diminta memberikan penilaian terhadap sistem menggunakan kuesioner dengan skala Likert 1–5, dimana nilai 1 menunjukkan sangat tidak setuju dan nilai 5 menunjukkan sangat setuju. Hasil penilaian responden kemudian diolah untuk mengetahui tingkat keberhasilan sistem dari sisi fungsi, kinerja, keandalan, dan kelayakan penerapan. Data hasil penilaian responden disajikan pada Tabel 4.6.

Tabel 4.6 Data Responden

Responden	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11
R1	4	4	4	4	3	5	5	5	4	5	5
R2	3	3	5	4	4	5	4	4	5	5	5
R3	5	5	5	3	3	5	5	5	4	5	5
R4	4	5	5	4	2	4	4	4	5	4	5
R5	3	4	4	3	3	4	5	5	4	5	4
R6	5	5	5	5	3	4	4	5	5	5	4
R7	4	4	4	2	3	4	4	5	4	4	4
R8	5	5	5	3	3	5	5	4	5	5	5
R9	5	5	5	4	4	5	4	5	4	5	5
R10	4	4	4	3	3	4	4	5	4	5	4
Total Skor	42	44	46	35	31	45	44	47	44	48	46

Berdasarkan hasil evaluasi yang melibatkan 10 responden, diperoleh nilai *Functional Suitability* sebesar 83,5%, *Performance Efficiency* sebesar 76%, *Reliability* sebesar 90%, dan *Feasibility* sebesar 94%. Aspek dengan nilai tertinggi adalah *Feasibility*, yang menunjukkan bahwa responden menilai sistem layak diterapkan sebagai solusi pemilahan sampah di lingkungan kampus. Sementara itu, aspek *Performance Efficiency* memperoleh nilai paling rendah dibandingkan aspek lainnya, yaitu sebesar 76%, yang mengindikasikan bahwa masih terdapat beberapa responden yang menilai waktu respon sistem dapat ditingkatkan.

Secara keseluruhan, total persentase yang diperoleh adalah sebesar 85,82%, yang termasuk dalam kategori Sangat Baik. Hasil tersebut menunjukkan bahwa sistem pemilah sampah otomatis berbasis YOLOv5s dan IoT yang dikembangkan telah mampu menjalankan fungsi-fungsi utama dengan baik, bekerja secara stabil, serta layak untuk diterapkan dalam membantu proses pemilahan sampah secara otomatis.

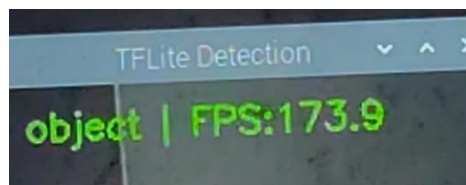
4.5 Pembahasan dan Evaluasi

Berdasarkan hasil pengujian yang telah dilakukan pada sistem, dapat diketahui bahwa sistem yang dikembangkan telah mampu bekerja dengan baik dari sisi model deteksi, implementasi perangkat keras, maupun interaksi dengan pengguna. Pembahasan ini dilakukan untuk mengevaluasi kinerja sistem secara keseluruhan serta mengidentifikasi kelebihan dan kekurangan yang masih terdapat pada sistem.

4.5.1 Evaluasi Model Deteksi YOLOv5s

Hasil proses training menunjukkan bahwa model YOLOv5s memiliki performa yang sangat baik dengan nilai mAP@0.50 sebesar 99% dan mAP@0.50–0.95 sebesar 88,5%. Nilai precision sebesar 0,986 dan recall sebesar 0,978 menunjukkan bahwa model mampu mendeteksi objek dengan tingkat ketepatan yang tinggi. Selain itu, hasil confusion matrix menunjukkan bahwa kesalahan klasifikasi antar kelas sangat kecil, sehingga model dapat membedakan jenis sampah dengan baik.

Pada masing-masing kelas, performa model juga menunjukkan hasil yang konsisten, di mana kelas botol plastik memiliki akurasi tertinggi, diikuti oleh kelas kaleng lalu kertas dan tisu. Hal ini menunjukkan bahwa model mampu mengenali objek dengan karakteristik yang berbeda secara akurat.



Gambar 4.23 FPS Model yang didapat

Selain akurasi, performa model juga dievaluasi dari sisi kecepatan deteksi. Berdasarkan hasil pengujian, model mampu mencapai kecepatan sebesar 173.9 *frames per second (FPS)* setelah dikonversi ke dalam format TensorFlow Lite (TFLite). Nilai ini menunjukkan bahwa model memiliki efisiensi yang sangat baik dalam proses inferensi dan mampu berjalan secara real-time pada perangkat dengan keterbatasan sumber daya seperti Raspberry Pi 3B+.

Namun demikian, pada implementasi sistem secara keseluruhan, waktu yang dibutuhkan untuk memproses satu objek sampah saat pendeteksian berada pada rentang 3 hingga 6 detik. Waktu ini dipengaruhi oleh beberapa faktor, seperti proses pengambilan citra dari kamera, pemrosesan menggunakan OpenCV, serta respon perangkat keras. Meskipun terdapat perbedaan antara kecepatan model dan waktu respon sistem, sistem tetap mampu bekerja secara otomatis dengan performa yang stabil dalam kondisi nyata.

Dengan demikian, kombinasi antara akurasi yang tinggi, kecepatan inferensi model yang cepat, serta kemampuan sistem dalam melakukan pemrosesan secara terintegrasi menunjukkan bahwa model YOLOv5s yang digunakan sudah optimal untuk diterapkan pada sistem pemilah sampah otomatis.

4.5.2 Evaluasi Implementasi Sistem dan Pengujian Unit

Dari sisi implementasi, sistem telah berhasil mengintegrasikan berbagai komponen perangkat keras dan perangkat lunak, meliputi webcam, Raspberry Pi 3B+, motor servo, sensor ultrasonik, LCD 16x2 I2C, serta Firebase. Seluruh komponen dapat bekerja secara terintegrasi untuk mendukung proses pemilahan sampah secara otomatis. Kamera digunakan untuk menangkap citra objek yang kemudian diproses oleh model YOLOv5s. Hasil deteksi digunakan sebagai dasar pengambilan keputusan untuk menggerakkan motor servo sehingga sampah dapat diarahkan ke kategori yang sesuai. Selain itu, sensor ultrasonik mampu memantau volume tempat sampah, sedangkan LCD dan Firebase berfungsi sebagai media penyampaian informasi hasil deteksi dan kondisi volume tempat sampah.

Berdasarkan hasil pengujian unit yang telah dilakukan, setiap komponen sistem menunjukkan kinerja yang sesuai dengan perancangan. Modul object detection mampu mengenali sebagian besar objek sampah yang telah dilatih pada model dengan tingkat keberhasilan yang cukup baik. Meskipun demikian, masih ditemukan beberapa kesalahan deteksi pada objek yang memiliki karakteristik visual berbeda dari dataset pelatihan atau memiliki kemiripan bentuk dengan kategori lain. LCD dapat menampilkan informasi hasil deteksi dan volume tempat sampah dengan jelas, sedangkan motor servo mampu bergerak sesuai perintah untuk melakukan proses pemilahan sampah. Sensor ultrasonik juga menunjukkan hasil pembacaan yang relatif stabil dalam mengukur volume tempat sampah, meskipun terdapat sedikit perbedaan hasil pengukuran pada beberapa kondisi tertentu. Selain itu, pengiriman data ke Firebase berjalan dengan baik sehingga informasi hasil deteksi dan volume tempat sampah dapat dipantau secara *real-time*.

4.5.3 Evaluasi Pengujian Sistem Secara Keseluruhan

Pengujian sistem secara menyeluruh yang dilakukan di lingkungan nyata, yaitu *Hall FTI* UII, menunjukkan bahwa sistem pemilah sampah otomatis mampu bekerja sesuai dengan fungsi yang telah dirancang. Sistem dapat melakukan proses deteksi objek, klasifikasi jenis sampah, pemilahan menggunakan motor servo, monitoring volume tempat sampah, serta pengiriman data ke Firebase secara terintegrasi. Selama pengujian berlangsung, sistem telah dicoba oleh sekitar 20 hingga 30 pengguna dengan berbagai jenis sampah yang termasuk dalam kategori kertas dan tisu, botol plastik, serta kaleng.

Berdasarkan hasil pengujian lapangan, sebagian besar percobaan menunjukkan bahwa sistem mampu mendeteksi dan mengklasifikasikan objek dengan baik. Namun, masih ditemukan beberapa kegagalan deteksi yang disebabkan oleh faktor eksternal, seperti pencahayaan yang kurang optimal, posisi objek yang tidak sesuai dengan area pengambilan gambar, serta karakteristik objek yang berbeda dari data pelatihan. Faktor-faktor tersebut mempengaruhi kemampuan model dalam mengenali objek secara akurat.

Selain pengujian lapangan, dilakukan pula evaluasi kinerja dan kelayakan sistem dengan melibatkan 10 responden. Hasil evaluasi menunjukkan bahwa aspek *Functional Suitability* memperoleh nilai 83,5%, *Performance Efficiency* sebesar 76%, *Reliability* sebesar 90%, dan *Feasibility* sebesar 94%. Secara keseluruhan, sistem memperoleh nilai sebesar 85,82% yang termasuk dalam kategori Sangat Baik. Hasil tersebut menunjukkan bahwa sistem dinilai mampu menjalankan fungsi-fungsi utama dengan baik, memiliki tingkat keandalan yang tinggi, serta layak diterapkan sebagai solusi pendukung pemilahan sampah otomatis di lingkungan kampus.

Secara keseluruhan, hasil pengujian dan evaluasi menunjukkan bahwa sistem telah berhasil memenuhi tujuan penelitian. Meskipun masih terdapat beberapa keterbatasan yang mempengaruhi performa deteksi pada kondisi tertentu, sistem telah mampu bekerja secara stabil dan memiliki potensi untuk dikembangkan lebih lanjut guna meningkatkan akurasi, kecepatan respon, serta ketahanan sistem terhadap berbagai kondisi lingkungan.

4.5.4 Kelebihan dan Keterbatasan Sistem

Berdasarkan hasil evaluasi yang telah dilakukan, sistem memiliki beberapa kelebihan dan keterbatasan sebagai berikut:

Kelebihan Sistem:

1. Mampu mendeteksi dan mengklasifikasikan jenis sampah dengan tingkat akurasi yang tinggi.
2. Dapat bekerja secara otomatis dan *real-time*.
3. Terintegrasi dengan baik antara perangkat keras dan perangkat lunak sehingga seluruh komponen dapat bekerja secara sinkron.

Keterbatasan Sistem:

1. Performa sistem masih dipengaruhi oleh kondisi pencahayaan lingkungan.
2. Sensor ultrasonik masih memiliki potensi kesalahan pengukuran pada kondisi tertentu.
3. Sistem masih terbatas pada beberapa jenis sampah sesuai dengan dataset yang digunakan.
4. Kecepatan sistem masih dapat ditingkatkan untuk memperoleh performa yang lebih optimal.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil perancangan, implementasi, dan pengujian yang telah dilakukan, sistem pemilah sampah otomatis berbasis *image detection* dan *Internet of Things* (IoT) berhasil dikembangkan sesuai dengan tujuan penelitian. Sistem memanfaatkan Raspberry Pi 3B+ sebagai pengendali utama yang mengintegrasikan berbagai komponen, seperti webcam, motor servo, sensor ultrasonik, LCD 16x2 I2C, dan Firebase sehingga mampu bekerja secara otomatis dalam proses pemilahan sampah.

Model deteksi objek yang digunakan adalah YOLOv5s yang dilatih menggunakan dataset citra sampah yang terdiri dari tiga kategori, yaitu kertas dan tisu, botol plastik, serta kaleng. Model kemudian dikonversi ke format TensorFlow Lite (TFLite) agar dapat dijalankan pada Raspberry Pi 3B+ yang memiliki keterbatasan sumber daya komputasi. Hasil pelatihan menunjukkan performa yang sangat baik dengan nilai mAP@0.50 sebesar 99%, sehingga model mampu mengenali objek sampah sesuai kategori yang telah ditentukan.

Hasil pengujian unit menunjukkan bahwa setiap komponen sistem dapat berfungsi sesuai dengan perancangan. Modul object detection mampu mengenali sebagian besar objek yang diuji, motor servo dapat melakukan proses pemilahan sesuai hasil klasifikasi, sensor ultrasonik mampu mengukur volume tempat sampah, serta LCD dan Firebase dapat menampilkan informasi hasil deteksi dan kondisi volume tempat sampah secara *real-time*.

Berdasarkan pengujian object detection menggunakan berbagai variasi objek sampah, sistem berhasil mendeteksi 14 dari 18 objek yang diuji dengan tingkat keberhasilan sebesar 77,78%. Beberapa kegagalan deteksi terjadi pada objek yang memiliki karakteristik visual berbeda dari dataset pelatihan atau memiliki kemiripan bentuk dengan kategori lain. Hal ini menunjukkan bahwa kualitas dan keberagaman dataset sangat mempengaruhi performa model dalam proses deteksi.

Hasil *Hardware Acceptance Testing* (HAT) menunjukkan bahwa seluruh fungsi utama sistem, mulai dari deteksi objek, proses pemilahan menggunakan motor servo, monitoring volume tempat sampah, hingga pengiriman data ke Firebase dapat berjalan dengan baik dan terintegrasi. Selain itu, hasil evaluasi kinerja dan kelayakan sistem yang melibatkan 10 responden memperoleh nilai keseluruhan sebesar 85,82% yang termasuk dalam kategori

Sangat Baik. Hasil tersebut menunjukkan bahwa sistem memiliki tingkat fungsionalitas, keandalan, dan kelayakan implementasi yang baik untuk diterapkan di lingkungan kampus.

Secara keseluruhan, sistem yang dikembangkan telah berhasil mencapai tujuan penelitian, yaitu mampu melakukan pemilahan sampah secara otomatis berdasarkan jenis sampah yang terdeteksi serta menyediakan informasi volume tempat sampah secara *real-time* melalui teknologi IoT. Sistem ini memiliki potensi untuk diterapkan sebagai solusi pendukung pengelolaan sampah yang lebih efektif dan efisien di lingkungan kampus maupun lingkungan publik lainnya.

5.2 Saran

Adapun beberapa saran untuk pengembangan penelitian selanjutnya adalah sebagai berikut:

1. Menambahkan jumlah dan variasi dataset agar model dapat mengenali lebih banyak jenis sampah dengan akurasi yang lebih tinggi.
2. Mengembangkan sistem agar dapat mendeteksi lebih banyak kategori sampah, tidak hanya tiga jenis.
3. Mengoptimalkan performa sistem pada Raspberry Pi agar proses deteksi lebih cepat dan stabil.
4. Mengembangkan desain alat agar lebih kuat, rapi, dan siap digunakan dalam skala yang lebih luas.

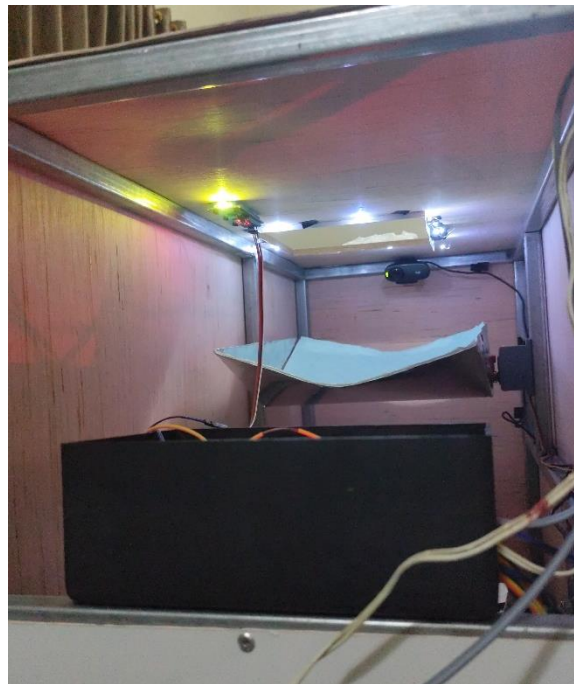
DAFTAR PUSTAKA

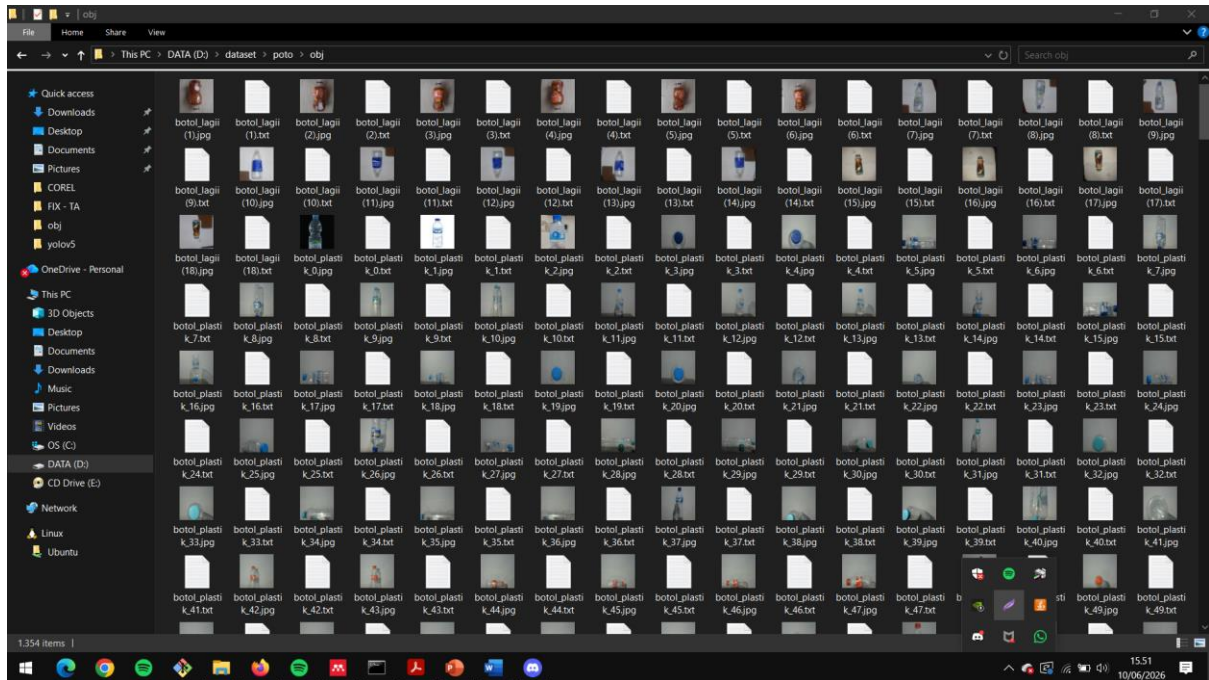
- Adlim, D. A., Ramadhan, I., Irawan, B., & Wahyu, S. (2025). Implementasi Object Detection Klasifikasi Sampah Organik dan Anorganik Menggunakan Raspberry Pi dengan Algoritma YOLO. *Kohesi: Jurnal Multidisiplin Saintek*, 10(3). <https://doi.org/10.8734/Kohesi.v1i2.365>
- Al-Rafi, M. F., Regasari, R., Putri, M., & Setiawan, E. (2025). Rancang Bangun Prototipe Sistem Pemilahan E-Waste Menggunakan MobileNetV3 Berbasis Raspberry Pi 4. *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer*, 9(7). <http://j-ptiik.ub.ac.id>
- Anggara, J., Ryansyah, E., & Dermawan, B. A. (2025). Implementasi Object Detection dalam Klasifikasi Sampah untuk Meningkatkan Efisiensi Pengelolaan Limbah. *JATI (Jurnal Mahasiswa Teknik Informatika)*, 9(3), 4923–4930. <https://doi.org/10.36040/jati.v9i3.13813>
- Dewi, N. M. N. B. S. (2021). Analisa Limbah Rumah Tangga Terhadap Dampak Pencemaran Lingkungan. *Jurnal Ganec Swara*, 15(2), 1159–1164. <http://journal.unmasmataram.ac.id/index.php/GARA>
- Fajar, S. F., Regasari, R. R. M. P., & Setyawan, G. E. (2025). Rancang Bangun Pemilahan Sampah Plastik Otomatis Menggunakan YOLO Pada Raspberry Pi. *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer*, 9(11). <http://j-ptiik.ub.ac.id>
- Fauzi, Y., Andiono, E., & Khamali, M. (2020). Aplikasi Object Detection and Tracking Untuk Penyandang Tunanetra dengan Internet of Things (IoT) (Menggunakan Bahasa Pemrograman Phyton). *Jurnal Gerbang STMIK Bani Saleh*, 1(10).
- Fernando, L. L., & Utaminigrum, F. (2024). Rancang Bangun Sistem Klasifikasi Sampah Menggunakan YOLOv8 Berbasis Raspberry Pi 4. *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer*, 8(6). <http://j-ptiik.ub.ac.id>
- Fernando, M., Jasa, L., & Hartati, R. S. (2022). Monitoring System Kecepatan dan Arah Angin Berbasis Internet of Things (IoT) Menggunakan Raspberry Pi 3. *Majalah Ilmiah Teknologi Elektro*, 21(1), 135–142. <https://doi.org/10.24843/MITE.2022.v21i01.P18>
- Hasibuan, M. R. R. (2023). *Manfaat Daur Ulang Sampah Organik dan Anorganik untuk Kesehatan Lingkungan*. <https://doi.org/10.31219/osf.io/yb42t>
- Hendri, F. R., & Utaminigrum, F. (2022). Rancang Bangun Sistem Pengklasifikasi Jenis Sampah Organik dan Anorganik menggunakan metode You Only Look Once versi 3

- berbasis Raspberry Pi. *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer*, 6(7), 3509–3514. <http://j-ptiik.ub.ac.id>
- Khairnar, P., Bansode, S., Sahota, B., & Chourasia, P. (2021). Automation of Home (IoT) using Raspberry Pi. *International Journal of Scientific Research in Science, Engineering and Technology*, 235–239. <https://doi.org/10.32628/ijsrset218335>
- Khoiriyah, H. (2021). Analisis Kesadaran Masyarakat Akan Kesehatan terhadap Upaya Pengelolaan Sampah di Desa Tegorejo Kecamatan Pegandon Kabupaten Kendal. *Indonesian Journal of Conservation*, 10, 13–20. <https://doi.org/10.15294/ijc.v10i1.30587>
- Kristanti, N., Samsugi, S., Surahman, A., Pratama, R. F., & Adam, R. I. (2022). Penerapan Sensor Ultrasonik pada Kotak Sampah Otomatis Menggunakan Telegram dan Alarm Suara. *Jurnal Teknik Dan Sistem Komputer (JTIKOM)*, 3(2), 67–78. <https://doi.org/10.33365/jtikom.v3i2.2347>
- Mahendra, D. A., & Winardi, S. (2023). Perancangan Realtime Database Firebase untuk IoT dan Unity Menggunakan Metode SDLC. *Jurnal Ilmu Komputer Dan Bisnis*, 14(2a), 72–82. <https://doi.org/10.47927/jikb.v14i2a.525>
- Nurfauziah, H., & Jamaliyah, I. (2022). Perbandingan Metode Testing Antara BlackBox dengan WhiteBox pada Sebuah Sistem Informasi. *Jurnal VISUALIKA*, 8(2), 105–113.
- Papakyriakou, D., & Barbounakis, I. S. (2023). Benchmarking and Review of Raspberry Pi (RPi) 2B vs RPi 3B vs RPi 3B+ vs RPi 4B (8GB). *International Journal of Computer Applications*, 185(3), 37–52. <https://doi.org/10.5120/ijca2023922693>
- Rahman, M. (2021). *Faktor Penyebab dan Dampak serta Kebijakannya Terhadap Permasalahan Pencemaran Sampah*. <https://doi.org/10.31219/osf.io/x6dve>
- Ramadhan, J. A., Haniva, D. T., & Suharso, A. (2023). Systematic Literature Review Penggunaan Metodologi Pengembangan Sistem Informasi Waterfall, Agile, dan Hybrid. *Journal Information Engineering and Educational Technology*, 7(1), 36–42. <https://doi.org/10.26740/jieet.v7n1.p36-42>
- Rizqolima, A. H., & Widhiantoro, D. (2025). Analisis Perbandingan Algoritma Deepface, YOLO, dan Tensorflow dalam Pengenalan Wajah. *Seminar Nasional Inovasi Vokasi*, 4, 768–777.
- Santoso, U. R. N., & Gamar, F. (2025). Deteksi Sampah Botol Plastik di Perairan Menggunakan YOLO v4-Tiny. *Jurnal Teknologi Dan Sistem Informasi Bisnis*, 7(1), 91–98. <https://doi.org/10.47233/jteksis.v7i1.1744>

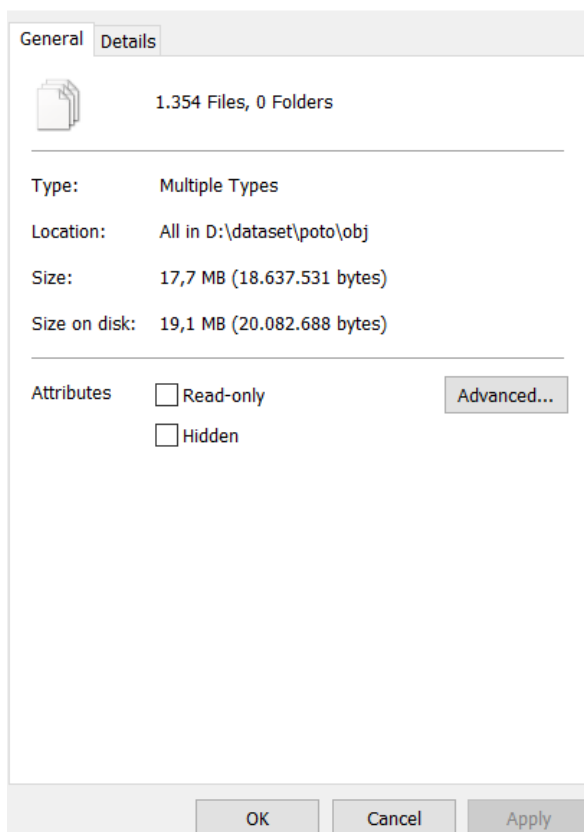
- Santoso, U. R. N., Gamar, F., & Darmawan, A. (2025). Implementasi Transformasi Homografi dan YOLO v4-Tiny untuk Deteksi Botol dan Kaleng. *Jurnal Teknologi Dan Sistem Informasi Bisnis*, 7(3), 442–449. <https://doi.org/10.47233/jteksis.v7i3.1981>
- Sarifa, M., Syukri, P. N., & Siregar, N. D. (2025). Analisis Faktor Penyebab Masyarakat Membuang Sampah Sembarangan Dan Solusi Melalui Pendidikan Pancasila. *JIIIC: Jurnal Intelek Insan Cendikia*, 2(4), 6422–6426. <https://jicnusantara.com/index.php/jiic>
- Selay, A., Andgha, G. D., Alfarizi, M. A., Wahyudi M Izdhihar Bintang, Falah, M. N., Khaira, M., & Encep, M. (2022). Internet Of Things. *Karimah Tauhid*, 1(6), 860–868. <https://doi.org/10.30997/karimahtauhid.v1i6.7633>
- Sholihah, K. K. A. (2020). Kajian tentang Pengelolaan Sampah di Indonesia. *Swara Bhumi*, 3(3).
- Slamat, J. U. (2021). *Perancangan Alat Perajang Sampah Organik Berbasis Mikrokontroler*.
- Sommerville, Ian. (2016). *Software engineering*. Pearson.
- Suciningtyas, I. K. L. N., Risandriya, S. K., Ismail, A., & Maulana, R. D. (2024). Rancangan Sistem Kontrol dan Monitoring pada Wire Selection Boxes. *Journal of Applied Electrical Engineering*, 8(2), 140–145. <https://doi.org/10.30871/jaee.v8i2.8797>
- Swasotomo, S. D. (2025). *Metode Waterfall dalam Pengembangan Sistem Informasi*.
- Utami, A. P., Pane, N. N., & Hasibuan, A. (2023). Analisis Dampak Limbah Sampah/Sampah Rumah Tangga terhadap Pencemaran Lingkungan Hidup. *Cross Border*, 6(2), 1107–1112.
- Wibysono, A. Y., Susilawati, H., & Matin, I. M. M. (2022). Rancang Bangun Alat Pemilah Sampah Organik dan Non Organik Berbasis Raspberry Pi. *Jurnal FUSE – Teknik Elektro*, 2(2), 88–96. <https://doi.org/10.52434/jft.v2i2.2338>
- Zulfachmi, Z., Zahara, A., & Hardinata, D. (2024). Klasifikasi Stingless Bee Menggunakan Metode Image Classification Berbasis OpenCV. *Jurnal Bangkit Indonesia*, 13(2), 7–12. <https://doi.org/10.52771/bangkitindonesia.v13i2.321>

LAMPIRAN





botol_plastik_4.jpg, ... Properties

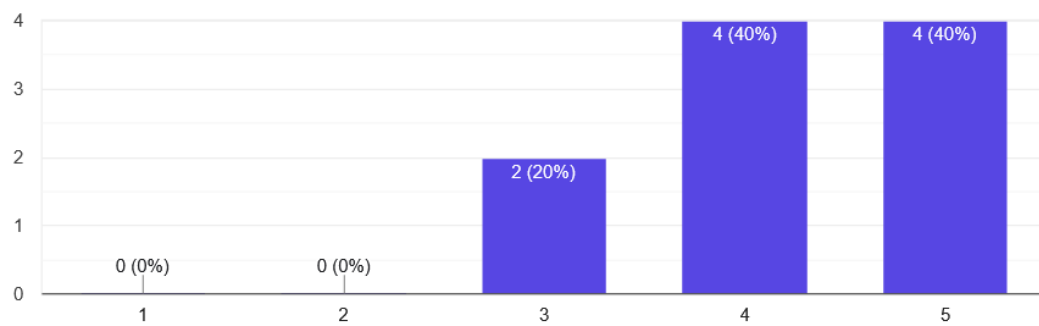




[Copy chart](#)

Sistem mampu mendeteksi jenis sampah dengan benar.

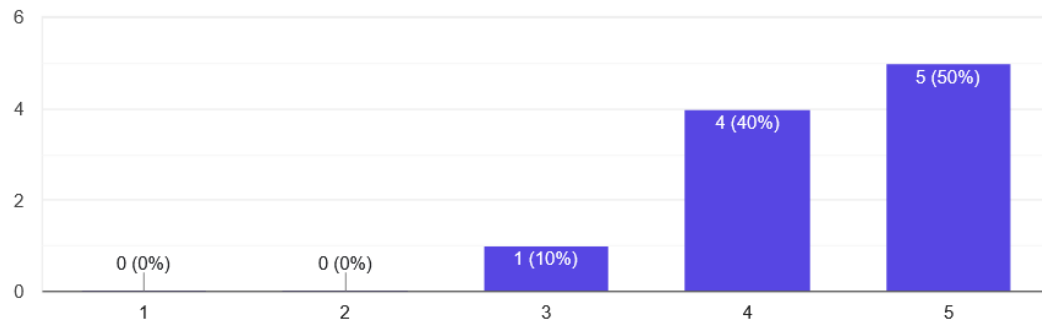
10 responses



[Copy chart](#)

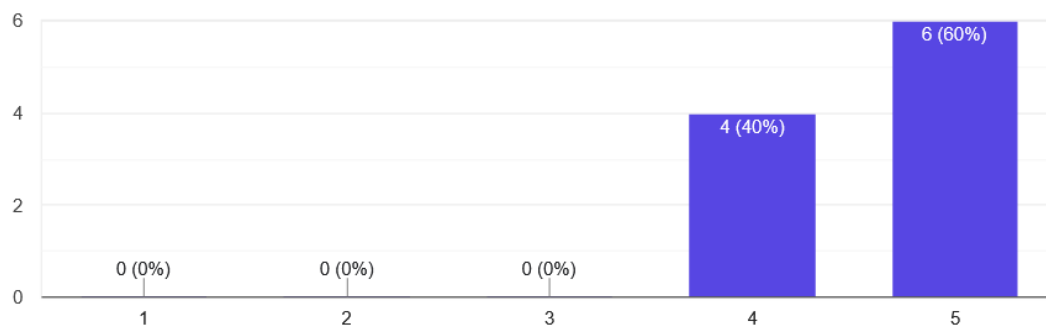
Sistem mampu mengklasifikasikan sampah sesuai kategori yang ditentukan.

10 responses

[Copy chart](#)

Sistem mampu mengarahkan sampah ke tempat yang sesuai melalui motor servo.

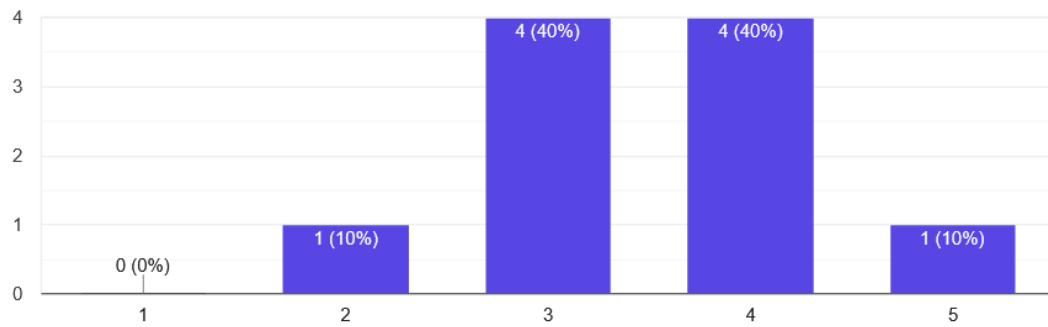
10 responses



 Copy chart

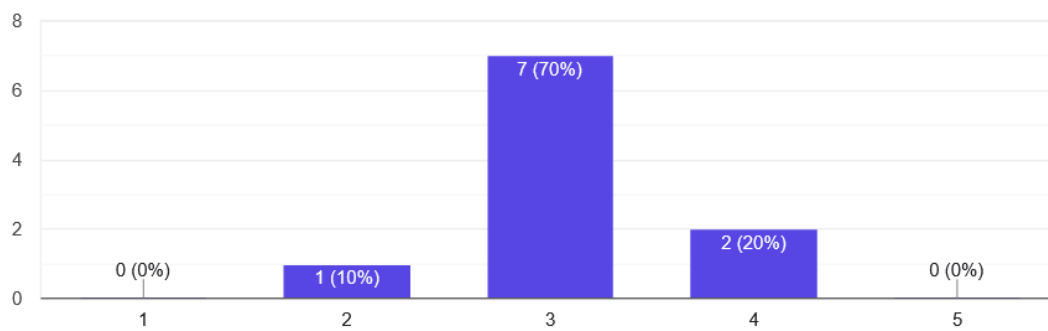
Sistem mampu menampilkan informasi volume tempat sampah dengan benar.

10 responses

 Copy chart

Sistem memberikan respon deteksi dalam waktu yang cepat.

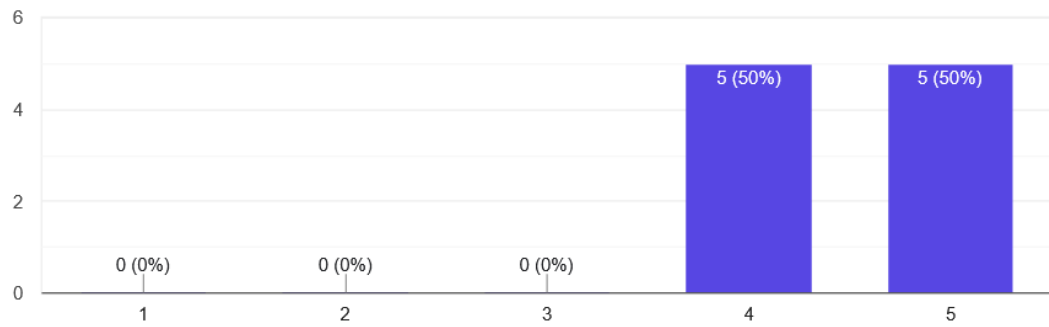
10 responses



 Copy chart

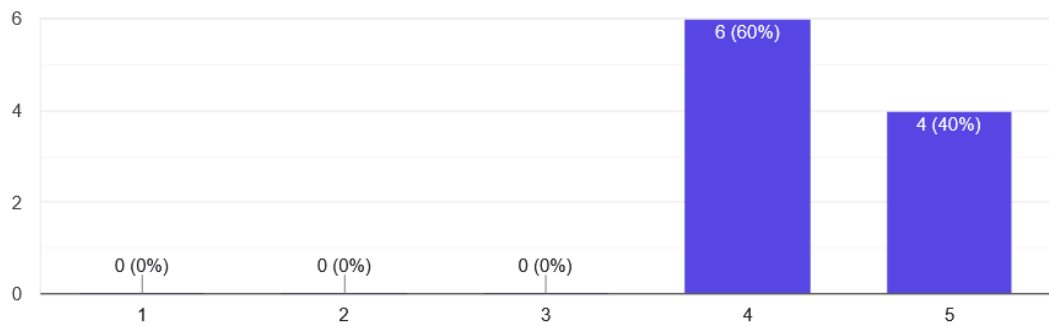
Pergerakan motor servo berlangsung dengan cepat setelah objek terdeteksi.

10 responses

 Copy chart

Sistem dapat beroperasi dengan stabil selama proses pengujian.

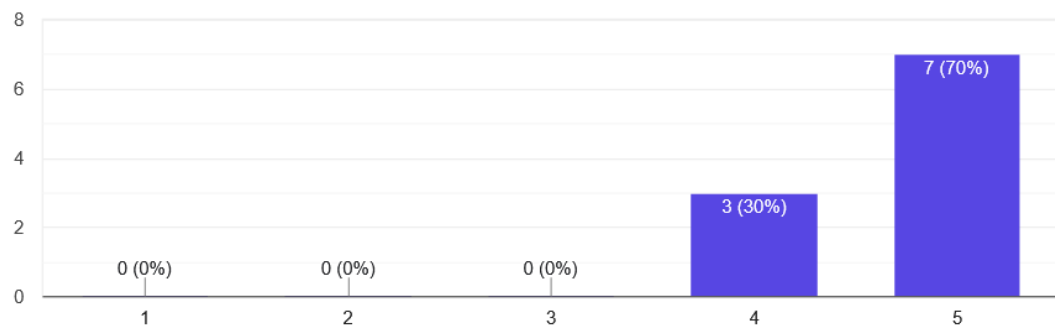
10 responses



[Copy chart](#)

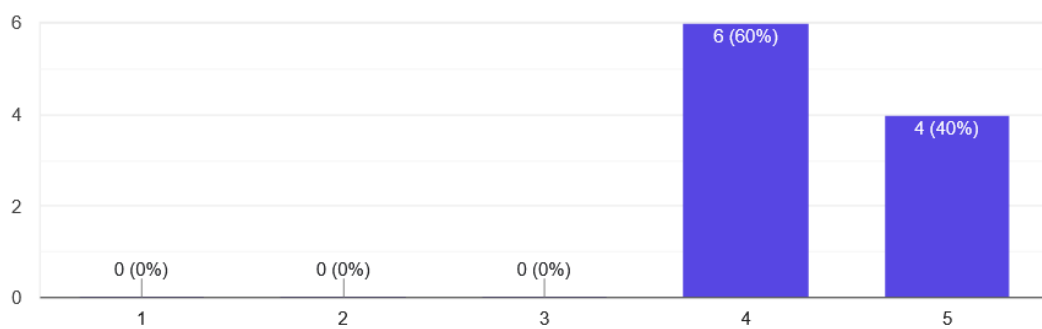
Hasil deteksi yang diberikan sistem konsisten pada objek yang sama.

10 responses

[Copy chart](#)

Sistem dapat bekerja tanpa mengalami gangguan atau kegagalan fungsi yang signifikan.

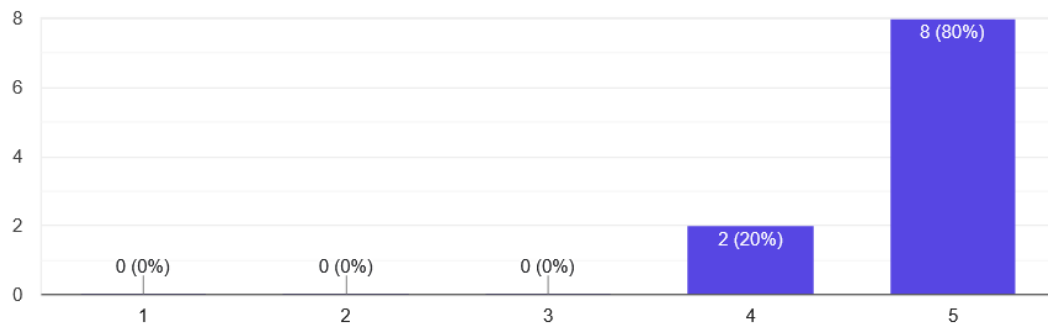
10 responses



 Copy chart

Sistem layak diterapkan di lingkungan kampus.

10 responses

 Copy chart

Sistem dapat membantu meningkatkan proses pemilahan sampah.

10 responses

