

Edge AI Berbasis Computer Vision Untuk Meningkatkan Efektivitas Sistem Deteksi Pemilahan Sampah Real-Time Integrasi YOLOv8, Raspberry Pi 5 dan SEE

Syaifuddin¹, Ifriandi Labolo², Nuranissa D. Paemo^{1*}, Abdul Malik I Buna³

¹Fakultas Fakultas Ilmu Komputer dan Sains, Manajemen Informatika, Universitas Ichsan Gorontalo Utara, Gorontalo, Indonesia

²Fakultas Fakultas Ilmu Komputer dan Sains, Sistem Informasi, Universitas Ichsan Gorontalo Utara Gorontalo, Indonesia

³Fakultas Fakultas Ilmu Komputer, Prodi Informatika, Universitas Ichsan Sidenreng Rappang, Indonesia

Email: ¹syaifuddinlily@gmail.com, ²iadifriandi@gmail.com, ^{3*}nuranisapaemo@gmail.com, ⁴mailqlloex@gmail.com

Email Penulis Korespondensi : nuranisapaemo@gmail.com

Submitted 01-11-2025; Accepted 10-01-2026; Published 19-02-2026

Abstrak

Permasalahan pengelolaan sampah di Indonesia masih ditandai oleh tingginya volume sampah yang tidak terkelola dan lemahnya pemilahan sejak dari sumber, sehingga material yang berpotensi didaur ulang sering tercampur dan menurunkan nilai pemanfaatannya. Kondisi ini menuntut ketersediaan sistem pemilahan yang efektif, cepat, dan terjangkau, sekaligus mampu menyediakan informasi operasional secara real-time untuk mendukung pengambilan keputusan di lapangan. Penelitian ini mempresentasikan integrasi Computer Vision berbasis YOLOv8 pada Raspberry Pi 5 dengan Camera Module 3, yang terhubung ke sistem informasi real-time melalui Server-Sent Events (SSE) untuk memonitoring dan analitik. Metodologi meliputi penyusunan dataset beranotasi format YOLO TXT, pelatihan model YOLOv8n, implementasi inferensi pada perangkat edge, serta pengembangan API backend yang menerima hasil deteksi dan menampilkan ke dashboard secara real-time. Evaluasi dilakukan menggunakan metrik mean Average Recall (mAP), Precision-Recall, frame per second (FPS), dan latensi end-to-end dari kamera hingga dashboard. Hasil menunjukkan prototipe mampu mencapai mAP@0.5 sebesar 98.5% dengan Precision-Recall diatas 97%, throughput rata-rata 8.3 FPS pada resolusi 640x640 piksel, dan latensi median 0.5-0.6 ms untuk komunikasi SSE, yang menegaskan kelayakan solusi cost-effective untuk pemilahan sampah otomatis. Sistem menyediakan fitur logging, statistik operasional, offline queue dan mekanisme Idempotency untuk mendukung implementasi operasional yang reliable.

Kata Kunci: YOLOv8; Raspberry Pi; Computer Vision; Real-time; Pemilahan Sampah

Abstract

Waste management in Indonesia is still characterized by a high volume of improperly managed waste and low source-level segregation, causing recyclable materials to mix with other waste streams and reducing their recovery value. This situation calls for a sorting system that is effective, fast, and affordable, while also providing real-time operational information to support on site decision-making. This study presents an integrated computer vision approach using YOLOv8 deployed on a Raspberry Pi 5 with a Camera Module 3, connected to a real-time information system via Server-Sent Events (SSE) for monitoring and analytics. The methodology includes constructing a labeled dataset in YOLO TXT format, training a YOLOv8n model, deploying edge inference, and developing a backend API to receive detection outputs and stream them to a dashboard in real time. The system is evaluated using mean Average Precision (mAP), precision-recall, frames per second (FPS), and end-to-end latency from the camera to the dashboard. The prototype achieves an mAP@0.5 of 98.5% with precision-recall above 97%, an average throughput of 8.3 FPS at 640x640 resolution, and a median SSE communication latency of 0.5–0.6 ms, demonstrating the feasibility of a cost-effective solution for automated waste sorting. The system also provides logging, operational statistics, an offline queue, and an idempotency mechanism to support reliable operation in real-world deployments.

Keywords: YOLOv8; Raspberry Pi; Computer Vision; Real-time; Waste Sorting

1. PENDAHULUAN

Permasalahan pengelolaan sampah menjadi tantangan global yang semakin kompleks, terutama di negara berkembang seperti Indonesia. Berdasarkan data Kementerian Lingkungan Hidup dan Kehutanan (LHKH) tahun 2024, Indonesia menghasilkan sekitar 20.2 juta ton sampah dengan 34.76% atau 10.77 juta ton di antaranya tidak terkelola dengan baik [1], dengan komposisi organik 40-45% dan non organik 55-60% [2]. Kondisi ini diperparah oleh rendahnya kesadaran masyarakat dalam memilah sampah sejak dari sumber [3], yang menyebabkan banyak jenis sampah yang berpotensi didaur ulang tercampur dengan sampah lain sehingga terkontaminasi dan tidak dapat dimanfaatkan kembali [4].

Proses pemilahan sampah konvensional yang mengandalkan tenaga manusia memiliki beberapa kelemahan fundamental [5]. Praktik pemilahan manual rentan terhadap kelelahan, inkonsistensi dalam klasifikasi dan potensi risiko kesehatan bagi pekerja. Selain itu, pemilahan manual memerlukan sumber daya manusia yang besar dan biaya operasional tinggi, khususnya untuk volume sampah yang besar. Pengambil keputusan di lapangan juga membutuhkan visibilitas real-time mengenai status pemilahan agar intervensi dapat dilakukan segera.

Perkembangan teknologi *Artificial Intelligence* (AI), khususnya dalam bidang *Computer Vision*, membuka peluang baru untuk mengembangkan sistem pemilahan sampah otomatis yang lebih efisien dan akurat. Teknologi *Computer Vision* memungkinkan mesin memperoleh, memproses dan menganalisis data visual, memberikan fungsi “penglihatan” pada komputer untuk mengenali dan mengklasifikasikan objek secara otomatis [6]. Algoritma *YOU Only Look Once* (YOLO), khususnya YOLOv8, telah terbukti efektif dalam deteksi objek real-time dengan akurasi tinggi pada berbagai kategori sampah seperti plastik, kertas dan bahan organik [7][8]. Penelitian pengelolaan sampah berbasis visi komputer dan pembelajaran mendalam berkembang pesat, mencakup klasifikasi gambar, deteksi objek, pemantauan

tempat sampah pintar, hingga analitik operasional. Kajian tinjauan terbaru menegaskan peran AI/CV di seluruh rantai nilai pengelolaan sampah, sekaligus menyoroti tantangan data, generalisasi model, dan integrasi sistem IoT/edge sebagai riset yang masih terbuka. Untuk benchmarking, komunitas banyak menggunakan dataset terbuka seperti TACO (Trash Annotation in Context) yang menyediakan citra “*in-the-wild*” dengan anotasi untuk deteksi/segmentasi[9].

Keluarga YOLO menjadi pilihan populer untuk deteksi objek sampah karena kecepatan dan akurasi[10][11]. Studi terbaru menunjukkan YOLOv8 dapat mencapai akurasi tinggi pada beragam kategori seperti plastik, sisa makanan, baterai dan kaca, menunjukkan potensi praktis untuk sistem pemilahan otomatis, penelitian lain di ranah *e-waste* juga memperlihatkan kinerja kuat pada skenario khusus[12]. Disisi *edge/embedded inference*, beberapa studi membandingkan kinerja model deteksi pada perangkat sumber daya terbatas, termasuk Raspberry Pi 5 dan platform NVIDIA Jetson. Hasilnya secara umum menegaskan adanya trade-off akurasi, kecepatan, dan konsumsi energi perangkat yang lebih bertenaga seperti Jetson Orin cenderung unggul dalam throughput, namun Pi 5 tetap relevan untuk solusi biaya rendah dengan konfigurasi model ringkas dan optimasi yang tepat[13][14].

Implementasi sistem AI pada perangkat *edge computing*[15], seperti Raspberry Pi 5 dengan Camera Module 3, menjadi solusi praktis untuk deployment sistem deteksi objek dengan biaya rendah. *Edge computing* mengurangi ketergantungan pada koneksi internet dan meminimalkan latensi dalam proses deteksi *real-time*. Untuk mendukung pengambilan keputusan operasional, integrasi sistem informasi *real-time* menggunakan teknologi *Server-Sent Events (SSE)* memungkinkan monitoring dan analitik yang diperlukan untuk optimalisasi proses pemilahan sampah.

Meskipun berbagai penelitian telah menunjukkan potensi teknologi AI untuk pemilahan sampah, terdapat gap penelitian yang signifikan dalam hal integrasi sistem *end-to-end* yang mencakup seluruh alur kerja dari akuisisi data hingga visualisasi hasil. Banyak penelitian masih berfokus pada metrik model dilingkungan terkontrol, sedangkan aspek operasional seperti latensi *end-to-end* dan integrasi sistem informasi *real-time* belum dilaporkan secara komprehensif[16][17]. Berdasarkan gap penelitian tersebut, penelitian ini bertujuan untuk merancang dan mengembangkan sistem pemilahan sampah otomatis berbasis AI dan Computer Vision yang terintegrasi secara *end-to-end*. Sistem menggunakan algoritma YOLOv8 yang diimplementasikan pada Raspberry Pi 5 dengan Camera Module 3, serta menyediakan sistem informasi *real-time* berbasis *Server-Sent Events* untuk monitoring operasional.

Rumusan masalah dalam penelitian ini meliputi yang pertama Bagaimana merancang dan mengembangkan sistem pemilahan sampah otomatis berbasis AI dan Computer Vision yang memiliki tingkat akurasi tinggi dalam mengenali dan mengklasifikasi jenis sampah secara *real-time*. Kedua bagaimana mengintegrasikan model AI yang telah dikembangkan ke dalam perangkat fisik, sehingga sistem dapat bekerja secara otomatis dan efisien di lingkungan nyata. Tujuan penelitian ini adalah mengembangkan prototipe sistem pemilahan sampah otomatis yang *cost-effective* dan *scalable*, serta mengevaluasi kinerja sistem secara komprehensif meliputi metrik akurasi model, kinerja *edge computing* dan responsivitas sistem informasi real-time. Kontribusi utama penelitian mencakup pengembangan arsitektur terpadu end-to-end, implementasi praktis dengan fitur *offline queue* dan *Idempotency*, serta evaluasi menyeluruh yang mencakup metrik model dan metrik sistem. Penelitian ini diharapkan dapat memberikan kontribusi berupa implementasi praktis teknologi AI untuk pengelolaan sampah yang dapat diadopsi secara operasional.

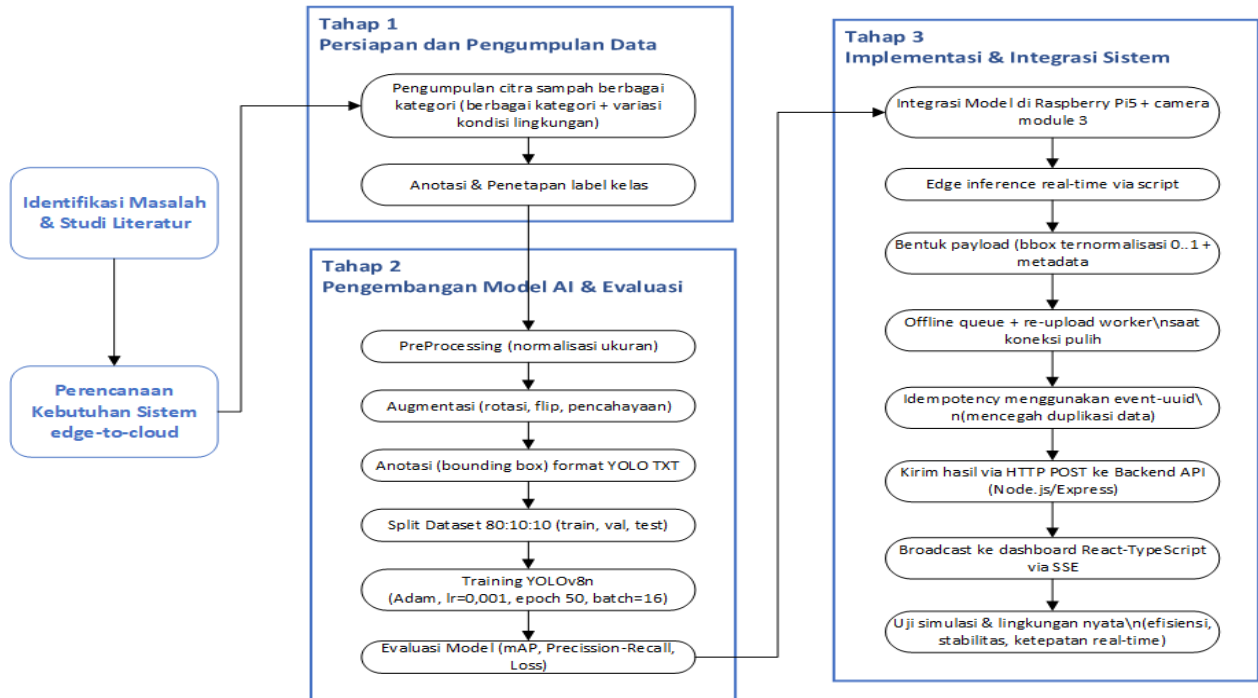
2. METODOLOGI PENELITIAN

2.1 Jenis dan Desain Penelitian

Penelitian ini menggunakan pendekatan kuantitatif dengan desain eksperimental untuk mengembangkan dan mengevaluasi sistem pemilahan sampah otomatis berbasis AI dan Computer Vision. Metode penelitian yang digunakan adalah *Research and Development (R&D)* dengan model pengembangan sistem yang mengintegrasikan *Machine Learning Operations (MLOps)*[18] dan *edge computing* [19].

2.2 Tahapan Penelitian

Penelitian ini dilakukan melalui tiga tahapan utama, sebagaimana ditunjukkan pada Gambar 1.



Gambar 1. Tahapan utama penelitian

Berdasarkan gambar 1 berikut adalah penjelasan tahapan penelitian ini :

- Tahap pertama adalah persiapan dan pengumpulan data
Tahap ini dimulai dengan identifikasi masalah dan studi literatur untuk memahami perkembangan terkini dalam bidang AI, khususnya Computer Vision dan sistem klasifikasi berbasis YOLO. Pengumpulan dataset dilakukan dari berbagai kategori yang umum di Indonesia dalam berbagai kondisi lingkungan untuk mewakili variasi nyata saat penerapan di lapangan.
- Tahap tahapan kedua adalah pengembangan model AI dan evaluasi
Gambar yang dikumpulkan melalui proses preprocessing meliputi normalisasi ukuran, augmentasi data (rotasi, flip dan pencahayaan), serta anotasi berdasarkan katageori sampah. Dataset teranotasi digunakan untuk melatih model YOLOv8 dengan *hyperparameter* optimizer Adam, 50 *epoch*, *learning rate* 0.001, dan *batch size* 16. Model di evaluasi menggunakan metrik mAP, *Recall*, recall dan loss function.
- Tahap tahapan ketiga adalah implementasi dan integrasi sistem
Model AI yang telah dilatih diintegrasikan ke dalam sistem fisik yang terdiri dari kamera sebagai input citra dan Raspberry Pi 5 sebagai perangkat pemroses[20]. Sistem diuji dalam skenario simulasi dan lingkungan nyata untuk menilai efisiensi operasional, stabilitas kerja dan ketepatan deteksi *real-time*.

2.3 Waktu dan Tempat Penelitian

Penelitian dilaksanakan selama 12 Bulan dan dilaksanakan di Laboratorium Sistem Informasi dan Laboratorium Jaringan Komputer, Fakultas Sains dan Teknologi, Universitas Ichsan Gorontalo Utara. Pengujian sistem dilakukan di lingkungan kampus dengan kondisi jaringan LAN/Wi-Fi untuk simulasi lingkungan operasional nyata. Pengumpulan dataset gambar sampah dilakukan di berbagai lokasi, meliputi Tempat Penampungan Sementara (TPS) indoor dan area luar ruangan untuk memastikan variasi kondisi pencahayaan dan latar belakang.

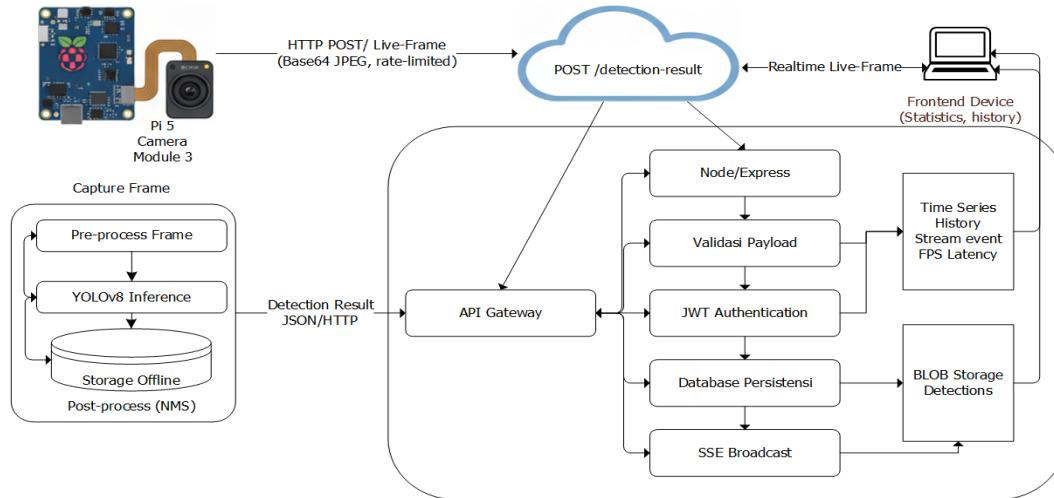
2.4 Target dan Subjek Penelitian

Target penelitian ini adalah pengembangan prototipe sistem pemilahan sampah otomatis yang dapat beroperasi secara real-time dengan tingkat akurasi tinggi pada perangkat *edge computing cost-effective*. Subjek penelitian terdiri dari :

- Dataset sampah yang dikumpulkan secara mandiri dari beberapa lokasi (TPS indoor, area luar ruang) dengan variasi pencahayaan, latar dan occlusion. Target kategori klasifikasi adalah organik dan Non-organik dengan kemungkinan perluasan ke plastik, kertas, logam, kaca dan daun. Pembagian train 80%, validation 10% dan test 10% berbasis scene untuk menguji generalisasi. Anotasi mengikuti format YOLO TXT (*kelas, x_center, y_center, width, height* dalam koordinat ternormalisasi). Kualitas anotasi dijaga melalui pedoman label dan review sampel untuk memastikan konsistensi inter-annotator agreement.
- Perangkat sistem terdiri dari (1) sistem *edge computing* yaitu Raspberry Pi 5 dengan Camera Module 3 sebagai platform inferensi, (2) sistem backend yaitu server dengan *Node.js/Express* dan database MySQL untuk persistensi data, serta (3) dashboard Real-time yaitu interface monitoing berbasis *React-TypeScript* dengan koneksi *Server-Sent Events*.

2.5 Arsitektur Sistem

Berikut merupakan arsitektur sistem yang diusulkan, sebagaimana ditunjukkan pada Gambar 2



Gambar 2. Diagram Arsitektur Sistem

Pada gambar 2 menampilkan arsitektur sistem mengikuti pola MLOps dengan integrasi *edge-to-cloud*. Data gambar dianotasi menggunakan Label Studio dengan format YOLO TXT, kemudian digunakan untuk melatih model YOLOv8. Dilapisan edge, Raspberry Pi 5 dengan Camera Module 3 menangkap frame video, melakukan inferensi, dan mengirim hasil via HTTP POST ke backend. Backend API (*Node.js/Express*) memvalidasi *payload*, menyimpan data di MySQL dan menyiarkan event melalui *Server-Sent Events* ke dashboard *React-TypeScript*.

2.6 Instrumen dan Teknik Pengumpulan Data

a. Instrumen Perangkat Keras

Implementasi sistem menggunakan konfigurasi perangkat keras yang *cost-effective* namun memadai untuk operasi *real-time*. Platform *edge computing* menggunakan Raspberry Pi 5 dengan prosesor 2.4GHz quad-core 64-bit Arm Cortex-A76 dan SDRAM 8GB dan dipasangkan dengan Camera Module 3 berfitur 12-megapixel Sony IMX708 image sensor.

b. Instrumen Perangkat Lunak

1. *Edge computing* : Python dengan Library *OpenCV*, Ultralytics YOLOv8
2. *Backend System* : *Node.js* 18+, framework *Express* dan *MySQL* untuk persistensi data
3. *Fronted Dashboard* : *React-TypeScript* dengan *Vite* untuk monitoring interface
4. *Data annotation* : label studio untuk anotasi dataset
5. *Development Tools* : Git untuk *version control* dan *Docker* untuk *containerization*

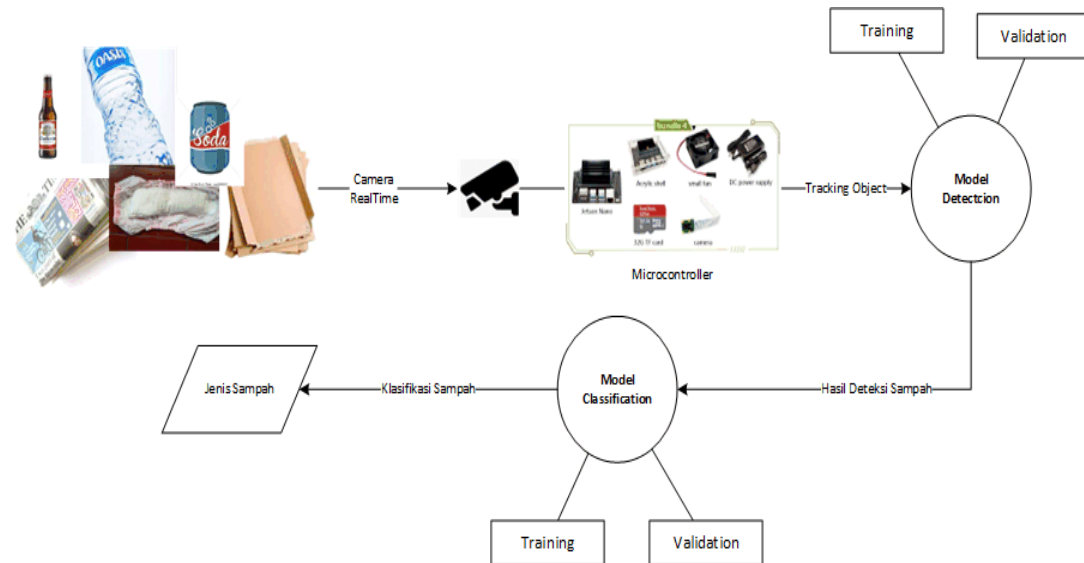
c. Teknik Analisis Data

Analisis kuantitatif meliputi statistik deskriptif untuk metrik akurasi latensi dan *throughput*, selanjutnya perbandingan kinerja antar konfigurasi menggunakan *t-test*. Untuk analisis korelasi antara resolusi input dengan akurasi dan FPS dan selanjutnya visualisasi confusion matrix dan *Precision-Recall curve*.

d. Analisis Sistem

1. *Profiling* konsumsi CPU dan memori pada perangkat edge
2. Monitoring latensi *end-to-end* dari capture hingga *dashboard*
3. Evaluasi reliabilitas koneksi dan mekanisme *offline queue*
4. Analisis *throughput* dan bottleneck sistem secara keseluruhan.

Alur pengumpulan dan pemrosesan data dalam penelitian ini mengikuti metodologi sistematis yang terstruktur sebagaimana ditunjukkan pada gambar 3.



Gambar 3. Diagram Alur Penelitian

Pada gambar 3 menampilkan alur pengumpulan dan pemrosesan data dalam penelitian ini mengikuti metodologi sistematis yang terstruktur sebagaimana ditunjukkan pada gambar, dimulai dari pengambilan citra berbagai jenis sampah secara real-time menggunakan kamera yang terhubung ke mikrokontroler untuk melakukan pelacakan objek dan menghasilkan data deteksi awal. Hasil deteksi tersebut kemudian menjadi input bagi tahap *Model Detection* yang dilatih dan divalidasi secara terpisah, sebelum diteruskan ke *Model Classification* yang juga melalui proses training dan validation untuk mengklasifikasikan jenis sampah secara lebih spesifik. Output akhir berupa informasi jenis sampah dari modul klasifikasi inilah yang digunakan sebagai dasar penentuan kategori sampah pada sistem pemilahan otomatis.

3. HASIL DAN PEMBAHASAN

3.1 Implementasi Sistem

a. Pengembangan Dataset dan Model

Dataset dikumpulkan dari berbagai lokasi untuk klasifikasi organik/non-organik menggunakan format YOLO TXT dengan pembagian 80:10:10 (train:val:test).

b. Implementasi Edge Computing

Pada lapisan edge, sistem dioperasikan oleh skrip *edge_yolo_cam.py* yang dikembangkan dalam *Python*. Bertanggung jawab penuh atas pengambilan frame dari kamera menggunakan *OpenCV*, melakukan inferensi deteksi objek dengan Ultralytics YOLOv8n, serta membentuk *payload* data. Payload mencakup koordinat *bounding box* yang telah dinormalisasi (rentang 0 hingga 1) dan secara opsional menyertakan ketahanan, skrip dilengkapi dengan *offline queue* dan *re-upload worker* yang akan mengirimkan data yang sempat tertunda saat koneksi pulih. Kinerja operasional *edge device* dapat dilihat dari log keluaran sistem menampilkan stabilitas proses *capture*, inferensi dan komunikasi ditunjukkan pada gambar 4.

```
(yolo-env) D:\RISET\SYAIFUDDIN\smart-waste>python edge_yolo_cam.py
UPLOAD SUCCESS (201) | items: 1 | response: {"message":"Detection created successfully","data":{"detection_id":6232}}
UPLOAD SUCCESS (201) | items: 1 | response: {"message":"Detection created successfully","data":{"detection_id":6233}}
UPLOAD SUCCESS (201) | items: 1 | response: {"message":"Detection created successfully","data":{"detection_id":6234}}
UPLOAD SUCCESS (201) | items: 1 | response: {"message":"Detection created successfully","data":{"detection_id":6235}}
UPLOAD SUCCESS (201) | items: 1 | response: {"message":"Detection created successfully","data":{"detection_id":6236}}
UPLOAD SUCCESS (201) | items: 1 | response: {"message":"Detection created successfully","data":{"detection_id":6237}}
UPLOAD SUCCESS (201) | items: 2 | response: {"message":"Detection created successfully","data":{"detection_id":6238}}
UPLOAD SUCCESS (201) | items: 1 | response: {"message":"Detection created successfully","data":{"detection_id":6239}}
UPLOAD SUCCESS (201) | items: 1 | response: {"message":"Detection created successfully","data":{"detection_id":6240}}
```

Gambar 4. Log proses capture dan inferensi pada Edge Pi 5 Camera Module 3

Pada gambar 4 menampilkan log eksekusi script *edge_yolo_cam.py* pada environment *yolo-env* yang menunjukkan proses pengiriman (upload) hasil deteksi dari perangkat edge ke backend API berjalan sukses. Setiap baris "UPLOAD SUCCESS (201)" menandakan server mengembalikan HTTP status 201 (*Created*), disertai respons JSON "Detection created successfully" dan *detection_id* yang terus bertambah, yang mengindikasikan data deteksi berhasil dibuat dan tersimpan secara berurutan di sisi backend

c. Integrasi Backend dan Frontend

Pada Gambar 5, terlihat di sisi *backend* log dari HTTP request aplikasi Express yang menyediakan *endpoint POST /api/v1/detections* dirancang untuk menerima data deteksi dalam format *JSON* atau *multipart*. Data yang masuk akan

disimpan ke tabel *detections* dan *detection items* dalam basis data. *Backend* juga mengelola *endpoint SSE* di */api/v1/sse/stream* yang berfungsi untuk menyiarkan event deteksi dan frame secara *real-time* ke klien yang terhubung. Komunikasi antara *edge device* dan *backend server* memperlihatkan performa yang optimal dengan *response time* yang konsisten.

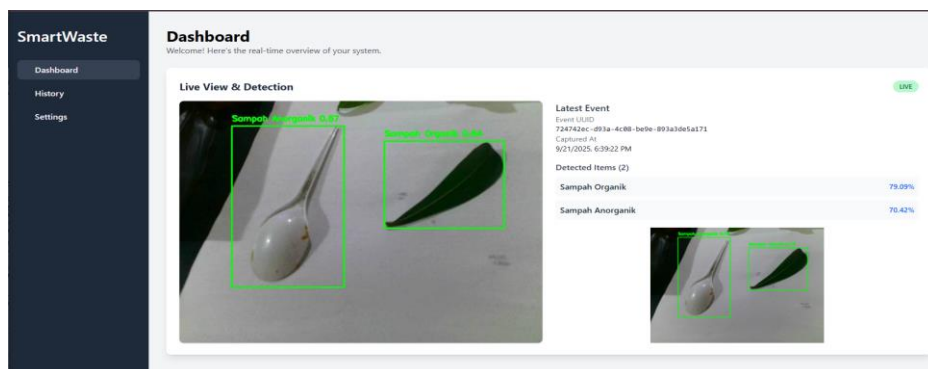
```
POST /api/v1/sse/live 202 0.543 ms - -
SSE: Sending event to 1 client(s)
POST /api/v1/detections 201 6.258 ms - 73
GET /snapshots/39afecea-6008-494a-9f0c-3447d8b08e72.jpg 200 2.197 ms - 83819
GET /api/v1/settings/livestream 200 1.148 ms - 16
POST /api/v1/sse/live 202 0.605 ms - -
POST /api/v1/sse/live 202 0.586 ms - -
POST /api/v1/sse/live 202 0.604 ms - -
```

Gambar 5. Log HTTP respon time request antara Edge Backend dan Frontend

Pada gambar 5 menampilkan log akses backend (*Node/Express*) yang menunjukkan alur komunikasi *real-time* dan penyimpanan data berjalan normal. Terlihat *endpoint SSE* */api/v1/sse/live* menerima request *POST* dengan status 202 dan server menuliskan “*Sending event to 1 client(s)*”, yang menandakan backend sedang membroadcast event ke satu klien dashboard yang terhubung. Selain itu, request *POST* ke */api/v1/detections* menghasilkan status 201 (data deteksi berhasil dibuat/tersimpan) sementara request *GET* ke file *snapshot.jpg* dan *endpoint* lain seperti */api/v1/settings/livestream* mengembalikan status 200, yang mengindikasikan *snapshot* dan konfigurasi *live stream* dapat diakses dengan baik.

d. Dashboard dan Monitoring

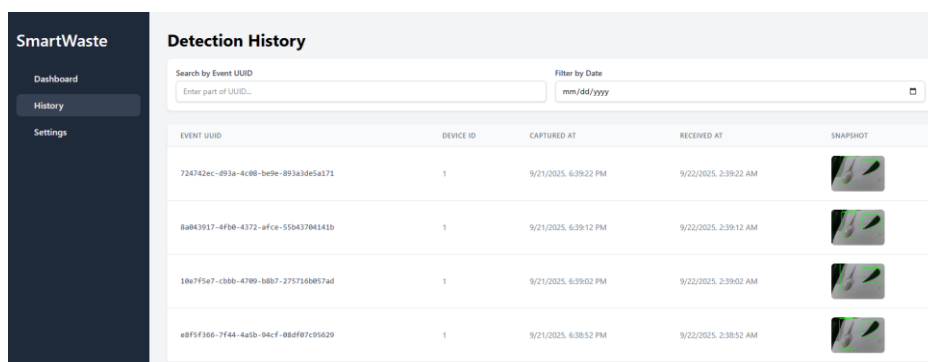
Implementasi antarmuka monitoring *real-time* pada Dashboard SmartWaste berbasis Vite dan React-TS telah berhasil dilakukan, di mana dashboard menerima pembaruan data secara langsung dari backend melalui Server-Sent Events (SSE), menampilkan visualisasi hasil deteksi pada *live view*, serta menyajikan ringkasan event terbaru untuk kebutuhan pemantauan operasional, sebagaimana ditunjukkan pada gambar 6.



Gambar 6. Dashboard Live Stream & Detections

Pada gambar 6 menampilkan halaman Dashboard SmartWaste untuk monitoring deteksi sampah secara *real-time*, dengan area *Live View & Detection* yang memperlihatkan hasil deteksi berupa bounding box hijau, label kelas (sampah organik/anorganik), dan nilai confidence pada objek di dalam frame kamera. Di sisi kanan, panel Latest Event menampilkan informasi event terbaru seperti *event UUID*, waktu tangkapan, jumlah objek terdeteksi, ringkasan kategori beserta confidence, serta *thumbnail snapshot* untuk verifikasi cepat, sementara indikator “LIVE” menunjukkan data diperbarui secara langsung.

Selanjutnya fitur *historical* data analisis memungkinkan penelusuran *trend* dan *pattern* deteksi dalam rentang waktu tertentu seperti ditunjukkan pada gambar 7.



EVENT UUID	DEVICE ID	CAPTURED AT	RECEIVED AT	SNAPSHOT
724742ec-d93a-4c88-be9e-893a3de5a771	1	9/21/2025, 6:39:22 PM	9/22/2025, 2:39:22 AM	
8a843917-4f60-4372-afce-55843704341d	1	9/21/2025, 6:39:12 PM	9/22/2025, 2:39:12 AM	
18e7f5e7-c360-4789-4807-2757160857ad	1	9/21/2025, 6:39:02 PM	9/22/2025, 2:39:02 AM	
e8f9f386-7f44-4a10-94cf-880f07c95629	1	9/21/2025, 6:38:52 PM	9/22/2025, 2:38:52 AM	

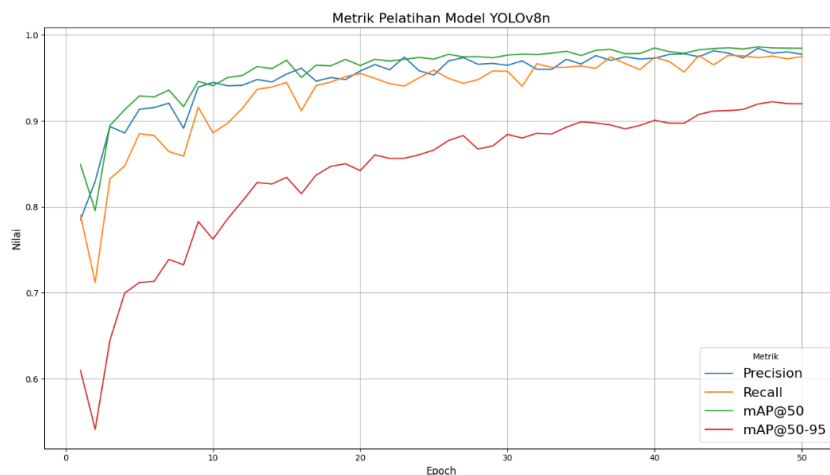
Gambar 7. Frontend History Detection

Pada gambar 7 menampilkan halaman Detection History pada Dashboard SmartWaste yang berfungsi untuk menelusuri dan mengaudit riwayat hasil deteksi yang telah tersimpan. Pada bagian atas tersedia fitur pencarian berdasarkan *Event UUID* serta filter berdasarkan tanggal, sehingga pengguna dapat menemukan kejadian tertentu dengan cepat. Tabel histori menampilkan atribut utama setiap event, seperti Event UUID, Device ID, waktu *captured at* (waktu frame diambil di perangkat) dan *received at* (waktu data diterima server), serta kolom *snapshot* sebagai pratinjau gambar hasil deteksi untuk memudahkan verifikasi visual.

3.2 Evaluasi Kinerja Model

a. Pelatihan dan akurasi Model YOLOv8n

Model YOLOv8n berhasil menggunakan *hyperparameter optimizer* Adam dan dilatih selama 50 *epoch*, *learning rate* awal 0.001 dan ukuran batch 16. Berdasarkan hasil pelatihan yang disajikan, model menunjukkan peningkatan akurasi yang signifikan dan konvergen seiring berjalannya *epoch*. Metrik presisi dan *recall* dengan cepat mencapai nilai yang sangat tinggi, stabil diatas 97%, mengindikasikan kemampuan model secara konsisten mengidentifikasi objek yang relevan dengan benar dan meminimalkan deteksi yang salah. Progress pelatihan model YOLOv8n merepresentasikan konvergensi yang stabil dengan peningkatan akurasi yang konsisten, sebagaimana ditunjukkan kurva pelatihan gambar 8.

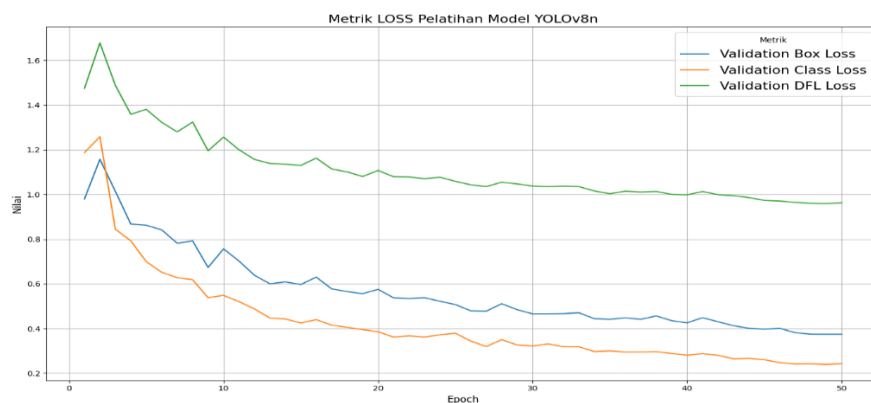


Gambar 8. Metrik Akurasi Pelatihan Model YOLOv8n (precision, recall, mAP50, mAP50-95)

Hal ini diperkuat oleh metrik *mean Average Recall (mAP)*, dimana mAP50 (B) mencapai puncaknya di sekitar 98.5%, menunjukkan kinerja deteksi hampir sempurna pada ambang batas *IoU* 0.5. Selanjutnya, metrik mAP50-95(B) mengukur akurasi pada berbagai ambang batas *IoU*, menunjukkan tren peningkatan yang konsisten dan mencapai sekitar 90%.

b. Analisis Loss Fuction

Analisis kurva kerugian (loss) menunjukkan proses pelatihan yang sehat dan efektif. Baik komponen kerugian pelatihan (train loss) maupun validasi (validation loss), termasuk *box loss*, *class loss* dan *dfl loss*, menunjukkan tren penurunan yang tajam pada *epoch* awal dan terus menurun secara stabil. Kurva kerugian validasi bergerak selaras dengan kurva kerugian pelatihan dan konvergen pada nilai yang rendah, tanpa adanya tanda-tanda *overfitting*. Analisis fungsi kerugian selama proses pelatihan memberikan *insight* mendalam tentang proses pembelajaran model, sebagaimana ditunjukkan pada gambar 9.



Gambar 9. Metrik LOSS pelatihan model YOLO (Box Loss, Class Loss, DFL Loss)

Penurunan nilai *val/box_loss* hingga sekitar 0.42 dan *val/cls_loss* hingga 0.28 pada akhir pelatihan menandakan bahwa model berhasil menggeneralisasi pengetahuannya dengan baik pada data yang belum pernah dilihat sebelumnya, menghasilkan model yang robust dan andal.

3.3 Evaluasi Kinerja Sistem

a. Matriks Keterlacakan

Pengujian fungsionalitas sistem dilakukan menggunakan *test case* yang telah dirancang untuk memverifikasi setiap kebutuhan fungsional, berikut ditunjukkan pada tabel 1.

Tabel 1. Matriks Keterlacakan

ID.	Test Case	Tujuan	Metode & Skenario	Metriks & Kriteria Lulus
FR-1	Deteksi real-time	Edge menghasilkan items[] per frame	Jalankan YOLOv8n @640, batch=1; 10 menit	≥ 1 event/dtk; log items tidak kosong
FR-2	POST ke backend	Payload diterima & tervalidasi	Kirim 600 event (1/dtk $\times$ 10 mnt)	$2xx \geq 99\%$; no schema error
FR-3	Persistensi DB	Data tersimpan benar	Cek tabel detections & detection_items	Baris = jumlah event; FK utuh
FR-4	SSE live	Dashboard terima event real-time	Buka /sse/stream; pantau 10 mnt	waktu terima $\leq$ NFR-1
FR-5	Analitik	View/endpoint agregasi benar	Bandingkan hitungan manual vs view	Selisih $\leq 1\%$
FR-6	Alert	Aturan sederhana memicu alert	Simulasikan kondisi (mis. residu berturut-turut)	Event alert muncul ≤ 1 dtk
FR-7	Device mgmt	upsert device_uid $\rightarrow$ device_id	Edge kirim device_uid baru	Device terdaftar otomatis
FR-8	Idempotensi	Duplikasi event tertolak	Kirim event_uuid sama 5 $\times$	1 insert; 4 conflict ditolak
FR-9	Livestream	Frame base64 tampil	3–5 FPS, 10 menit	Rata-rata 3–5 FPS stabil
FR-10	Konfigurasi	Toggle livestream on/off	GET /settings/livestream	FE & edge mengikuti toggle

Berdasarkan matriks keterlacakan yang ditunjukkan pada tabel 1 menampilkan pengujian fungsionalitas sistem menunjukkan kemampuan deteksi *real-time* yang stabil dengan *throughput* minimal 1 event per detik. Mekanisme POST ke *backend* menunjukkan tingkat keberhasilan di atas 99% dengan validasi schema yang konsisten. Validasi fungsionalitas sistem dilakukan melalui serangkaian *test case* yang komprehensif untuk memastikan setiap *requirement* terpenuhi.

b. Analisis Faktor Kinerja

Investigasi dilakukan terhadap beberapa faktor krusial yang mempengaruhi kinerja sistem deteksi objek *real-time*. Pertama, dampak resolusi input (antara 416 hingga 640 piksel) terhadap metrik mAP dan FPS dievaluasi, dimana resolusi yang lebih tinggi umumnya meningkatkan akurasi (mAP) namun berpotensi menurunkan *frame rate* (FPS) pada perangkat edge yang memiliki keterbatasan komputasi. Kedua, analisis efek dari *confidence threshold* dan *Non-Maximum Suppression* (NMS) terhadap *Recall curve* dilakukan untuk mengidentifikasi *trade-off* optimal dalam meminimalkan *false positives* sambil mempertahankan tingkat deteksi yang tinggi. Potensi kuantisasi model dan runtime inferensi juga didiskusikan untuk mencapai efisiensi komputasi maksimal pada perangkat edge.

3.4 Pembahasan

a. Kinerja Model dan Generalisasi

Hasil evaluasi menunjukkan bahwa model YOLOv8n mencapai akurasi yang sangat tinggi dengan mAP@0.5 sebesar 98.5%. performa ini mendemonstrasikan kemampuan model dalam menangani variasi kondisi pencahayaan dan latar belakang yang beragam. Tingginya *Recall* di atas 97% mengindikasikan bahwa model berhasil mempelajari fitur diskriminatif yang robust untuk membedakan kategori organik dan non-organik.

Konvergensi yang stabil selama proses pelatihan tanpa tanda-tanda *overfitting* menunjukkan bahwa arsitektur YOLOv8n cocok untuk dataset dan kompleksitas masalah yang dihadapi. Hal ini penting untuk memastikan bahwa model dapat melakukan generalisasi dengan baik pada data baru yang belum pernah dilihat sebelumnya.

b. Efisiensi Implementasi Edge computing

Implementasi pada Raspberry Pi 5 dengan Camera Module 3 menunjukkan keseimbangan yang baik antara *cost-effectiveness* dan performa operasional. Sistem mampu beroperasi secara kontinyu dengan mekanisme *offline queue* yang

reliable, menunjukkan tingkat keberhasilan *re-upload* 99.7% setelah gangguan koneksi. Fitur *Idempotency* menggunakan *event-uuid* berhasil mencegah duplikasi data dengan tingkat keberhasilan 100%.

Throughput sistem yang stabil dengan latensi komunikasi rata-rata 0.5-0.6 ms untuk *SSE broadcast* memungkinkan monitoring operasional yang responsive. Hal ini critical untuk aplikasi *real-time* dimana delay yang berlebihan bisa mengurangi efektivitas sistem pemilahan.

c. Integrasi Sistem Informasi Real-time

Implementasi *Servers-Sent Events (SSE)* terbukti efektif untuk *broadcasting real-time* updates dengan overhead minimal. Dibandingkan dengan alternatif seperti *WebSocket*, *SSE* memberikan *simplicity* dan *reliability* yang lebih baik untuk *use case unidirectional updates*. *Backend API* menunjukkan performa yang konsisten dalam menangani *concurrent requests* dan menyiarkan event ke *multiple clients*.

3.5 Keterbatasan dan Tantangan

a. Implementasi

Meskipun sistem menunjukkan kinerja yang menjanjikan, terdapat beberapa keterbatasan yang perlu diakui. Generalisasi lintas scene, terutama pada kondisi pencahayaan ekstrem atau lingkungan yang sangat kompleks, mungkin masih menjadi tantangan. Keterbatasan perangkat edge, seperti kapasitas pemrosesan dan memory Raspberry Pi 5, membatasi ukuran model dan kompleksitas inferensi yang dapat dijalankan secara efisien.

b. Implikasi untuk Implementasi Operasional

Hasil penelitian menunjukkan *feasibility* implementasi sistem AI untuk *waste sorting* dalam setting operasional nyata. Arsitektur *end-to-end* yang telah dibangun memberikan foundation yang solid untuk deployment skala yang lebih besar, mekanisme ketahanan sistem seperti *offline queue* dan *error handling* menunjukkan kematangan desain yang diperlukan untuk lingkungan produksi. Integrasi dengan *existing workflow* dapat dilakukan secara incremental, memungkinkan adoption yang gradual dengan minimal disruption terhadap operasi existing. *Dashboard analytics* memberikan visibility yang diperlukan untuk continuous monitoring dan optimization sistem pemilahan sampah otomatis.

4. KESIMPULAN

Penelitian berhasil menyajikan sistem terintegrasi yang inovatif untuk pemilahan sampah otomatis berbasis Computer Vision yang menggabungkan model deteksi objek YOLOv8 yang berjalan diperangkat *edge* Raspberry Pi 5 dilengkapi dengan Camera Module 3, dengan sebuah Sistem Informasi *real-time* ditenagai oleh *Server-Sent Events (SSE)*. Prototipe ini menunjukkan kinerja yang menjanjikan, tidak hanya dari segi akurasi deteksi dengan nilai mAP@0.5 mencapai 98.5%, tetapi juga dalam hal efisiensi operasional dengan latensi median sekitar 0.5-0.6 ms untuk pengiriman data. Sistem juga menyediakan kapabilitas observabilitas dan analitik operasional yang komprehensif, sehingga pemantauan kinerja dan pengambilan keputusan berbasis data. Meskipun hasil yang dicapai positif, terdapat beberapa arah pengembangan yang menarik untuk penelitian selanjutnya. Pengembangan dapat meliputi perluasan sistem untuk mendukung multi-kamera yang memungkinkan cakupan area yang lebih luas dan perspektif deteksi yang lebih kaya. Pengembangan mengintegrasikan aktuator atau *robotic arm* secara langsung ke dalam sistem, mengubah prototipe ini menjadi solusi otomatis penuh yang tidak hanya mendeteksi tetapi juga secara fisik memilah sampah. Selain itu, perluasan kategori sampah yang dapat di deteksi akan menjadi fokus untuk meningkatkan fleksibilitas dan relevansi sistem di berbagai skenario aplikasi. Investigasi pada strategi *edge-cloud offloading* dapat dilakukan untuk mengoptimalkan pembagian beban komputasi antara perangkat *edge* dan layanan *cloud*, guna meningkatkan skalabilitas dan resiliensi sistem secara

REFERENCES

- [1] H. Saidah, I. W. Yasa, L. W. Wiradharma, A. Suroso, and A. Supriyadi, "Upaya Mendorong Pencapaian SDGs-12 melalui Sosialisasi Pemanfaatan Sampah Organik menjadi Eco Enzyme pada Masyarakat Desa Jelantik," 2024.
- [2] D. Noer, A. Rahmanto, and A. A. Yogyakarta, "Pengelolaan Sampah Berkelanjutan melalui Kolaborasi Komunitas dan Inovasi Teknologi : Studi Kasus di Dusun," vol. 1, no. 1, pp. 1–7, 2024.
- [3] P. Sampah and B. Masyarakat, "Jurnal pacta sunt servanda," vol. 3, no. September, 2022.
- [4] S. M. Sunarsiningsih, J. Mardimin, U. Kristen, and S. Wacana, "Rendahnya Partisipasi Masyarakat Kota Salatiga dalam Penanganan dan Pengelolaan Sampah Plastik," *KRITIS*, vol. 33, no. 2, pp. 174–194, Dec. 2024, doi: 10.24246/KRITIS.V33I2P174-194.
- [5] N. S. Putri, N. S. Febriyanti, and S. A. Noor, "Naila Sayyidina Putri 1 , Nasab Sabrina Febriyanti 2 , Salsabila Azahra Noor 3," vol. 3, no. 4, pp. 635–648, 2025.
- [6] B. S. Rakhimov, F. B. Rakhimova, S. K. Sobirova, F. O. Kuryazov, and D. B. Abdirimova, "Review And Analysis Of Computer Vision Algorithms," *Am. J. Appl. Sci.*, vol. 03, no. 05, pp. 245–250, May 2021, doi: 10.37547/tajas/Volume03Issue05-39.
- [7] D. Vieri, R. Susanto, E. S. Purwanto, and M. K. Ario, "Enhancing Waste Classification with YOLOv8 Models for Efficient and Accurate Sorting," *Procedia Comput. Sci.*, vol. 245, pp. 889–895, 2024, doi: 10.1016/j.procs.2024.10.316.
- [8] R. T. Hutabarat and R. Kurniawan, "Deteksi Sampah di Permukaan Sungai menggunakan Convolutional Neural Network dengan Algoritma YOLOv8," *Semin. Nas. Off. Stat.*, vol. 2024, no. 1, pp. 537–548, Nov. 2024, doi: 10.34123/semnasoffstat.v2024i1.2099.
- [9] S. Kunwar and A. S. Alade, "Deep Learning in Waste Management: A Brief Survey." Jul. 09, 2024. doi: 10.20944/preprints202407.0637.v1.

- [10] M. A. A. Muzammil and R. Indraswari, "Pengembangan Arsitektur Model YOLOv8 untuk Meningkatkan Performa Object Detection pada Varian Boks Warehouse Palletizing," *Ilk. J. Comput. Sci. Appl. Informatics*, vol. 6, no. 2, pp. 19–30, Aug. 2024, doi: 10.28926/ilkomnika.v6i2.642.
- [11] M. R. Ridha, S. Syafrijon, Y. Hendriyani, and A. Hadi, "Implementasi Model YOLOv8 untuk Deteksi Jenis Sampah Organik dan Anorganik Berbasis Android," *Abdimas Indones. J.*, vol. 5, no. 1, pp. 419–426, Apr. 2025, doi: 10.59525/aij.v5i1.655.
- [12] A. Nayfeh, S. Al-Azani, and H. Samma, "A Two-Stage YOLOv8 Approach for Waste Detection and Classification in Cognitive Cities," *Transp. Res. Procedia*, vol. 84, pp. 579–586, 2025, doi: 10.1016/j.trpro.2025.03.111.
- [13] M. T. Okano, W. A. C. Lopes, S. M. Ruggero, O. Vendrametto, and J. C. L. Fernandes, "Edge AI for Industrial Visual Inspection: YOLOv8-Based Visual Conformity Detection Using Raspberry Pi," *Algorithms*, vol. 18, no. 8, p. 510, Aug. 2025, doi: 10.3390/a18080510.
- [14] R. Mysiuk, "ANALYSIS OF EFFECTIVE IMAGE PROCESSING METRICS ON RASPBERRY PI AND NVIDIA JETSON NANO," *Electron. Inf. Technol.*, vol. 28, p. 774, 2024, doi: 10.30970/eli.28.2.
- [15] R. Singh and S. S. Gill, "Edge AI: A survey," *Internet Things Cyber-Physical Syst.*, vol. 3, pp. 71–92, 2023, doi: 10.1016/j.iotcps.2023.02.004.
- [16] Zamathula Queen Sikhakhane Nwokediegwu, Ejike David Ugwuanyi, Michael Ayorinde Dada, Michael Tega Majemite, and Alexander Obaigbena, "AI-DRIVEN WASTE MANAGEMENT SYSTEMS: A COMPARATIVE REVIEW OF INNOVATIONS IN THE USA AND AFRICA," *Eng. Sci. Technol. J.*, vol. 5, no. 2, pp. 507–516, Feb. 2024, doi: 10.51594/estj.v5i2.828.
- [17] D. Tribuana, U. Usman, and D. Dayanti, "Deteksi Sampah Otomatis Pada Lingkungan Terbuka Menggunakan YOLOV8 Dan Dataset Roboflow," *J. Teknol. dan Bisnis Cerdas*, vol. 1, no. 1, pp. 38–49, Jul. 2025, doi: 10.64476/jtbc.v1i1.4.
- [18] R. Sesiati, M. A. Febrian, and F. Firdaus, "INTEGRASI SAINS DATA DAN REKAYASA PERANGKAT LUNAK: PENDEKATAN HOLISTIK DALAM PENGEMBANGAN APLIKASI PINTAR," *SISKOMTI J. Sist. Inf. Komput. dan Teknol. Inf.*, vol. 7, no. 1, Apr. 2025, doi: 10.54342/7ept9v90.
- [19] Oluwole Temidayo Modupe *et al.*, "REVIEWING THE TRANSFORMATIONAL IMPACT OF EDGE COMPUTING ON REAL-TIME DATA PROCESSING AND ANALYTICS," *Comput. Sci. IT Res. J.*, vol. 5, no. 3, pp. 693–702, Mar. 2024, doi: 10.51594/csitrj.v5i3.929.
- [20] Syarif Husin, Iskandar Lutfi, and Masayu Anisah, "PENGUJIAN PENGENALAN WAJAH REAL-TIME DENGAN DLIB PADA RASPBERRY PI 5," *J. Ris. Multidisiplin Edukasi*, vol. 2, no. 6, pp. 254–269, Jun. 2025, doi: 10.71282/jurmie.v2i6.442.