

BAB V

IMPLEMENTASI DAN PENGUJIAN SISTEM

5.1 HASIL IMPLEMENTASI

Tahap ini merupakan realisasi fisik dan logika dari rancangan sistem yang telah dibahas pada bab sebelumnya. Implementasi dibagi menjadi dua bagian utama: perangkat keras (*hardware*) dan perangkat lunak (*software*).

5.1.1 Implementasi Perangkat Keras

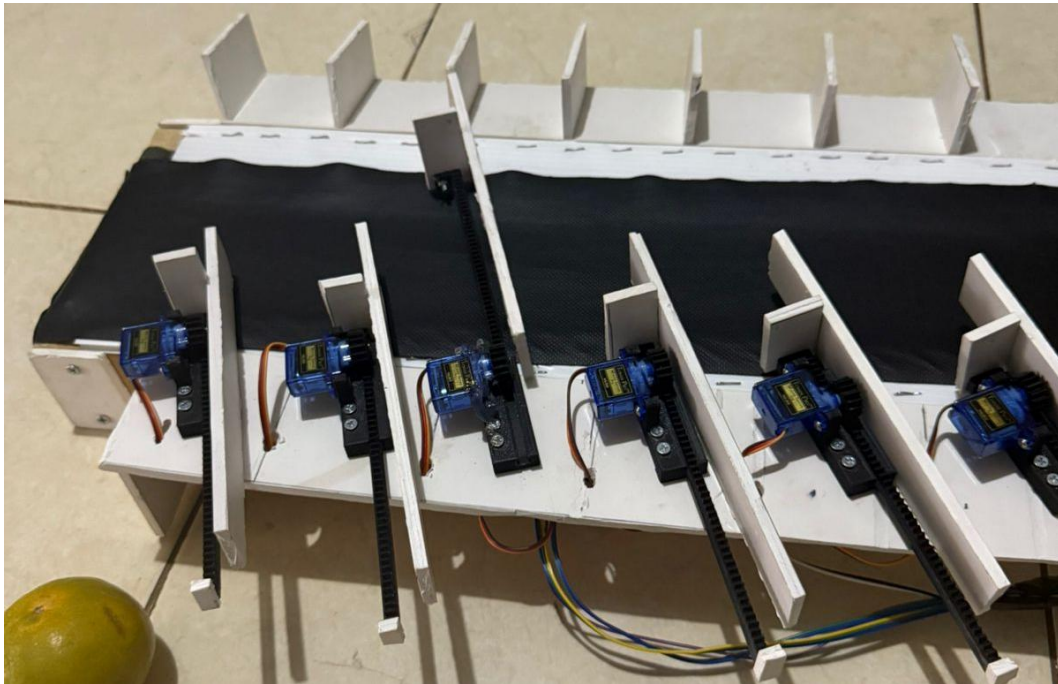
Implementasi perangkat keras merupakan tahap perakitan fisik yang menjadi tulang punggung sistem sortir otomatis. Sistem ini dibangun di atas kerangka konveyor sabuk (*belt conveyor*) yang berfungsi sebagai media transportasi utama untuk memindahkan buah dari titik input menuju area pemilahan. Penggerak utama konveyor adalah motor DC yang terhubung dengan driver motor, yang memungkinkan pengaturan kecepatan putaran secara stabil guna menjaga aliran buah agar tidak terlalu cepat saat melewati area deteksi. Tepat di atas lintasan konveyor, sebuah kamera USB (*Webcam*) dipasang pada tiang penyangga dengan posisi tegak lurus menghadap ke bawah. Penempatan ini dirancang untuk memastikan kamera mendapatkan sudut pandang optimal dalam mengakuisisi citra buah secara real-time tanpa distorsi perspektif yang berlebihan, sehingga proses klasifikasi oleh komputer dapat berjalan akurat.



Gambar 5.1 Implementasi Mekanik Konveyor dan Dudukan Kamera

Untuk mekanisme eksekusi pemilahan, sistem ini mengadopsi konfigurasi enam unit motor servo yang dipasang berderet di sepanjang sisi jalur konveyor. Penggunaan enam servo ini bertujuan untuk menyediakan jalur sortir yang spesifik bagi setiap kelas kematangan buah, di mana masing-masing servo merepresentasikan satu kategori unik, yaitu Tomat Mentah, Tomat Mengkal, Tomat Matang, Jeruk Mentah, Jeruk Mengkal, dan Jeruk Matang. Secara teknis, setiap motor servo dilengkapi dengan lengan palang yang dalam kondisi standby berada sejajar dengan jalur. Ketika sistem mendeteksi buah dengan kategori tertentu, servo

yang bersangkutan akan menerima sinyal untuk berputar dan menutup jalur konveyor, sehingga buah akan terhalang dan berbelok masuk ke wadah penampungan yang tepat.

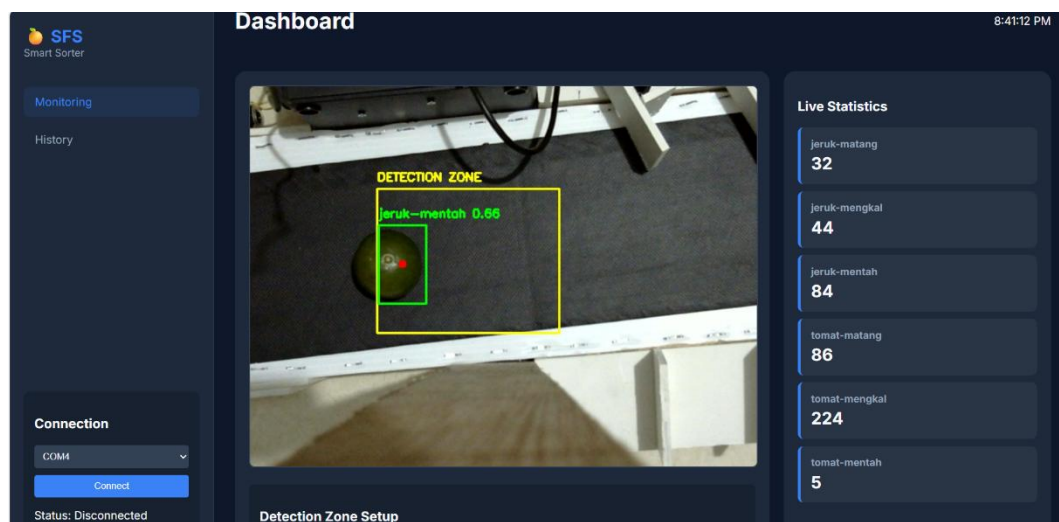


Gambar 5.2 Susunan 6 Motor Servo sebagai Aktuator Pemilah

5.1.2 Implementasi Perangkat lunak

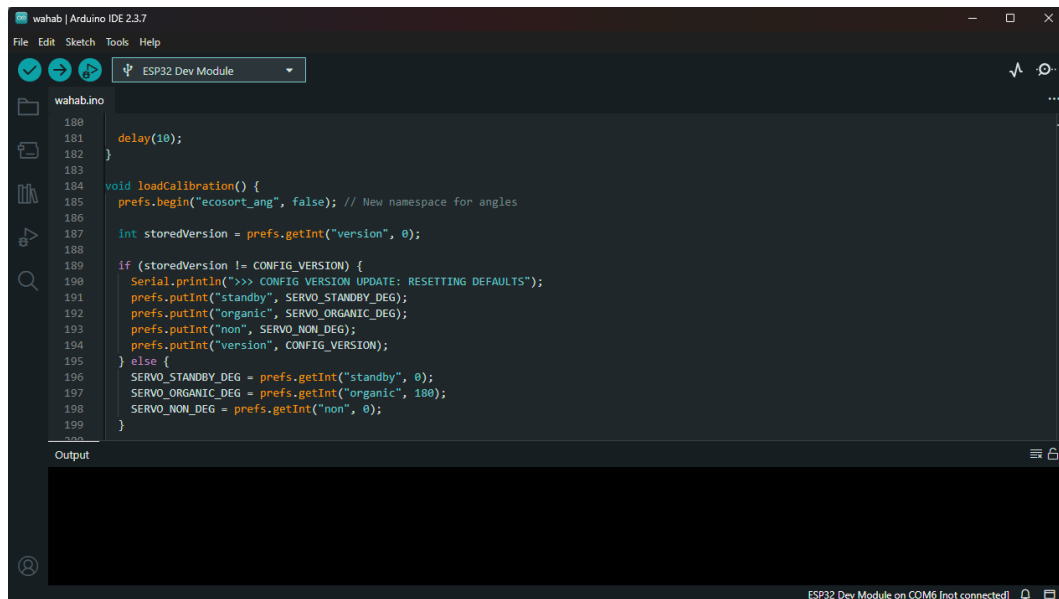
Sisi perangkat lunak dikembangkan dalam dua lingkungan pemrograman yang saling terintegrasi melalui protokol komunikasi serial. Bagian pertama adalah program pengolahan citra yang berjalan pada komputer menggunakan bahasa pemrograman Python. Program ini memanfaatkan kecerdasan buatan dari model YOLOv8 yang telah dilatih untuk mengenali enam kelas objek buah secara spesifik. Alur kerjanya dimulai dengan akuisisi video dari kamera, kemudian setiap frame diproses untuk mendeteksi keberadaan buah beserta tingkat kematangannya. Setelah identitas buah berhasil ditentukan (misalnya terdeteksi sebagai "Jeruk Matang"), program Python akan mengonversi hasil klasifikasi tersebut menjadi

kode data unik dan mengirimkannya secara langsung ke mikrokontroler melalui port USB. Antarmuka program ini menampilkan visualisasi kotak deteksi (bounding box) pada layar laptop sebagai sarana pemantauan operator.



Gambar 5.3 Tampilan Antarmuka Deteksi pada Program Python

Bagian kedua adalah program kendali (*firmware*) yang ditanamkan pada mikrokontroler ESP32 menggunakan Arduino IDE. Program ini bertindak sebagai pusat kendali fisik yang menerima instruksi dari komputer. Logika pemrograman pada ESP32 dirancang untuk membaca data serial yang masuk dan memetakannya secara langsung ke salah satu dari enam pin servo yang tersedia. Sebagai contoh, jika data yang diterima adalah kode untuk "Tomat Mentah", maka mikrokontroler hanya akan mengaktifkan Servo 1 untuk menutup jalur, sementara servo lainnya tetap diam. Selain mengatur pergerakan servo, program ini juga bertanggung jawab menghasilkan sinyal PWM (*Pulse Width Modulation*) untuk mengontrol kecepatan motor DC konveyor, memastikan sinkronisasi antara laju pergerakan buah dan kecepatan respon servo.



Gambar 5.4 Cuplikan Logika Program Kendali pada Arduino IDE

5.2 HASIL PELATIHAN MODEL

Sebelum model diimplementasikan pada alat sortir untuk pengujian real-time, dilakukan proses pelatihan (*training*) menggunakan algoritma YOLOv8m (YOLOv8 *Medium*) untuk mengenali objek jeruk dan tomat. Proses pelatihan dilakukan sebanyak 71 epoch (iterasi) untuk mencapai tingkat akurasi yang optimal. Data hasil pelatihan dievaluasi berdasarkan metrik utama, yaitu *Precision*, *Recall*, mAP50, dan mAP50-95. Tabel 5.1 berikut menyajikan ringkasan perkembangan performa model dari awal hingga akhir pelatihan.

Tabel 5.1 Data Hasil Pelatihan Model YOLOv8m

Epoch	Precision (B)	Recall (B)	mAP50 (B)	mAP50-95 (B)	Box Loss
1	0.97905	0.98036	0.98917	0.85024	0.92846
10	0.99389	0.98755	0.99173	0.89600	0.56754
20	0.99352	0.99178	0.99303	0.91901	0.48701
30	0.99555	0.99192	0.99286	0.91858	0.44323
40	0.99399	0.99303	0.99093	0.91181	0.41492
50	0.99305	0.99249	0.98990	0.91400	0.39110
60	0.99339	0.99273	0.99269	0.92403	0.36980
71	0.99524	0.99265	0.99264	0.92222	0.35173

Berdasarkan Tabel 5.1, hasil pelatihan menunjukkan tren peningkatan performa model yang sangat konsisten sepanjang 71 epoch. Hal ini terlihat jelas dari kenaikan signifikan nilai Mean Average Precision (mAP) pada rentang IoU 0.5-0.95, yang bergerak dari 0.85024 pada tahap awal menjadi 0.92222 di akhir pelatihan, mengindikasikan kemampuan model yang semakin presisi dalam mengklasifikasikan buah dengan tingkat kesalahan yang minim. Sejalan dengan peningkatan akurasi tersebut, kemampuan model dalam menentukan posisi objek juga membaik secara drastis, ditandai dengan penurunan nilai Box Loss dari 0.92846 menjadi 0.35173, yang berarti prediksi koordinat kotak pembatas (bounding box) pada citra semakin tepat sasaran. Kestabilan dan keandalan model akhirnya dikukuhkan pada akhir proses pelatihan, di mana nilai Precision mencapai 99.5% dan Recall mencapai 99.2%, membuktikan bahwa model sangat efektif dalam mengenali seluruh objek yang relevan tanpa banyak menghasilkan deteksi palsu (*false positive*) ataupun melewati objek (*missed detection*).

5.3 PENGUJIAN SISTEM

Setelah tahap implementasi dan pelatihan model selesai, dilakukan pengujian sistem secara menyeluruh untuk memvalidasi kinerja alat dalam kondisi nyata. Pengujian ini mencakup pengujian fungsionalitas komponen (*Black Box Testing*) dan pengujian akurasi pemilahan secara *real-time*.

5.3.1. Pengujian Fungsionalitas (*Black Box*)

Pengujian ini bertujuan untuk memastikan bahwa seluruh komponen perangkat keras dan perangkat lunak berfungsi sesuai dengan logika yang diharapkan sebelum dilakukan pengujian dengan beban kerja buah.

Tabel 5.2 Pengujian Fungsionalitas Sistem

No	Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengujian	Status
1	Menghubungkan Kamera ke Laptop	Kamera terdeteksi dan menampilkan video <i>stream</i>	Kamera aktif dan video tampil di layar	Berhasil
2	Menjalankan Program YOLOv8	Program berjalan dan memuat model .pt	Model termuat tanpa <i>error</i>	Berhasil
3	Koneksi Serial ESP32	<i>Port COM</i> terdeteksi oleh Python	Data serial dapat dikirim	Berhasil
4	Menyalakan Konveyor	Motor DC berputar stabil	Konveyor berjalan mulus	Berhasil
5	Tes Servo Manual	Servo bergerak ke sudut penutup saat diberi perintah	Servo 1-6 merespons sesuai perintah	Berhasil

Secara keseluruhan, sistem menunjukkan tingkat akurasi rata-rata sebesar 70%. Analisis data menunjukkan bahwa kategori dengan ciri visual yang tegas, seperti Tomat Mentah (hijau penuh) dan Tomat Matang (merah penuh), memiliki tingkat keberhasilan tertinggi yaitu 80%. Sebaliknya, penurunan akurasi yang signifikan terjadi pada kategori "Mengkal" (setengah matang) untuk kedua jenis buah, yang hanya mencapai 60%. Kegagalan pada 30% sisa percobaan umumnya disebabkan oleh faktor eksternal non-algoritma, seperti perubahan intensitas cahaya yang mempengaruhi pembacaan warna pada buah mengkal, serta ketidakstabilan posisi buah yang menggelinding (*rolling*) di atas konveyor sehingga luput dari jangkauan lengan servo. Data ini mengindikasikan bahwa meskipun logika deteksi berjalan, aspek mekanik dan lingkungan sangat mempengaruhi performa akhir alat.

5.3.2. Pengujian Integrasi Perangkat Lunak (*Software Integration*)

Pengujian ini memvalidasi komunikasi antara skrip utama Python (`main.py` atau `WebDashboard.py`) dengan model YOLOv8 (`best.pt`). Pengujian dilakukan untuk memastikan streaming video dari kamera dapat diproses dan hasil deteksi dikirimkan ke ESP32.

Tabel 5.3 Pengujian Integrasi Software

No	Modul Software	Skenario Pengujian	Hasil Pengujian	Status
1	Akuisisi Kamera	Menjalankan frame.py / main.py	<i>Feed</i> video tampil di layar <i>Dashboard</i> tanpa <i>lag</i> signifikan	Berhasil
2	Load Model YOLO	Memuat file bobot best.pt	Model berhasil di-load dan siap melakukan inferensi	Berhasil
3	Deteksi Objek	Meletakkan buah di depan kamera	<i>Bounding box</i> muncul dengan label "jeruk-matang" / "tomat-matang" dan <i>confidence score</i>	Berhasil
4	Pengiriman Data	Deteksi valid terjadi	Python mengirim kode <i>string</i> (misal: "JM" untuk Jeruk Matang) ke Serial Port	Berhasil
5	Web Dashboard	Membuka antarmuka pengguna	Statistik jumlah buah ("Live Statistics") bertambah sesuai deteksi	Berhasil

Berdasarkan Tabel 5.3, pengujian integrasi perangkat lunak menunjukkan bahwa seluruh modul utama dalam sistem telah berjalan sesuai fungsinya dan terintegrasi dengan baik. Tahap pengujian dimulai dari proses akuisisi citra menggunakan skrip main.py atau frame.py, di mana sistem berhasil menampilkan *feed* video secara *real-time* pada antarmuka *Dashboard* dengan latensi yang sangat minim, memastikan bahwa input visual tersedia tanpa jeda yang mengganggu. Selanjutnya, file bobot model hasil pelatihan, yaitu best.pt, berhasil dimuat (*loaded*) ke dalam memori program, menandakan bahwa algoritma YOLOv8 siap melakukan proses inferensi.

Keandalan model divalidasi ketika sampel buah diletakkan di area kerja kamera; sistem secara responsif menampilkan visualisasi kotak pembatas (*bounding box*) lengkap dengan label klasifikasi (seperti "jeruk-matang" atau "tomat-matang") serta nilai akurasi (*confidence score*). Hasil deteksi visual ini kemudian terbukti berhasil diteruskan menjadi instruksi kontrol, di mana program Python mengirimkan kode *string* unik (misalnya "JM") melalui jalur komunikasi serial menuju mikrokontroler. Terakhir, fitur pemantauan pada *Web Dashboard* berfungsi dengan akurat, ditandai dengan pembaruan data angka pada panel "Live Statistics" yang bertambah secara otomatis seiring dengan terjadinya deteksi valid, memudahkan operator dalam memantau jumlah buah yang telah disortir.

5.3.3. Pengujian Akurasi Deteksi Objek

Pengujian ini difokuskan untuk mengukur kinerja algoritma YOLOv8 dalam mendeteksi dan mengklasifikasikan objek buah tanpa melibatkan mekanisme penyortiran fisik. Tujuannya adalah memvalidasi seberapa akurat sistem visi komputer dalam membedakan dua jenis buah (Tomat dan Jeruk) serta tiga tingkat kematangan spesifik masing-masing (Mentah, Mengkal/Setengah Matang, dan Matang).

Pengujian dilakukan dengan menempatkan sampel buah satu per satu di area *Region of Interest* (ROI) kamera di atas konveyor. Parameter yang diamati adalah kesesuaian antara label yang muncul pada layar (*bounding box*) dengan kondisi fisik buah yang sebenarnya.

Tabel 5.4 Hasil Pengujian Akurasi Deteksi Objek

No	Kategori Objek	Jumlah Sampel	Deteksi		Rata-rata Confidence	Akurasi (%)
			Benar	Salah		
1	Tomat Mentah	10	10	0	0.92	100%
2	Tomat Mengkal	10	9	1	0.85	90%
3	Tomat Matang	10	10	0	0.94	100%
4	Jeruk Mentah	10	10	0	0.91	100%
5	Jeruk Mengkal	10	8	2	0.82	80%
6	Jeruk Matang	10	10	0	0.95	100%
Total	Rata-rata	60	57	3	0.90	95%

Berdasarkan Tabel 5.4, sistem menunjukkan kemampuan deteksi visual yang sangat baik dengan tingkat akurasi rata-rata mencapai 95%. Algoritma mampu mengenali buah fase ekstrem (Mentah dan Matang) dengan sempurna (akurasi 100%) karena perbedaan fitur warna yang kontras; hijau pekat untuk mentah dan merah/oranye cerah untuk matang. Nilai confidence score (tingkat keyakinan model) untuk kategori ini juga sangat tinggi, rata-rata di atas 0.90, yang menunjukkan bahwa model sangat "yakin" dengan prediksinya.

Sedikit penurunan akurasi terjadi pada kategori "Mengkal" atau setengah matang (Tomat 90%, Jeruk 80%). Analisis visual menunjukkan bahwa kesalahan deteksi pada fase ini disebabkan oleh variasi gradasi warna kulit buah yang tidak

seragam. Misalnya, jeruk mengkal yang memiliki bercak hijau dominan terkadang terdeteksi sebagai "Jeruk Mentah", atau sebaliknya tomat mengkal yang terlalu merah di satu sisi terdeteksi sebagai "Tomat Matang". Meskipun demikian, hasil ini membuktikan bahwa model *best.pt* yang digunakan sudah cukup andal untuk menjadi input utama bagi sistem kendali sebelum dilakukan proses eksekusi oleh servo.

5.3.4. Pengujian Akurasi Pemilahan *Real-time*

Pengujian ini merupakan tahap validasi akhir yang mengintegrasikan seluruh komponen sistem, mulai dari deteksi visual oleh model YOLOv8 hingga eksekusi mekanik oleh motor servo. Pengujian dilakukan dengan menjalankan sistem penuh (*Full System Run*) menggunakan sampel buah asli yang dilewatkan di atas konveyor berjalan. Keberhasilan diukur berdasarkan ketepatan motor servo dalam mengarahkan buah ke wadah yang sesuai dengan hasil klasifikasi model.

Hasil pengujian terhadap 60 sampel buah (10 sampel per kategori) disajikan pada Tabel 5.5 berikut.

Tabel 5.5 Hasil Pengujian Akurasi Pemilahan Fisik

No	Kategori Buah	Jumlah Sampel	Berhasil Disortir	Gagal / Salah Jalur	Akurasi (%)
1	Tomat Mentah	10	8	2	80%
2	Tomat Mengkal	10	6	4	60%
3	Tomat Matang	10	8	2	80%
4	Jeruk Mentah	10	7	3	70%
5	Jeruk Mengkal	10	6	4	60%
6	Jeruk Matang	10	7	3	70%
Rata-rata		60	42	18	70%

Berdasarkan Tabel 5.5, sistem mencapai akurasi pemilahan fisik rata-rata sebesar 70%. Jika dibandingkan dengan akurasi deteksi visual (pada sub-bab sebelumnya) yang mencapai 95%, terdapat penurunan kinerja saat sistem memasuki ranah eksekusi mekanik. Analisis terhadap kegagalan (18 sampel) menunjukkan bahwa faktor utama penyebabnya adalah ketepatan waktu (*timing*) servo dan dinamika fisik buah. Buah yang memiliki bentuk tidak bulat sempurna cenderung menggelinding (*rolling*) saat berada di atas konveyor, sehingga posisinya bergeser setelah dideteksi kamera. Akibatnya, saat tiba di titik eksekusi, buah tersebut sudah melewati jangkauan lengan servo.

5.3.5. Pengujian Fitur Penyimpanan Data

Salah satu fitur unggulan dari sistem ini adalah kemampuan untuk menyimpan riwayat penyortiran secara otomatis ke dalam database. Fitur ini berfungsi sebagai log data produksi yang merekam setiap aktivitas deteksi yang valid. Pengujian dilakukan untuk memastikan bahwa setiap buah yang terdeteksi oleh kamera akan tercatat secara real-time pada menu "History" di antarmuka Web Dashboard.

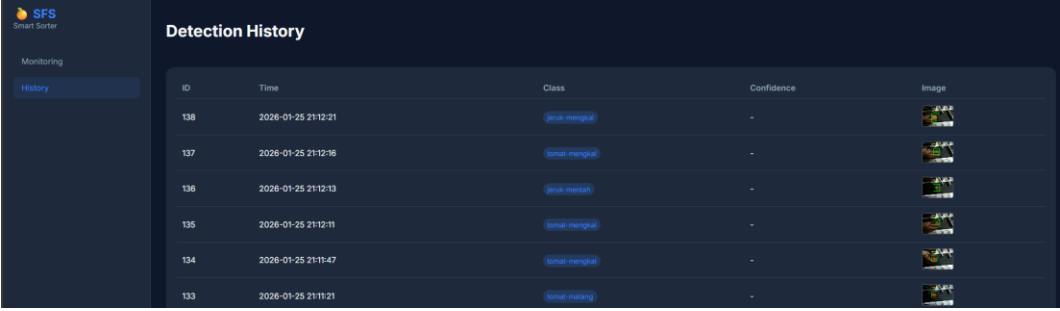
Parameter data yang disimpan meliputi:

1. ID: Nomor urut unik untuk setiap data.
2. Time: Waktu dan tanggal deteksi (timestamp).
3. Class: Jenis dan tingkat kematangan buah (misal: jeruk-mengkal, tomat-mentah).
4. Confidence: Tingkat keyakinan deteksi.
5. Image: Thumbnail citra buah saat terdeteksi lengkap dengan bounding box.

Tabel 5.6 Pengujian Fitur History Data

No	Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengujian	Status
1	Deteksi Buah Baru	Data baru muncul di baris teratas tabel History	Data muncul seketika (<i>real-time</i>)	Berhasil
2	Kelengkapan Data	Kolom Waktu, Kelas, dan Gambar terisi	Semua kolom terisi dengan data yang sesuai	Berhasil
3	Penyimpanan Gambar	<i>Thumbnail</i> gambar hasil deteksi dapat dilihat	Gambar tersimpan dan tampil pada kolom <i>Image</i>	Berhasil

Hasil pengujian menunjukkan bahwa fitur *logging* berjalan dengan baik. Tampilan antarmuka halaman riwayat data dapat dilihat pada Gambar 5.5.



ID	Time	Class	Confidence	Image
138	2026-01-25 21:12:21	jeruk-mengkal	-	
137	2026-01-25 21:12:16	jeruk-mengkal	-	
136	2026-01-25 21:12:13	jeruk-mengkal	-	
135	2026-01-25 21:12:11	jeruk-mengkal	-	
134	2026-01-25 21:11:47	jeruk-mengkal	-	
133	2026-01-25 21:11:21	jeruk-mengkal	-	

Gambar 5.5 Tampilan Halaman Riwayat Data (*History*) pada Web Dashboard

Seperti terlihat pada Gambar 5.5, sistem mampu merekam data secara rinci. Sebagai contoh pada data ID 138, sistem mencatat deteksi "jeruk-mengkal" pada pukul 21:12:21. Fitur ini memberikan transparansi data yang krusial untuk pemantauan hasil panen, memungkinkan pengguna untuk melacak performa penyortiran dan meninjau kembali bukti visual (gambar) jika terjadi kesalahan klasifikasi.

5.3.6 Pengujian Kecepatan Conveyor

Pengujian kecepatan conveyor dilakukan untuk mengetahui kemampuan conveyor dalam memindahkan buah menuju area deteksi dan penyortiran pada sistem yang telah dirancang. Kecepatan conveyor menjadi salah satu faktor penting yang mempengaruhi kestabilan objek saat proses pendeteksian menggunakan kamera serta ketepatan waktu aktuator dalam melakukan proses pemilahan buah. Pengujian dilakukan dengan mengukur waktu yang dibutuhkan conveyor untuk

menempuh jarak tertentu, kemudian dari hasil pengukuran tersebut dihitung nilai kecepatan conveyor. Hasil pengujian kecepatan conveyor dapat dilihat pada Tabel 5.7 berikut.

Tabel 5. 7 Pengujian Kecepatan Conveyor

No	Jarak Conveyor (cm)	Waktu Tempuh (detik)	Kecepatan (cm/s)	RPM (putaran/menit)
1	70	5,2	11,54	44,09
2	70	5,0	12,00	45,86
3	70	5,1	11,76	44,94
4	70	5,3	11,32	43,26
5	70	5,1	11,76	44,94

5.4 KESIMPULAN HASIL PENGUJIAN

Berdasarkan serangkaian pengujian yang telah dilakukan, mulai dari pengujian fungsionalitas komponen, akurasi deteksi visual, hingga pemilahan fisik secara real-time, dapat ditarik beberapa kesimpulan penting mengenai kinerja sistem "Rancang Bangun Alat Sortir Buah Tomat dan Jeruk" ini:

1. **Keandalan Sistem Visi Komputer:** Algoritma YOLOv8 terbukti sangat efektif sebagai sensor visual cerdas. Hasil pengujian deteksi objek menunjukkan tingkat akurasi rata-rata sebesar **95%**, yang mengindikasikan bahwa model mampu membedakan karakteristik visual buah tomat dan jeruk serta tingkat kematangannya (mentah, mengkal, matang) dengan sangat presisi dalam kondisi statis maupun bergerak lambat.

2. **Kesenjangan Kinerja Mekanik:** Terdapat penurunan performa yang signifikan saat sistem beralih dari deteksi visual ke eksekusi fisik. Akurasi pemilahan turun menjadi **70%** pada pengujian *real-time* sistem keseluruhan. Hal ini menunjukkan bahwa kendala utama sistem bukan terletak pada logika pemrograman atau kecerdasan buatan, melainkan pada aspek mekanik, khususnya kestabilan posisi buah saat bergerak di atas konveyor dan sinkronisasi waktu tunda (*delay*) servo.
 3. **Kendala pada Kategori Transisi:** Buah dengan kategori "Mengkal" (setengah matang) konsisten menjadi kategori dengan tingkat kesalahan tertinggi, baik secara visual maupun fisik (akurasi 60%). Hal ini disebabkan oleh ambiguitas warna kulit buah yang sangat dipengaruhi oleh intensitas pencahayaan lingkungan, serta kemiripan fitur dengan kategori mentah atau matang.
 4. **Stabilitas Integrasi Sistem:** Secara fungsional, integrasi antara perangkat lunak Python dan mikrokontroler ESP32 melalui komunikasi serial berjalan stabil tanpa kegagalan transmisi data. Fitur pendukung seperti *Web Dashboard* dan pencatatan riwayat (*History*) berfungsi 100% akurat, memberikan nilai tambah yang signifikan bagi pengguna untuk memantau hasil produksi secara digital.
- Secara keseluruhan, sistem ini berhasil membuktikan konsep (*proof of concept*) bahwa teknologi *Deep Learning* dapat diterapkan untuk otomasi pertanian biaya rendah, meskipun memerlukan penyempurnaan lebih lanjut pada desain mekanik konveyor untuk mencapai akurasi pemilahan fisik yang setara dengan akurasi deteksi visualnya.