

Real-Time Garbage Detection and Classification with YOLO Architecture

Tim Mahasiswa:

Alan Nabi - 2702375821

Johannes Jourdan Triadi - 2702348272

Nathaniel Wijaya - 2702286646

I. Introduction

1.1. Latar Belakang:

Pertumbuhan populasi dan pola konsumsi modern menyebabkan peningkatan volume sampah secara signifikan. Proses pemilahan sampah secara manual masih banyak digunakan, namun metode ini tidak efisien, memakan waktu, rentan kesalahan, serta berisiko terhadap kesehatan pekerja. Kesalahan klasifikasi sampah juga berdampak pada rendahnya kualitas daur ulang akibat kontaminasi silang antar material.

Perkembangan teknologi Computer Vision dan Deep Learning membuka peluang untuk mengotomatisasi proses deteksi dan klasifikasi sampah secara real-time. Salah satu arsitektur yang banyak digunakan untuk deteksi objek adalah YOLO (You Only Look Once), yang dikenal memiliki kecepatan dan akurasi tinggi[1].

Seiring dengan pertumbuhan populasi di seluruh dunia, volume sampah terus meningkat. Ini membuat pengelolaan limbah dan upaya daur ulang sangat sulit. Pemilahan sampah secara manual memakan banyak waktu, tidak efisien, dan rentan terhadap kesalahan klasifikasi karena bergantung pada kemampuan dan konsistensi karyawan. Ini terutama berlaku untuk sampah dengan jenis dan volume yang sangat besar.

Pemilahan sampah secara manual sangat bergantung pada ketelitian pekerja dan pengetahuan tentang jenis material yang dapat didaur ulang. Prosesnya lambat dan berisiko tinggi terhadap kontaminasi silang, yang mengurangi kualitas material yang akan diproses ulang. Selain itu, pekerjaan pemilahan manual menempatkan pekerja dalam risiko kesehatan dan keselamatan karena mereka harus berinteraksi langsung dengan limbah berbahaya. Oleh karena itu, diperlukan solusi inventif yang dapat melakukan klasifikasi sampah secara otomatis, cepat, akurat, dan konsisten tanpa bergantung pada tenaga manusia.

Kita dapat memanfaatkan kemampuan sistem untuk mengklasifikasikan sampah secara otomatis dengan tingkat akurasi tinggi berkat teknologi AI, khususnya computer vision dan deep learning. Sistem klasifikasi sampah yang menggunakan kecerdasan buatan memiliki kemampuan untuk secara akurat mengklasifikasikan sampah berdasarkan material seperti kertas, plastik, logam, kaca, organik, dll., serta mengidentifikasi jenis sampah secara real-time dari gambar atau video. Karena sistem AI dapat secara otomatis mengenali dan memilah berbagai jenis sampah tanpa memerlukan intervensi manual lagi, hal ini dapat meningkatkan efisiensi operasional fasilitas daur ulang, tempat pembuangan akhir (TPA), dan sistem pengelolaan sampah[2]. Dengan teknologi ini, operator fasilitas daur ulang dapat berkonsentrasi pada pengawasan dan optimalisasi proses. Di sisi lain, sistem AI menangani tugas klasifikasi yang repetitif dan membutuhkan banyak waktu.

1.2. Tujuan:

Computer Vision saat ini dapat mendeteksi objek umum, tetapi seringkali tidak dapat membedakan kategori sampah dengan tepat, terutama untuk sampah dengan berbagai bentuk, warna, atau kondisi[3]. Proyek ini bertujuan untuk mengatasi masalah tersebut dengan mengembangkan Garbage Object Detection & Classification System yang mengintegrasikan state-of-the-art object detection model (YOLOv8n) dengan dataset yang mencakup berbagai kategori sampah rumah tangga. Sistem ini diintegrasikan dengan IoT memungkinkan fasilitas pengelolaan limbah, smart bins, dan aplikasi waste management untuk mengklasifikasikan sampah secara otomatis dan akurat[1], sehingga meningkatkan tingkat daur ulang, mengurangi kontaminasi, dan mendukung upaya keberlanjutan lingkungan.

1.3. Ruang Lingkup:

Proyek ini mencakup tahapan persiapan dataset, pelatihan model deteksi objek berbasis YOLOv8, evaluasi performa model, serta implementasi model untuk kebutuhan inferensi.

II. Dataset

Dataset yang digunakan adalah Garbage Classification v2[4], yang berisi kumpulan citra sampah dari berbagai kategori material seperti plastik, kertas, kaca, logam, dan sampah organik.

2.1. Data Explanation:

Dataset ini berisi gambar-gambar sampah rumah tangga pada umumnya yang sesuai dengan target model.

Sumber: <https://www.kaggle.com/datasets/sumn2u/garbage-classification-v2>

2.2. Eksplorasi:

Pada tahap awal, dilakukan pemeriksaan struktur dataset dan distribusi kelas. Dataset kemudian disesuaikan ke dalam format YOLO yang terdiri dari folder *train* dan *validation* dengan pasangan citra dan label. Insight awal menunjukkan adanya variasi bentuk, warna, dan kondisi objek sampah yang menambah kompleksitas proses deteksi.

III. Modeling

3.1. Arsitektur Model:

Model yang digunakan adalah YOLOv8, yang merupakan arsitektur *single-stage object detection* berbasis *Convolutional Neural Network (CNN)*. YOLOv8 mampu melakukan deteksi bounding box dan klasifikasi kelas secara bersamaan dalam satu proses inferensi.

YOLOv8 dipilih karena:

- Memiliki kecepatan inferensi tinggi dan cocok untuk aplikasi real-time
- Akurat dalam mendeteksi objek dengan berbagai skala
- Mudah diimplementasikan menggunakan library Ultralytics

Versi yang digunakan adalah **YOLOv8n (Nano)** karena ringan dan efisien untuk eksperimen awal.

3.2. Skenario Pelatihan:

Model dilatih menggunakan pretrained weights (transfer learning) dari yolov8n.pt. Konfigurasi pelatihan meliputi:

- Epoch: 50

- Image size: 640×640
- Optimizer: Default Ultralytics (SGD/AdamW)

3.3. Diagram Arsitektur:

Arsitektur YOLOv8 terdiri dari *backbone*, *neck*, dan *head* yang berfungsi untuk ekstraksi fitur, fusi fitur multi-skala, serta prediksi bounding box dan kelas objek.

IV. Evaluation (Evaluasi)

4.1. Metrik Evaluasi:

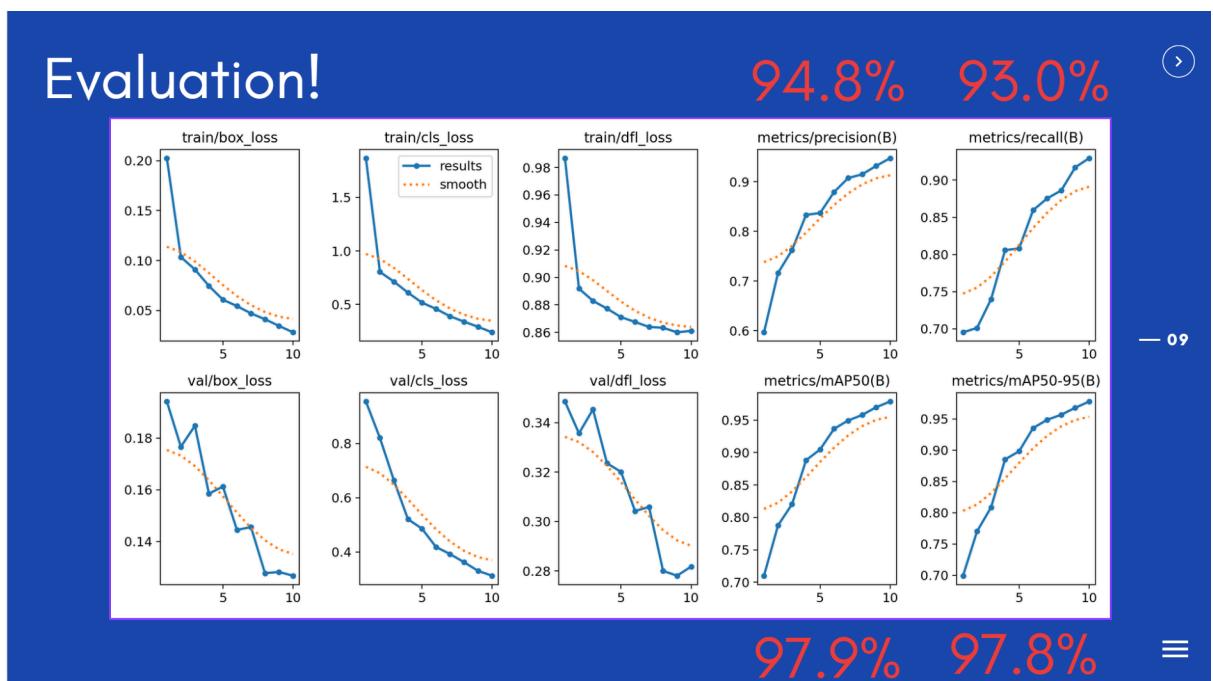
Evaluasi model dilakukan menggunakan metrik standar deteksi objek, yaitu:

- Precision
- Recall
- mAP@0.5
- mAP@0.5:0.95
- Confusion Matrix

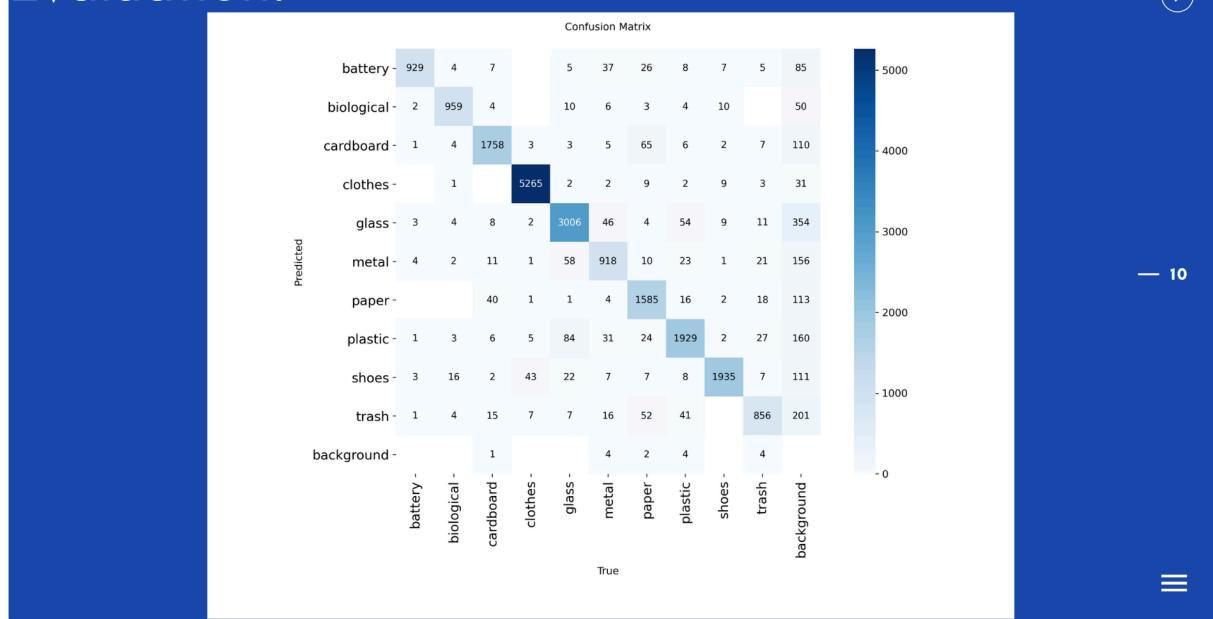
4.2. Comparison (Perbandingan):

Evaluasi dilakukan berdasarkan hasil validasi selama proses training. Model dasar YOLOv8n digunakan sebagai baseline tanpa augmentasi lanjutan.

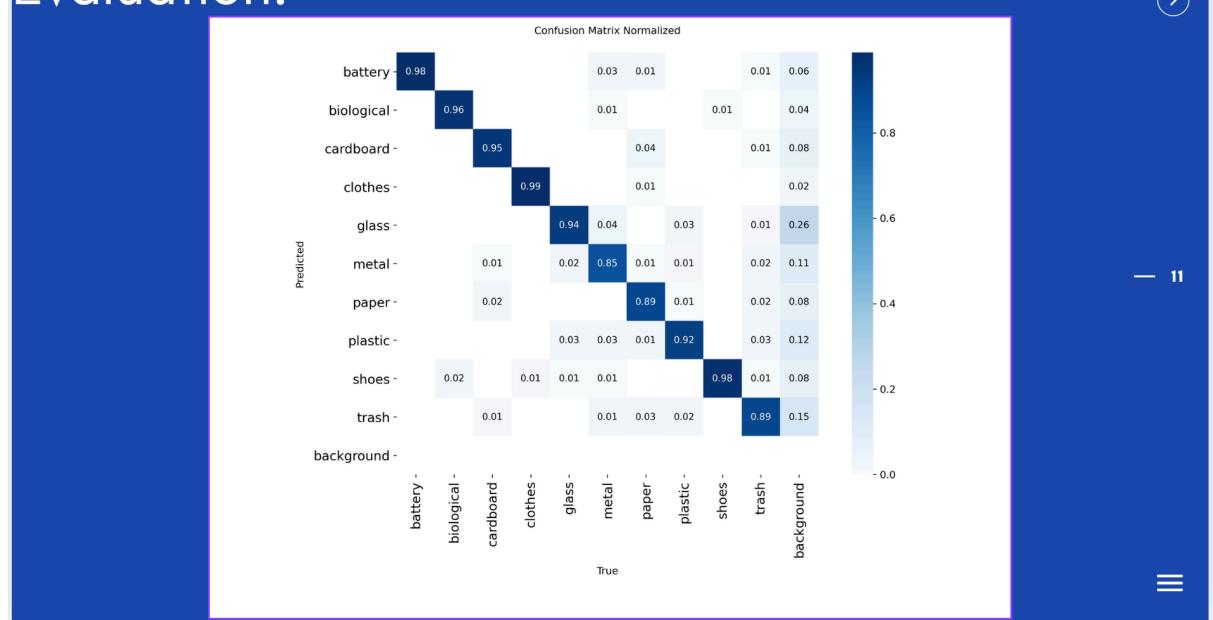
4.3. Visualisasi Hasil:



Evaluation!

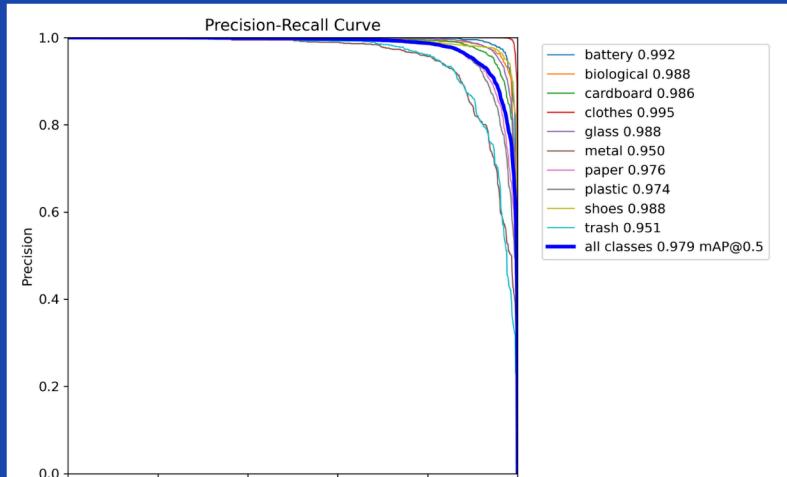


Evaluation!



Evaluation!

12



Evaluation!

13



V. Deployment (Implementasi)

5.1. Aplikasi:

Untuk tahap implementasi, kami menggunakan **Streamlit**, sebuah *framework* berbasis Python yang memungkinkan pembuatan antarmuka web interaktif secara cepat untuk proyek Machine Learning. Streamlit dipilih karena kemudahannya dalam

mengintegrasikan library Computer Vision seperti OpenCV dan PyTorch tanpa perlu membangun *backend* yang kompleks dari awal.

5.2. Alur Kerja:

Aplikasi dirancang untuk memberikan pengalaman inferensi yang intuitif dengan alur sebagai berikut:

1. **Input Pengguna:** Pengguna dapat memberikan input melalui dua cara: mengunggah gambar/video statis atau menggunakan akses kamera (*webcam*) secara langsung.
2. **Pemuatan Model:** Aplikasi secara otomatis memuat model YOLOv8n (file .pt) yang telah dilatih menggunakan GPU (CUDA) untuk memastikan proses deteksi berjalan dengan latensi rendah.
3. **Preprocessing:** Gambar input disesuaikan ukurannya (resize) ke resolusi 640x640 agar sesuai dengan format input model.
4. **Inferetion:** Model melakukan pemindaian objek pada setiap frame. Algoritma YOLO mendekripsi koordinat *bounding box*, mengklasifikasikan jenis sampah, dan memberikan skor keyakinan (*confidence score*).
5. **Output:** Hasil inferensi ditampilkan kembali ke antarmuka Streamlit dalam bentuk gambar/video yang telah dilengkapi dengan kotak pembatas (*bounding box*) dan label kategori sampah (misalnya: *Biological, Plastic, Metal*).

VI. Reflection (Refleksi)

6.1. Analisis Kritis:

Berdasarkan eksperimen yang dilakukan, kita menganalisis performa model melalui beberapa skenario pelatihan:

1. **Analisis Perbandingan Skenario (YOLOv8 vs YOLOv12):** Secara teoretis, YOLOv12 menawarkan arsitektur *attention* yang lebih canggih. Namun, secara praktis, hasil analisis kita menunjukkan bahwa **YOLOv12 kurang worth it** untuk skenario pelatihan ini. Perilaku model YOLOv12 menunjukkan lonjakan waktu pelatihan (*training time*) yang sangat signifikan tanpa peningkatan akurasi (*mAP*) yang proporsional dibandingkan YOLOv8. Dalam skenario *real-time*, YOLOv8 terbukti lebih stabil dan efisien dalam penggunaan sumber daya.
2. **Analisis Pengaruh Dataset :** kita telah menguji perilaku model pada **7 dataset berbeda**. Hasil analisis menunjukkan adanya *performance gap* yang besar antara dataset mentah (klasifikasi) dan dataset ter-anotasi (*annotated*). Model yang dilatih dengan data ber-anotasi (seperti YOLO format) menunjukkan bounding box yang jauh lebih presisi, namun memiliki keterbatasan pada spesifitas label (seperti tipe sampah pada TACO yang terlalu lemah), dan juga kualitas dataset yang kurang memadai.

6.2 Kendala & Pembelajaran:

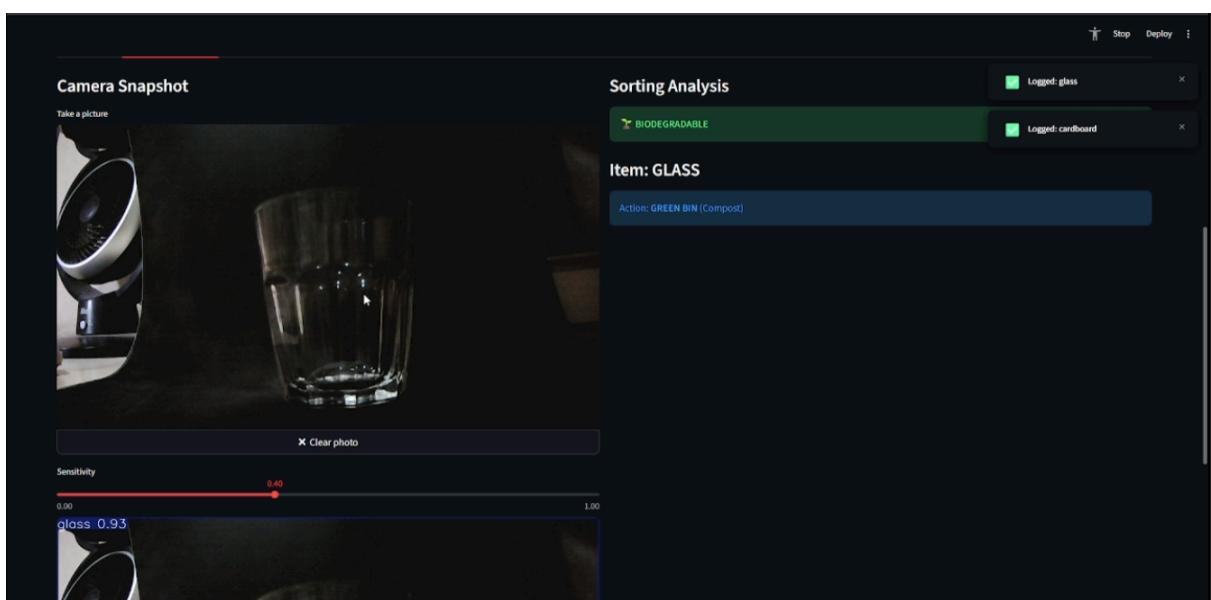
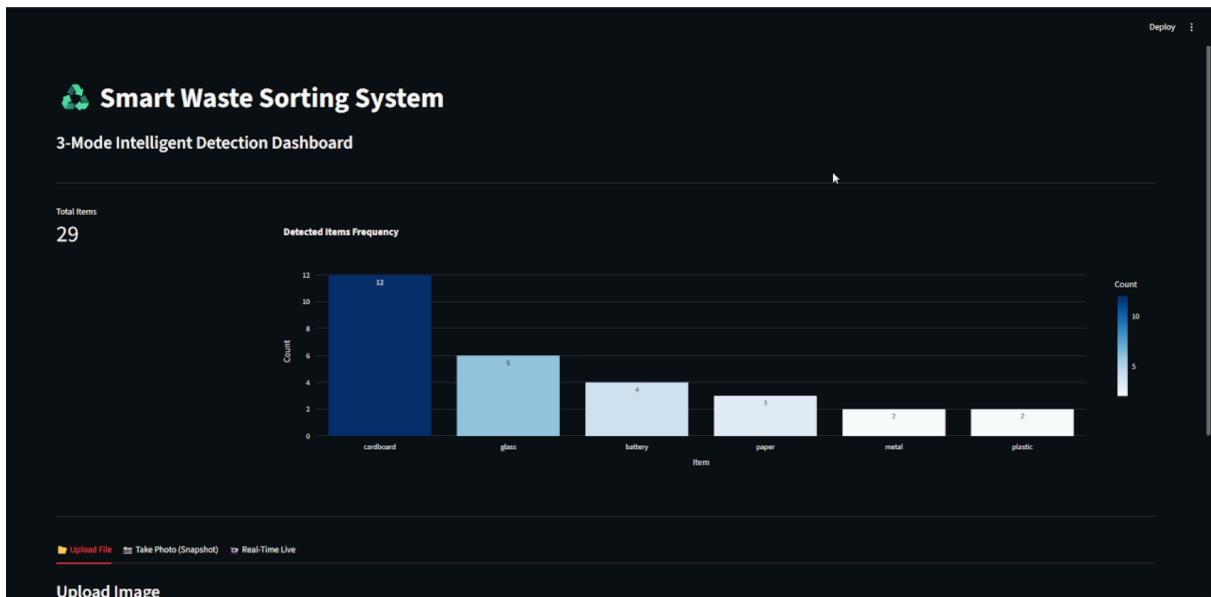
Hambatan teknis utama adalah menemukan titik temu antara akurasi model dan kecepatan inferensi. kita mempelajari bahwa dataset yang baik bukan hanya soal jumlah gambar, melainkan kualitas anotasi bounding box yang konsisten. Jika anotasi tidak konsisten, model akan menunjukkan perilaku loss yang fluktuatif dan sulit mencapai konvergensi. jadi makanya di model terakhir, kami menggunakan fake bounding box untuk yolov8n, karena dataset kita merupakan dataset yang tidak ter annotated.

VII. References

- [1] W. A. Syafei, G. E. Patricia, E. Handoyo, R. Charistyandi, M. H. B. Pasaribu, and O. D. Nurhayati, “IoT Based Smart Trash Bank for Organic and Inorganic Trash Separation,” in *2024 International Conference on Computer Engineering, Network, and Intelligent Multimedia (CENIM)*, IEEE, Nov. 2024, pp. 1–6. doi: 10.1109/CENIM64038.2024.10882807.
- [2] S. Selvi, M. Suhash, R. Lingeswaran, and N. Soundar, “Design of Smart Trash Can for Waste Management Using MobileNet Model,” in *2025 IEEE 6th International Conference in Robotics and Manufacturing Automation (ROMA)*, IEEE, Aug. 2025, pp. 52–56. doi: 10.1109/ROMA66616.2025.11155737.
- [3] D. Wang and Z. Zhang, “A Comparison Study of Image Data Augmentation Techniques in Computer Vision,” in *2025 5th International Conference on Computer Vision, Application and Algorithm (CVAA)*, IEEE, Aug. 2025, pp. 410–414. doi: 10.1109/CVAA66438.2025.11193215.
- [4] Suman Kunwar. (2025). Garbage Dataset [Data set]. Kaggle. <https://doi.org/10.34740/KAGGLE/DSV/14118792>

VIII. LAMPIRAN

Screenshot Aplikasi



Link Repository

<https://github.com/n-Alan777/Smart-Waste-Sorter-YOLOv8>

Link Deploy

<https://drive.google.com/drive/folders/16RdhTTW-pZDazk8pDsBh02Wa8l9-Y5Yw?usp=sharing>

Link Demo Video

<https://drive.google.com/drive/folders/16RdhTTW-pZDazk8pDsBh02Wa8l9-Y5Yw?usp=sharing>

Link Video Presentation

<https://drive.google.com/drive/folders/16RdhTTW-pZDazk8pDsBh02Wa8l9-Y5Yw?usp=sharing>

Link Power Point

https://www.canva.com/design/DAG7fndLhiY/9sSsbAWzEoRUzzESKtXzeA/edit?utm_content=DAG7fndLhiY&utm_campaign=designshare&utm_medium=link2&utm_source=sharebutton

Contribution statement:

Alan Nabil - 2702375821	Membuat Model Trash Detection, Membuat Frontend & Backend app, Membuat & Pengurus Repositories Github, Membuat Laporan, Membuat Powerpoint Presentation, Membuat Demo Video,Membuat Video Presentasi
Johannes Jourdan Triadi - 2702348272	Membuat Model Trash Detection, Membuat Laporan, Membuat Powerpoint Presentation, Membuat Video Presentasi
Nathaniel Wijaya - 2702286646	Membuat Model Trash Detection, Membuat Laporan, Membuat Powerpoint Presentation, Membuat Video Presentasi