



Department of Digital Business

Journal of Artificial Intelligence and Digital Business (RIGGS)

Homepage: <https://journal.ilmudata.co.id/index.php/RIGGS>

Vol. 5 No. 2 (2026) pp: 15175-15181

P-ISSN: 2963-9298, e-ISSN: 2963-914X

Pengembangan Aplikasi Mobile Klasifikasi Object Detection Sampah Anorganik Menggunakan Arsitektur SSD MobilenetV2

Ngakan Gde Satria Abirama¹, Ketut Agus Seputra², I Nyoman Saputra Wahyu Wijaya³

¹⁻³Program Studi Ilmu Komputer, Fakultas Teknik dan Kejuruan, Universitas Pendidikan Ganesha
ramaabi470@gmail.com

Abstrak

Pengelolaan sampah anorganik yang masih dilakukan secara manual menyebabkan proses pemilahan menjadi kurang efisien dan berpotensi meningkatkan kesalahan identifikasi jenis sampah serta Kondisi tersebut dapat menghambat upaya pengelolaan dan daur ulang sampah yang efektif. Penelitian ini bertujuan mengembangkan aplikasi mobile dengan menanamkan model SSD-MobileNetV2 yang mampu mendeteksi sampah plastik dan kaca secara real-time untuk membantu proses identifikasi dan pemilahan sampah plastik dan kaca secara lebih cepat dan akurat.. Penelitian menggunakan metode Research and Development (R&D) yang meliputi tahapan pengumpulan dan pengolahan data, pengembangan model, pengembangan aplikasi, serta pengujian aplikasi. Model dilatih menggunakan pendekatan transfer learning dengan partial fine-tuning pada 10 layer terakhir backbone MobileNetV2 dan diimplementasikan ke dalam aplikasi Android menggunakan TensorFlow Lite sehingga dapat dijalankan secara optimal pada perangkat Android tanpa memerlukan koneksi internet. Berdasarkan hasil evaluasi, model memperoleh akurasi sebesar 89,90%, presisi 90,00%, recall 90,00%, F1-score 90,00%, dan mean Average Precision (mAP) sebesar 95,88%. Hasil pengujian Black Box Testing menunjukkan seluruh fitur aplikasi berjalan sesuai dengan kebutuhan sistem, sedangkan pengujian System Usability Scale (SUS) menunjukkan aplikasi mudah dipahami dan digunakan oleh pengguna. Hasil penelitian menunjukkan bahwa aplikasi yang dikembangkan mampu mendeteksi sampah plastik dan kaca secara cepat dan akurat serta berpotensi menjadi solusi pendukung pemilahan sampah anorganik berbasis perangkat mobile untuk meningkatkan efektivitas pengelolaan sampah.

Kata Kunci: SSD-Mobilenetv2, Deteksi Sampah, Tensorflow Lite, Android, Transfer Learning.

1. Latar Belakang

Permasalahan sampah anorganik merupakan salah satu tantangan dalam mewujudkan lingkungan berkelanjutan hingga sekarang. Dua jenis sampah anorganik, plastik dan kaca yang sulit terurai dan berpotensi menimbulkan pencemaran jika tidak diolah dengan baik melalui proses pemilahan yang baik [1]. Selain itu, proses pengolahan sampah di Indonesia sebagian besar dilakukan secara konvensional, dimana sampah dikumpulkan dan dibawa ke Tempat Pembuangan Akhir. Masih sedikit tempat yang melakukan pemilihan sampah untuk tujuan tertentu. Permasalahan pemilahan sampah semakin kompleks karena masih dilakukan secara manual yang perlu waktu, tenaga dalam melakukan identifikasi. Di sisi lain, perkembangan teknologi khususnya perangkat lunak dan kecerdasan buatan memberikan peluang dalam pengembangan sistem klasifikasi sampah kaca dan plastik dengan cepat, praktis dan mudah digunakan. Dengan memanfaatkan teknik deteksi objek memungkinkan proses identifikasi dilakukan secara otomatis dan *real-time* [2].

Penelitian mengenai klasifikasi sampah dengan memanfaatkan *computer vision* menunjukkan peningkatan perkembangan yang tinggi dalam beberapa tahun ini. Berbagai penelitian memanfaatkan arsitektur *Convolutional Neural Network* (CNN) khususnya MobileNetV2. Arsitektur tersebut banyak dipilih karena dengan kompleksitas komputasi yang rendah dapat menghasilkan performa yang baik terutama pada perangkat dengan sumberdaya terbatas [3]. Selain itu, pada penelitian Pritama dkk (2024) menunjukkan penerapan *transfer learning* dengan arsitektur MobileNetV2 mampu menghasilkan akurasi yang baik untuk klasifikasi sampah organik dan anorganik tanpa meningkatkan kompleksitas model secara signifikan [4]. Implementasi TensorFlow Lite juga memungkinkan model dapat diintegrasikan pada perangkat Android, sehingga dapat digunakan secara *offline* dan waktu respon yang cepat [5]. Pada kajian terbaru juga menunjukan arsitektur dari MobileNet menjadi salah satu yang paling banyak digunakan untuk implementasi model pada aplikasi karena menawarkan keseimbangan akurasi, komputasi dan konsumsi memori yang baik [6].

Walaupun berbagai penelitian telah menunjukkan performa MobileNetV2 yang baik, pada penelitian terdahulu masih terdapat beberapa keterbatasan. Sebagian besar penelitian hanya berfokus pada peningkatan performa model klasifikasi atau deteksi objek tanpa mengintegrasikan model kedalam aplikasi atau web yang dapat digunakan [7], [8]. Beberapa penelitian yang telah mengembangkan aplikasi Android hanya menyediakan fungsi mendeteksi tanpa adanya fitur menghitung jumlah objek hasil deteksi ataupun penyimpanan riwayat deteksi [7], [9]. Selain itu, sebagian besar penelitian menggunakan banyak kategori sampah sehingga belum memberikan perhatian khusus pada klasifikasi sampah anorganik yang banyak dijumpai dalam aktivitas sehari-hari, yaitu sampah plastik dan kaca. Hal ini menunjukkan masih adanya kesenjangan penelitian dalam pengembangan aplikasi model yang mampu mengintegrasikan model deteksi objek secara *offline* dan menyediakan pencatatan secara *real-time* [10], [11].

Dengan kesenjangan yang tersebut, penelitian ini menawarkan pengembangan aplikasi mobile berbasis Android. Melakukan integrasi SSD-MobileNetV2 dalam format TensorFlow Lite untuk melakukan klasifikasi sampah plastik dan kaca secara *real-time*. Tidak hanya melakukan deteksi objek secara umum seperti penelitian sebelumnya, penelitian ini mengembangkan aplikasi yang tidak hanya dapat melakukan klasifikasi sampah plastik atau kaca saja, namun juga menghitung jumlah objek yang terdeteksi secara otomatis serta menyimpan riwayat deteksi tanpa tergantung pada koneksi internet. Dengan memanfaatkan SQLite yang dapat menyimpan data secara lokal pada perangkat [12], [13]. Pemanfaatan SSD-MobileNetV2 dipilih karena memiliki karakteristik ringan dan sesuai dengan implementasi mobile dengan keterbatasan sumber daya komputasi [4]. SSD-MobileNetV2 juga dapat menghasilkan model dengan ukuran yang cukup kecil serta mendukung format TensorFlow Lite. Pengembangan aplikasi berbasis Android juga dinilai mampu meningkatkan aksesibilitas, kemudahan penggunaan, serta mendukung implementasi solusi digital yang dapat dimanfaatkan secara langsung oleh pengguna di lapangan. Berbagai penelitian menunjukkan bahwa aplikasi mobile dapat menjadi media yang efektif untuk mengintegrasikan proses komputasi dengan antarmuka yang mudah digunakan serta mendukung pengujian fungsionalitas dan usability sistem [14], [15].

Oleh karena itu, penelitian ini bertujuan mengintegrasikan model SSD-MobileNetV2 berbasis TensorFlow Lite dengan mekanisme penghitungan jumlah objek secara otomatis serta penyimpanan riwayat deteksi dalam satu aplikasi Android yang mampu bekerja secara *offline*. Selain mengevaluasi performa model deteksi, penelitian ini juga menguji aspek fungsionalitas dan *usability* aplikasi sehingga diharapkan mampu menghasilkan solusi yang akurat, efisien, mudah digunakan, dan dapat mendukung proses pemilahan sampah anorganik secara lebih efektif.

2. Metode Penelitian

Metode penelitian yang digunakan yaitu metode *Research and Development* dengan pendekatan kuantitatif untuk mengembangkan aplikasi *mobile*. Tujuan menggunakan Metode ini yaitu memungkinkan pengembangan dan penyempurnaan produk berbasis teknologi melalui serangkaian tahapan yang sistematis dan terstruktur [16]. Proses penelitian dilakukan dalam beberapa tahapan, yaitu pengumpulan dan pengolahan data, pengembangan model, pengembangan aplikasi, dan pengujian aplikasi.

2.1. Pengumpulan dan Pengolahan Data

Pada awal penelitian, diawali dengan pengumpulan dataset citra sampah yang terdiri dari dua kategori, yaitu sampah plastik dan sampah kaca. Pengumpulan dilakukan dengan metode primer dan sekunder, dimana data diperoleh dari dataset public pada Kaggle serta melakukan pengumpulan secara mandiri menggunakan smartphone. Dengan menggunakan dua metode ini dapat meningkatkan variasi objek, sudut pengambilan, pencahayaan, serta latar dari data citra. Setelah selesai melakukan pengumpulan data, dilakukan pembagian data menjadi 3 bagian, yaitu pelatihan, validasi dan pengujian 80:10:10. Khusus pada data pelatihan dilakukan augmentasi data dengan tujuan untuk memperbanyak data serta meningkatkan variasi data yang digunakan pada penelitian.

2.2. Pengembangan Model

Model deteksi objek dikembangkan menggunakan arsitektur SSD-MobileNetV2 dengan framework TensorFlow. Tahap pengembangan meliputi proses pelabelan dataset, pelatihan model, dan evaluasi performa model. Dalam proses evaluasi model, digunakan akurasi, presisi, *recall* dan *F1-Score*. Setelah model selesai dilakukan pengujian dan diperoleh model terbaik, selanjutnya dilakukan konversi format ke TensorFlow Lite.

2.3. Pengembangan Aplikasi

Aplikasi dikembangkan menggunakan Android Studio dengan bahasa pemrograman Kotlin dan menerapkan arsitektur Model–View–ViewModel (MVVM). Model TensorFlow Lite diintegrasikan ke dalam aplikasi sehingga

proses inferensi dapat dilakukan secara lokal (on-device) tanpa memerlukan koneksi internet. Aplikasi menyediakan beberapa fungsi utama, yaitu mendeteksi objek sampah secara real-time melalui kamera, mengklasifikasikan jenis sampah menjadi plastik atau kaca, menghitung jumlah objek yang terdeteksi, serta menyimpan riwayat hasil deteksi ke dalam basis data SQLite. Penggunaan TensorFlow Lite memungkinkan proses deteksi berlangsung secara efisien pada perangkat mobile.

2.4. Pengujian Aplikasi

Pengujian fungsional aplikasi dilakukan menggunakan metode Black Box Testing untuk memastikan seluruh fitur berjalan sesuai dengan kebutuhan sistem. Pengujian usability dilakukan menggunakan metode System Usability Scale (SUS) untuk mengukur tingkat kemudahan penggunaan aplikasi berdasarkan penilaian pengguna. Hasil dari seluruh pengujian digunakan sebagai dasar dalam mengevaluasi kualitas aplikasi yang telah dikembangkan

3. Hasil dan Diskusi

3.1. Hasil

Proses pengumpulan data dalam penelitian ini menghasilkan dataset citra sampah anorganik yang terbagi ke dalam dua kategori utama, yaitu sampah plastik sebanyak 511 citra gambar dan sampah kaca sebanyak 473 citra gambar, sehingga total keseluruhan dataset yang digunakan dalam penelitian ini berjumlah 984 citra gambar. Dataset tersebut diperoleh melalui kombinasi dua sumber, yaitu sumber terbuka seperti Google Open Images dan Kaggle, serta dataset khusus yang dikumpulkan secara mandiri dengan mengambil gambar sampah menggunakan kamera *smartphone* pada berbagai lokasi dan kondisi lingkungan. Penggabungan kedua sumber data ini bertujuan untuk memperkaya variasi visual objek yang akan dikenali oleh model, sehingga tidak hanya bergantung pada citra dari sumber terbuka yang mungkin memiliki karakteristik terbatas. Dari hasil pengumpulan data diperoleh data seperti pada contoh gambar 1.



Gambar 1 Contoh Dataset Sampah Plastik dan Kaca

Setelah dataset selesai dikumpulkan, dilakukan pembagian dataset terlebih dahulu agar hanya data pelatihan yang dilakukan augmentasi. Proses augmentasi yang diterapkan adalah perubahan *brightness*, *random contrast*, *noise* dan *flip*. Hasil dari pembagian dan augmentasi data yang dilakukan dirincikan pada tabel 1. Setelah proses augmentasi, dilakukan tahap anotasi menggunakan platform roboflow untuk menandai *bounding box* dan label kelas pada setiap objek sampah, sebagai persiapan sebelum data digunakan dalam proses pelatihan model.

Tabel 1 Persebaran Dataset

Data	Awal	Sesudah Augmentasi
Pelatihan	787	2361
Validasi	98	98
Pengujian	99	99

Sebelum melakukan proses pelatihan, semua data citra dilakukan proses penyesuaian orientasi berdasarkan metadata EXIF. Tahapan ini diperlukan agar seluruh objek berada dalam posisi tegak. Dilakukan juga *resize* ukuran dari citra agar seragam menjadi 320 x 320. Dalam proses pelatihan menggunakan platform Anaconda untuk membangun environment dengan versi python adalah 3.9.13. Diawal pelatihannya perlu dilakukan konfigurasi dari penggunaan arsitektur SSD-MobileNetV2. Optimizer Adam digunakan dengan learning rate sebesar 0,001, yang merupakan nilai awal (default) untuk algoritma tersebut. Optimizer Adam dipilih karena mampu memberikan konvergensi yang stabil melalui kombinasi adaptive learning rate dan momentum. Memanfaatkan metode transfer learning dari dataset COCO dilakukan juga penyesuaian sebagian layer pada backbone MobileNetV2. Pada tahap fine-tuning, hanya 10 layer terakhir dari backbone MobileNetV2 yang ditetapkan sebagai trainable sehingga bobot pada layer tersebut diperbarui selama proses pelatihan, sedangkan layer lainnya tetap dibekukan. Selain itu, layer deteksi SSD juga dilatih untuk mempelajari klasifikasi objek dan regresi *bounding box*. Strategi *partial fine-tuning* ini diterapkan untuk meningkatkan kemampuan adaptasi model terhadap dataset penelitian sekaligus mengurangi risiko *overfitting*. Dalam pelatihan juga menggunakan metode *callbacks* berupa *early stopping* agar proses pelatihan yang dapat dihentikan pada *epoch* dengan nilai validasi loss yang terbaik. Dengan menggunakan nilai

patience 20 dalam penelitian ini. Adapun dalam proses pelatihan dilakukan percobaan pada nilai batch size yang digunakan dengan rentang 4 hingga 32. Dari hasil pelatihan model diperoleh hasil seperti pada tabel 2.

Tabel 2 Nilai Akurasi dan Loss Validasi Pelatihan Model

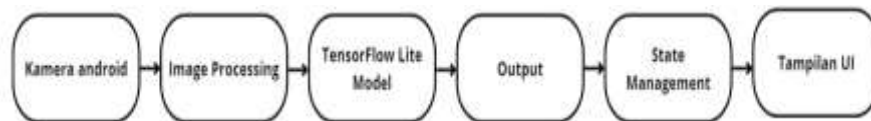
Batch Size	Akurasi Validasi	Loss Validasi
4	90,91%	0,3779
8	86,87%	0,3516
16	85,86%	0,3696
32	-	-

Dari hasil pelatihan model, nilai *batch size* 4 menghasilkan akurasi validasi sebesar 90,91%, lebih tinggi dibandingkan batch size 8 dan 16. Meskipun batch size 8 menghasilkan nilai validation loss yang lebih rendah, penelitian ini memilih batch size 4 karena memiliki kemampuan klasifikasi yang lebih baik berdasarkan akurasi validasi. Sedangkan pada saat menggunakan *batch size* 32 laptop yang digunakan mengalami *Out of Memory* (OOM) kerna VRAM tidak cukup dalam memproses seluruh *batch*. Dengan model terbaik yang diperoleh dari tahap pelatihan dilakukan evaluasi model dengan data pengujian dengan hasil seperti pada tabel 3.

Tabel 3 Hasil Evaluasi Model

Metrik Evaluasi	Nilai
Akurasi	89.90%
Presisi	90.00%
Recall	90.00%
F1-Score	90.00%
mAP	95.88%

Dari hasil evaluasi model yang dilakukan dapat mengidikasikan kemampuan model yang konsisten dalam mengkalsifikasi sampah plastik dan sampah kaca. Selain itu dari nilai mAP 95,88% menunjukkan model memiliki kemapuan yang baik dalam ketepatan mendeteksi objek dan lokasi bounding boxnya. Sehingga model ini sudah layak untuk dilakukan integrasi pada sebuah aplikasi. Namun sebelum melakukan integrasi, model terlebih dahulu harus disimpan dengan format .tflite agar dapat ditanam langsung pada aplikasi. Aplikasi dikembangkan menggunakan Android Studio, yaitu Integrated Development Environment (IDE) resmi untuk pengembangan aplikasi Android dengan bahasa pemrograman Kotlin. Adapun alur kerja model dalam aplikasi diilustrasikan pada gambar 2.



Gambar 2 Alur Kerja Aplikasi

Berdasarkan Gambar 2, proses deteksi diawali dengan pengambilan citra secara *real-time* menggunakan kamera pada perangkat Android. Citra yang diperoleh kemudian melalui tahap *image processing* berupa penyesuaian format dan ukuran agar sesuai dengan kebutuhan model TensorFlow Lite. Selanjutnya, citra diproses oleh model TensorFlow Lite untuk menghasilkan prediksi berupa kelas objek, nilai *confidence score*, serta koordinat *bounding box*. Hasil prediksi tersebut diteruskan ke komponen *state management* untuk mengelola data deteksi sebelum ditampilkan pada antarmuka pengguna (*User Interface*). Implementasi alur ini memungkinkan proses deteksi sampah berlangsung secara lokal (*on-device*) tanpa memerlukan koneksi internet, sehingga aplikasi mampu memberikan hasil deteksi secara cepat dan responsif.

Implementasi aplikasi menghasilkan beberapa fitur utama, yaitu deteksi sampah secara *real-time*, klasifikasi jenis sampah menjadi plastik atau kaca, visualisasi *bounding box* pada objek yang terdeteksi, penampilan nilai *confidence score*, serta penyimpanan riwayat hasil deteksi ke dalam basis data SQLite. Seluruh fitur berhasil diintegrasikan dengan model TensorFlow Lite dan dapat berjalan sesuai dengan rancangan sistem.



Gambar 3 Tampilan Ketika Aplikasi Dijalankan

Dari gambar 3 menunjukkan tampilan aplikasi saat proses deteksi berlangsung. Ketika pengguna menekan ikon kamera, aplikasi secara otomatis menangkap citra kemudian melakukan proses inferensi menggunakan model TensorFlow Lite yang tersimpan secara lokal pada perangkat. Hasil inferensi ditampilkan dalam bentuk *bounding box* yang menandai lokasi objek, disertai informasi jenis sampah dan nilai *confidence score* sebagai tingkat keyakinan model terhadap hasil klasifikasi. Selain itu, hasil deteksi beserta gambar yang diambil akan disimpan secara otomatis ke dalam riwayat aplikasi menggunakan basis data SQLite sehingga pengguna dapat melihat kembali hasil deteksi yang telah dilakukan. Selain melakukan klasifikasi, aplikasi juga menghitung jumlah objek sampah yang berhasil dideteksi pada setiap proses inferensi sehingga pengguna memperoleh informasi mengenai jenis dan jumlah sampah secara bersamaan. Selanjutnya dari hasil pengujian aplikasi dengan metode *black box testing* diperoleh hasil seperti pada tabel 4.

Tabel 4 Hasil Pengujian Black Box

No	Fitur yang Diuji	Skenario Pengujian	Hasil yang Diharapkan	Hasil	Status
1	Membuka aplikasi	Aplikasi dijalankan	Halaman utama tampil	Sesuai	Berhasil
2	Kamera	Menekan tombol kamera	Kamera aktif	Sesuai	Berhasil
3	Deteksi plastik	Mengarahkan kamera ke objek plastik	Objek dikenali sebagai plastik	Sesuai	Berhasil
4	Deteksi kaca	Mengarahkan kamera ke objek kaca	Objek dikenali sebagai kaca	Sesuai	Berhasil
5	Bounding box	Objek terdeteksi	Bounding box muncul	Sesuai	Berhasil
6	Confidence score	Objek terdeteksi	Nilai confidence tampil	Sesuai	Berhasil
7	Riwayat	Menyimpan hasil deteksi	Data tersimpan pada SQLite	Sesuai	Berhasil

Berdasarkan hasil Black Box Testing pada Tabel 4, seluruh fungsi utama aplikasi berhasil dijalankan sesuai dengan kebutuhan sistem. Pengujian meliputi proses membuka aplikasi, aktivasi kamera, deteksi objek, klasifikasi sampah, visualisasi *bounding box*, penampilan nilai *confidence score*, serta penyimpanan riwayat deteksi. Seluruh skenario pengujian memperoleh status berhasil tanpa ditemukan kesalahan fungsional. Aplikasi berhasil mengidentifikasi objek sampah plastik dan sampah kaca menggunakan model TensorFlow Lite yang diintegrasikan pada perangkat Android. Selain memastikan aspek fungsional aplikasi, penelitian ini juga melakukan evaluasi terhadap tingkat kemudahan penggunaan aplikasi menggunakan metode System Usability Scale (SUS). Pengujian ini dilakukan untuk mengetahui sejauh mana aplikasi mudah dipahami, dipelajari, dan digunakan oleh pengguna. Dalam penelitian ini terdapat 25 responden yang diminta untuk menggunakan aplikasi tanpa adanya petunjuk atau arahan sebelumnya. Hasil yang diperoleh dari evaluasi aspek fungsional aplikasi secara terperinci ditampilkan pada tabel 5.

Tabel 5 Hasil Pengujian SUS

Pernyataan	Skor 1	Skor 2	Skor 3	Skor 4	Skor 5	Total
1	0	0	5	13	7	25
2	0	0	2	8	15	25
3	11	10	2	1	1	25
4	12	6	2	2	3	25
5	10	7	5	3	0	25
6	0	1	2	11	11	25
7	0	0	3	10	12	25
8	11	12	1	1	0	25

9	2	0	2	11	10	25
10	7	5	3	7	3	25

Berdasarkan hasil kuesioner System Usability Scale (SUS) pada Tabel 5, sebagian besar responden memberikan penilaian positif terhadap aplikasi yang dikembangkan. Hal ini terlihat dari dominasi jawaban pada skala 4 (Setuju) dan 5 (Sangat Setuju) untuk pernyataan positif, sedangkan pada pernyataan negatif mayoritas responden memilih skala 1 (Sangat Tidak Setuju) dan 2 (Tidak Setuju). Distribusi jawaban tersebut menunjukkan bahwa aplikasi dinilai mudah dipahami, mudah dipelajari, serta memiliki antarmuka yang mendukung proses penggunaan oleh pengguna.

3.2. Pembahasan

Hasil penelitian menunjukkan bahwa model SSD-MobileNetV2 yang dikembangkan mampu mendeteksi sampah plastik dan kaca dengan performa yang baik. Konfigurasi pelatihan menggunakan transfer learning, *partial fine-tuning* pada 10 layer terakhir backbone MobileNetV2, serta pemilihan batch size 4 menghasilkan akurasi validasi tertinggi sebesar 90,91%. Selain itu, hasil evaluasi pada data pengujian menunjukkan akurasi sebesar 89,90%, nilai presisi, *recall*, dan *F1-score* masing-masing sebesar 90,00%, serta mAP sebesar 95,88%. Hasil tersebut menunjukkan bahwa model tidak hanya mampu mengklasifikasikan objek secara konsisten, tetapi juga memiliki ketepatan yang tinggi dalam menentukan lokasi objek melalui *bounding box*. Tingginya nilai mAP mengindikasikan bahwa model memiliki kemampuan deteksi objek yang baik sehingga sesuai untuk diterapkan pada sistem deteksi sampah berbasis perangkat bergerak.

Performa model dipengaruhi oleh penerapan transfer learning yang memanfaatkan bobot awal dari dataset COCO sehingga proses pembelajaran dapat berfokus pada karakteristik sampah plastik dan kaca tanpa harus melatih model dari awal. Strategi *partial fine-tuning* dengan mengaktifkan 10 layer terakhir MobileNetV2 memungkinkan model menyesuaikan fitur tingkat tinggi terhadap dataset penelitian, sementara sebagian besar layer tetap mempertahankan kemampuan ekstraksi fitur yang telah dipelajari sebelumnya. Di sisi lain, hasil percobaan menunjukkan bahwa penggunaan *batch size* yang lebih besar tidak selalu menghasilkan performa yang lebih baik. Meskipun *batch size* 8 memiliki nilai *validation loss* yang lebih rendah, *batch size* 4 menghasilkan akurasi validasi tertinggi sehingga dipilih sebagai konfigurasi terbaik. Sementara itu, penggunaan *batch size* 32 tidak dapat diterapkan karena keterbatasan kapasitas VRAM perangkat yang menyebabkan proses pelatihan mengalami Out of Memory (OOM).

Hasil implementasi juga menunjukkan bahwa model berhasil diintegrasikan ke dalam aplikasi Android menggunakan TensorFlow Lite sehingga proses deteksi dapat dilakukan secara on-device tanpa memerlukan koneksi internet. Pengujian Black Box Testing menunjukkan seluruh fitur utama aplikasi, mulai dari pengambilan gambar, deteksi objek, visualisasi *bounding box*, hingga penyimpanan riwayat deteksi, telah berjalan sesuai dengan kebutuhan sistem. Selain itu, hasil pengujian System Usability Scale (SUS) memperlihatkan bahwa sebagian besar responden memberikan penilaian positif terhadap aplikasi, yang menunjukkan bahwa aplikasi mudah dipahami, dipelajari, dan digunakan. Dengan demikian, penelitian ini berhasil menjawab tujuan penelitian, yaitu mengembangkan aplikasi mobile yang mampu mendeteksi sampah plastik dan kaca secara *real-time* dengan performa deteksi yang baik serta tingkat kemudahan penggunaan yang dapat diterima oleh pengguna.

4. Kesimpulan

Penelitian ini berhasil mengembangkan aplikasi mobile berbasis SSD-MobileNetV2 untuk mendeteksi sampah anorganik jenis plastik dan kaca secara *real-time*. Model yang dikembangkan memperoleh akurasi sebesar 89,90%, presisi 90,00%, *recall* 90,00%, *F1-score* 90,00%, dan *mean Average Precision* (mAP) sebesar 95,88%, sehingga menunjukkan kemampuan yang baik dalam melakukan klasifikasi objek sekaligus menentukan lokasi *bounding box*. Model tersebut berhasil diintegrasikan ke dalam aplikasi Android menggunakan TensorFlow Lite, sehingga proses deteksi dapat dilakukan secara on-device tanpa memerlukan koneksi internet. Hasil pengujian fungsional menunjukkan seluruh fitur aplikasi berjalan sesuai dengan kebutuhan sistem, sedangkan pengujian *usability* menunjukkan bahwa aplikasi mudah dipahami dan digunakan oleh pengguna. Dengan demikian, aplikasi yang dikembangkan berpotensi mendukung proses identifikasi dan pemilahan sampah secara lebih praktis melalui perangkat mobile. Untuk penelitian selanjutnya, pengembangan dapat dilakukan dengan menambahkan variasi kelas sampah, memperluas jumlah dan keragaman dataset, serta mengevaluasi penggunaan arsitektur deteksi objek lain yang lebih mutakhir untuk meningkatkan akurasi dan kemampuan generalisasi model pada berbagai kondisi lingkungan.

Referensi

- [1] B. Fang dkk., *Artificial intelligence for waste management in smart cities: a review*, vol. 21, no. 4. Springer International Publishing, 2023. doi: 10.1007/s10311-023-01604-3.
- [2] Z. Dong, L. Yuan, B. Yang, F. Xue, dan W. Lu, "Benchmarking computer vision models for automated construction waste sorting," *Resour. Conserv. Recycl.*, vol. 213, no. February, hlm. 1–8, 2025, doi: 10.1016/j.resconrec.2024.108026.
- [3] E. Oktayaessofa, C. A. Sari, E. H. Rachmawanto, dan N. M. Yaacob, "Classification of Organic and Non-Organic Waste With Cnn-MobileNet-V2," *Jurnal Teknik Informatika (Jutif)*, vol. 5, no. 4, hlm. 1173–1180, 2024, doi: 10.52436/1.jutif.2024.5.4.2165.
- [4] A. D. Pritama, V. Sandy Kusuma, W. M. Baihaqi, dan P. Subarkah, "Enhancing Waste Classification with MobileNetV2: Adding a Plastic Sachets Class for Sustainable Management," 2025. doi: 10.15294/edukom.v12i1.18931.
- [5] F. Andriani, A. Yuliaty, A. Yaturahmah, dan S. Mutmainah, "MobileNetV2 Transfer Learning Implementation for Waste Classification," *Indonesian Applied Research Computing and Informatics*, vol. 1, no. 2, hlm. 37–46, 2026, doi: <https://doi.org/10.64479/iarci.v1i2.62>.
- [6] F. Fotovvatikhah, I. Ahmedy, R. M. Noor, dan M. U. Munir, "A Systematic Review of AI-Based Techniques for Automated Waste Classification," *Sensors*, vol. 25, no. 10, 2025, doi: 10.3390/s25103181.
- [7] O. Chacón-Albero, M. Campos-Mocholí, C. Marco-Detchart, V. Julian, J. A. Rincon, dan V. Botti, "AI for Sustainable Recycling: Efficient Model Optimization for Waste Classification Systems," *Sensors*, vol. 25, no. 12, hlm. 1–18, 2025, doi: 10.3390/s25123807.
- [8] I. G. P. Sindu, M. Sudarma, R. S. Hartati, dan N. Gunantara, "Classification of Tri Pramana learning activities in virtual reality environment using convolutional neural network," *IAES International Journal of Artificial Intelligence*, vol. 13, no. 3, hlm. 2840–2853, 2024, doi: 10.11591/ijai.v13.i3.pp2840-2853.
- [9] W. H. A. Wijaya, K. Agustini, dan I. N. E. Mertayasa, "Pengembangan Platform Digital Agrow untuk Pemanfaatan Buah dan Sayur Imperfect Guna Mengurangi Food Waste dan Peningkatan Pendapatan Petani," *RIGGS: Journal of Artificial Intelligence and Digital Business*, vol. 4, no. 4, hlm. 5732–5739, 2025, doi: 10.31004/riggs.v4i4.4466.
- [10] K. A. Seputra, A. A. G. Y. Paramartha, G. A. Pradnyana, dan K. Y. E. Aryanto, "A Middleware Applications Design for Health Information Sharing," *Jurnal Rekayasa Sistem dan Teknologi Informasi*, vol. 8, no. 3, hlm. 321–332, 2024, doi: 10.29207/resti.v8i3.5707.
- [11] A. M. Sari, I. M. G. Sunarya, dan I. D. Maysanjaya, "Perancangan Aplikasi Analisis Sentimen Mengenai Program Petani Milenial Berbasis IndoBERT," *Kumpulan Artikel Mahasiswa Pendidikan Teknik Informatika*, vol. 15, no. 1, hlm. 118–126, 2026, doi: <https://doi.org/10.23887/karmapati.v15i1.108665>.
- [12] D. P. M. Saraswati, K. A. Seputra, dan L. J. E. Dewi, "Developing the Interface of HaoSpace Mood Tracking Application Using Design Thinking," *Journal of Information System Research (JOSH)*, vol. 7, no. 1, hlm. 21–30, 2025, doi: 10.47065/josh.v7i1.8365.
- [13] K. A. Suputra, A. Aan, J. Permana, dan U. P. Ganesha, "Implementasi Mobile Homecare Berbasis User Centered Desain (UCD)," *Masaliq Jurnal Pendidikan dan Sains*, vol. 6, no. 1, hlm. 442–456, 2026, doi: <https://doi.org/10.58578/masaliq.v6i1.8737>.
- [14] G. S. Mahendra, I. N. I. Wiradika, dan K. Y. E. Aryanto, "Android-based decision support system using MAGIQ-MARCOS for digital bank selection," *MATRIX: Jurnal Manajemen Teknologi dan Informatika*, vol. 15, no. 3, hlm. 148–162, 2025, doi: 10.31940/matrix.v15i3.148-162.
- [15] L. J. E. Dewi, K. A. Seputra, dan I. N. S. W. Wijaya, "Pengembangan Aplikasi Mobile Sistem Informasi Produk Pertanian Kabupaten Buleleng," *SINTECH (Science and Information Technology) Journal*, vol. 5, no. 1, hlm. 66–74, 2022, doi: 10.31598/sintechjournal.v5i1.999.
- [16] M. Waruwu, "Metode Penelitian dan Pengembangan (R&D): Konsep, Jenis, Tahapan dan Kelebihan," *Jurnal Ilmiah Profesi Pendidikan*, vol. 9, no. 2, hlm. 1220–1230, 2024, doi: <https://doi.org/10.29303/jipp.v9i2.2141>.