

## Klasifikasi Citra Wadah Minuman Reusable dan Non-Reusable Menggunakan *MobileNetV2*

Dea Ramanda<sup>1</sup>, Fuad Nur Hasan<sup>2\*</sup>, Antonius Yadi Kuntoro<sup>3</sup>

<sup>1,2,3</sup>Program Studi Informatika, Universitas Bina Sarana Informatika, Kota Jakarta Pusat, Indonesia

### INFORMASI ARTIKEL

*Sejarah Artikel:*

Diterima Redaksi: 19-07-2025

Revisi Akhir: 17-08-2025

Diterbitkan Online: 10-09-2025

### KATA KUNCI

Klasifikasi Citra,

Wadah Minuman,

*Reusable*,

*Computer Vision*,

*MobileNet2*

### KORESPONDENSI

E-mail: fuad.fnu@bsi.ac.id

### ABSTRACT

Single-use plastic waste, particularly from beverage bottles, remains a significant contributor to the increasing volume of waste in Indonesia. The limited use of reusable beverage containers underscores the urgent need for technological innovations that can support efficient waste segregation. Addressing this issue, the present study proposes a computer vision-based image classification system designed to automatically distinguish between reusable and non-reusable drinking containers. This research adopts a quantitative experimental approach, employing the *MobileNetV2* architecture through transfer learning techniques. The model was trained with augmented and normalized datasets to enhance its generalization across diverse image inputs. Evaluation results demonstrate strong classification performance, achieving 96% accuracy, 99% precision (for tumblers), 95% recall, and a 97% F1-score. These outcomes indicate the effectiveness of *MobileNetV2* in identifying visual patterns between container types and its potential for deployment in image-driven waste management systems.

## 1. PENDAHULUAN

Permasalahan pengelolaan sampah di Indonesia terus menjadi tantangan besar, terutama seiring dengan meningkatnya jumlah penduduk dan konsumsi produk-produk berbahan dasar plastik [1]. Berdasarkan data dari Sistem Informasi Pengelolaan Sampah Nasional Kementerian Lingkungan Hidup dan Kehutanan tahun 2024, volume sampah di Indonesia telah mencapai 33 juta ton per tahun, dengan sampah plastik menyumbang sekitar 19,64% dari total tersebut. Salah satu sumber utama sampah plastik adalah botol minuman sekali pakai yang banyak digunakan. Meskipun praktis dan murah, botol plastik merupakan jenis sampah non-organik yang sulit terurai secara alami [2]. Sementara itu, alternatif seperti penggunaan wadah minuman pakai ulang (*reusable*) yang lebih ramah lingkungan belum banyak diadopsi secara luas oleh masyarakat.

Melihat kondisi tersebut, dibutuhkan pendekatan berbasis teknologi untuk mendukung sistem pemilahan sampah secara otomatis, akurat, dan efisien. Salah satu solusi yang berkembang adalah penggunaan teknologi *computer vision* yang

dikombinasikan dengan algoritma *machine learning* dalam tugas klasifikasi citra. Teknologi ini memungkinkan sistem mengenali objek visual berdasarkan pola tertentu dengan bantuan algoritma *machine learning*, dimana model klasifikasi dilatih dari data berlabel untuk membedakan objek berdasarkan karakteristik visualnya [3]. Dalam konteks ini, *MobileNetV2* menjadi salah satu arsitektur *deep learning* yang banyak digunakan karena sifatnya yang ringan, efisien, dan ideal untuk diterapkan pada perangkat dengan keterbatasan sumber daya.

Untuk memastikan akurasi dan efektivitas model yang dikembangkan, penelitian ini menggunakan pendekatan kuantitatif dengan metode eksperimen [4], dengan menerapkan arsitektur *MobileNetV2* sebagai model utama. *MobileNetV2* dipilih karena efisien, ringan, dan cocok untuk pengolahan dataset berukuran kecil serta perangkat dengan sumber daya terbatas [5].

Sejumlah studi sebelumnya telah menggarisbawahi efektivitas pendekatan berbasis *computer vision* dalam pengelolaan sampah berbasis citra. Ramadhan dkk. dalam penelitian [6] yang dilakukannya yang mengembangkan sistem deteksi botol plastik di lingkungan perairan menggunakan YOLOv4-Tiny pada

Raspberry Pi 5, yang berhasil mencapai akurasi hingga 80% dan mAP sebesar 97,94%. Meskipun fokusnya pada deteksi real-time, penelitian tersebut membuktikan kemampuan model ringan pada perangkat edge. Selanjutnya, Prasetio dan Pratiwi juga dalam penelitian [7] mengembangkan aplikasi mobile berbasis MobileNet untuk klasifikasi jenis sampah plastik dan berhasil memperoleh akurasi sebesar 98%, menunjukkan potensi pemanfaatan model ringan untuk klasifikasi citra di perangkat bergerak. Penelitian [8] yang serupa juga dilakukan oleh Rismayadi dan kawan-kawan, yang menggunakan *MobileNetV2* dalam aplikasi Android untuk klasifikasi limbah dengan hasil akurasi pelatihan 95,45% dan validasi 84,83%.

Di sisi lain, deteksi sampah juga menjadi perhatian dalam penelitian [9] oleh Theofilus dan Kurniawan melalui penggunaan YOLOv8 untuk mendeteksi sampah mengapung di Sungai Ciliwung, dengan capaian presisi 84,02%, recall 91,03%, dan f1-score 87,38%. Muhammad dan kawan-kawan dalam penelitian [10] juga bahkan mengintegrasikan *SSD-MobileNetV2* dengan sistem Arduino dan Raspberry Pi untuk mendeteksi dan memilah sampah otomatis secara real-time, dengan akurasi sistem hingga 92%.

Meskipun beragam pendekatan telah dilakukan, sebagian besar penelitian tersebut belum secara spesifik menargetkan pemisahan antara wadah minuman reusable dan non-reusable sebagai kategori klasifikasi. Padahal, klasifikasi berbasis fungsi pakai memiliki urgensi tersendiri dalam mendorong perilaku konsumsi yang lebih ramah lingkungan. Selain itu, implementasi sistem klasifikasi pada antarmuka berbasis web interaktif juga masih jarang dijumpai, padahal pendekatan ini memiliki keunggulan dalam hal aksesibilitas dan potensi edukatif terhadap masyarakat luas.

Penelitian ini hadir untuk menjawab kebutuhan tersebut dengan mengembangkan model klasifikasi citra berbasis *MobileNetV2* yang dirancang khusus untuk membedakan antara wadah minuman reusable dan non-reusable. Model dilatih dengan pendekatan transfer learning dan teknik augmentasi data guna meningkatkan generalisasi dan akurasi terhadap citra baru. Selain itu, model dikemas dalam bentuk aplikasi berbasis web menggunakan Gradio dan dideploy melalui platform Hugging Face, sehingga memungkinkan penggunaan yang luas tanpa ketergantungan pada perangkat keras khusus.

Dengan pendekatan tersebut, penelitian ini memiliki dua kontribusi utama, yaitu pertama, menawarkan solusi klasifikasi citra berbasis deep learning yang spesifik dan kontekstual terhadap isu konsumsi plastik; kedua, menyediakan sistem interaktif berbasis web yang dapat diakses secara luas untuk mendukung edukasi dan pengambilan keputusan dalam pengelolaan sampah. Inovasi ini diharapkan dapat memperluas pemanfaatan teknologi visual dalam bidang lingkungan dan mendorong perubahan perilaku masyarakat menuju pola konsumsi yang lebih berkelanjutan.

## 2. TINJAUAN PUSTAKA

### 2.1. Computer Vision

Computer Vision adalah bidang ilmu yang mempelajari metode untuk mengambil informasi berupa angka atau simbol. [11] Beberapa proses yang dilakukan dalam Computer Vision meliputi pengambilan gambar, pengeditan gambar, pemisahan bagian-bagian gambar, ekstraksi ciri-ciri, serta penjernihan gambar.

### 2.2. Klasifikasi Citra

Proses mengatributkan label kategori atau kelas pada citra berdasarkan karakteristik visual atau fitur-fitur yang diekstraksi dari citra merupakan pengertian dari klasifikasi citra. Proses ini bertujuan mengenali objek atau pola yang ada dalam citra, lalu mengelompokkan objek tersebut ke dalam kelas tertentu. [12]

### 2.3. Convolutional Neural Network

Convolutional Neural Network (CNN) adalah metode pembelajaran mesin yang digunakan untuk memproses data berupa gambar. Metode ini mampu mengidentifikasi berbagai aspek dalam gambar dengan cara memberikan bobot dan bias yang dapat dipelajari, sehingga mampu membedakan satu objek dari objek lainnya.

### 2.4. Transfer Learning

*Transfer learning* adalah metode yang bekerja dengan memanfaatkan arsitektur network yang telah ada [13]. Dalam metode ini, model yang telah dilatih untuk suatu tugas akan diubah dan disesuaikan parameter internalnya sehingga dapat digunakan untuk menyelesaikan tugas lain

### 2.5. TensorFlow

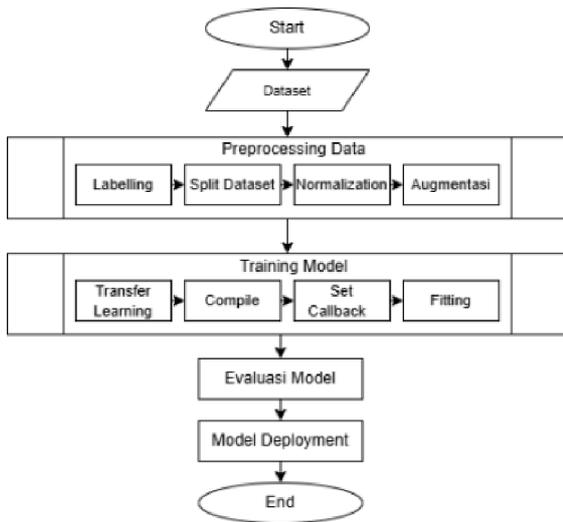
“TensorFlow adalah kerangka kerja komputasi untuk mengembangkan model machine learning. TensorFlow menyediakan berbagai alat yang memudahkan pembuatan model dan dapat menjalankan grafis pada beberapa platform hardware, termasuk CPU, GPU, dan TPU [14].

### 2.6. Hugging Face

Hugging Face adalah sebuah platform berbasis website komunitas yang berfokus pada pengembangan model kecerdasan buatan, terutama bidang pemrosesan bahasa alami (NLP) [15]. Di Hugging Face, seseorang dapat membagikan model, dataset, atau hasil aplikasi dari penelitian atau pekerjaan mereka, sehingga orang lain dapat melihat, mempelajari, atau menggunakan hasil karya tersebut.

## 3. METODOLOGI

Penjelasan dari setiap tahapan dalam diagram Gambar 1 dijabarkan pada subbab berikut ini:



Gambar 1 Metodologi Penelitian

### 3.1. Dataset

Data yang digunakan dalam penelitian ini berupa citra dua jenis wadah minuman, yaitu wadah reusable (seperti tumbler atau botol minum pakai ulang) dan wadah non-reusable (seperti botol plastik sekali pakai).

Sampel citra yang digunakan dalam penelitian ini berjumlah total 1200 gambar. Gambar tersebut diperoleh sebagian di dapat dari pencarian daring, khususnya dari laman Google Image ataupun open source dataset, atau dokumentasi manual menggunakan kamera laptop, dengan pengambilan gambar dilakukan dari berbagai sudut, jarak, dan kondisi pencahayaan. Variasi ini dilakukan untuk meningkatkan keragaman dataset dan memperkuat kemampuan generalisasi model. Proses pengumpulan data ini dilakukan dengan tetap memperhatikan variasi jenis wadah, bentuk, warna, serta kondisi latar belakang guna menghasilkan model klasifikasi yang lebih akurat.

### 3.2. Preprocessing Data

Data preprocessing merupakan tahapan menyiapkan data sebelum nantinya diproses ke tahap berikutnya. Beberapa langkah preprocessing yang dilakukan meliputi:

#### 3.2.1. Labelling Data

Setiap citra secara otomatis diberi label sesuai dengan nama folder utamanya, yaitu "tumbler" untuk wadah minuman reusable dan "botol" untuk wadah non-reusable. Proses pelabelan dilakukan secara otomatis menggunakan struktur direktori, di mana setiap folder mewakili satu kelas. Pendekatan ini dipilih untuk mempercepat proses pelabelan dan meminimalkan kesalahan manusia, selama struktur folder data telah disusun dengan benar. Selanjutnya, label dalam bahasa Inggris diubah ke dalam Bahasa Indonesia menggunakan pemetaan (mapping) label agar konsisten dengan narasi penelitian..

#### 3.2.2. Split Dataset

Dataset citra yang telah melalui proses preprocessing dibagi menjadi tiga bagian dengan rasio perbandingan 70:20:10, yaitu

70% data digunakan untuk pelatihan, 20% data digunakan untuk validasi, dan 10% data digunakan untuk pengujian. Pembagian ini bertujuan agar model dapat belajar dari data pelatihan, sementara performanya dapat dievaluasi secara berkala menggunakan data validasi, guna mencegah overfitting dan memastikan generalisasi model berjalan baik, lalu data diuji menggunakan data yang tidak pernah dilihat sebelumnya menggunakan data uji.

#### 3.2.3. Normalization

Normalisasi merupakan proses dimana mengubah skala nilai piksel dari rentang 0–255 menjadi 0–1, sehingga model dapat belajar secara lebih efisien dan stabil [11]. Dalam penelitian ini, proses normalisasi dilakukan menggunakan parameter  $rescale=1.0/255.0$  pada objek `ImageDataGenerator` dari library Keras seperti yang terlihat pada gambar IV.8. Adapun nilai 255 dipilih karena merupakan nilai maksimum dari piksel gambar berwarna RGB.

#### 3.2.4. Augmentasi

Augmentasi data merupakan teknik untuk menambah variasi pada dataset secara artifisial dengan melakukan transformasi citra, tanpa perlu mengumpulkan data baru [12]. Dalam penelitian ini dilakukan augmentasi citra menggunakan teknik seperti rotasi, perbesaran atau pengecilan, translasi, pencerminan, perubahan kecerahan, shear dan channel shift. Pada penelitian ini, augmentasi diterapkan hanya pada dataset pelatihan yaitu training set menggunakan berbagai metode transformasi citra melalui pustaka `ImageDataGenerator` dari Keras. Teknik-teknik ini diharapkan mampu memperkaya variasi citra latih dan meningkatkan kemampuan generalisasi model terhadap data baru.

### 3.3. Training Model

Setelah preprocessing, data dilanjutkan ke tahap pelatihan. Proses ini terdiri dari beberapa langkah utama, yaitu compile model (compile), membuat callback, pelatihan (training).

#### 3.3.1. Transfer Learning

Dalam tahapan ini, digunakan pendekatan transfer learning yaitu memanfaatkan model yang sudah ada untuk dikembangkan atau dimodifikasi lebih lanjut untuk menghemat waktu pelatihan, meningkatkan akurasi, serta mengurangi kebutuhan akan jumlah data latih yang besar [13]. *MobileNetV2* dipilih karena arsitektur ini dirancang untuk kebutuhan komputasi yang ringan namun tetap memiliki performa yang bagus, terutama pada tugas klasifikasi citra.

#### 3.3.2. Compile

Sebelum proses pelatihan dimulai, model harus dilakukan compile. Proses compile ini mencakup penentuan fungsi loss function, optimizer, dan learning rate yang akan digunakan selama pelatihan.

1. Loss Function : Loss function digunakan untuk mengukur seberapa jauh prediksi model dari label sebenarnya. Pada penelitian ini digunakan binary crossentropy sebagai loss function yang diperuntukan untuk klasifikasi yang hanya terdiri dari dua kelas (biner).
2. Optimizer : Optimizer berperan dalam memperbarui bobot-bobot neural network berdasarkan nilai error (loss) yang dihasilkan. Penelitian ini menggunakan optimizer

Adam (Adaptive Moment Estimation) karena mampu mengubah laju pembelajaran berdasarkan seberapa besar dan sering perubahan gradien terjadi .

3. Learning Rate: Learning rate berguna mengontrol seberapa besar langkah yang diambil oleh algoritma optimisasi selama proses pelatihan.

### 3.3.3. Callback

Callback berperan layaknya asisten yang dapat membantu mengontrol dan memonitor proses training. Salah satu langkah yang dapat dilakukan adalah pemberhentian training otomatis (early stopping), menyimpan model terbaik, atau mencatat log training itu sendiri jika sudah mencapai performa tertentu [14]. Proses ini merupakan proses start atau mulai bagi tahap training model. Setelah data dibagi ke dalam batch tertentu, data akan dilatih selama beberapa epoch. Selama proses pelatihan, model berusaha menyesuaikan bobot internalnya dengan bantuan optimizer yang telah ditentukan.

### 3.4. Evaluasi Model

Setelah proses training selesai, dilakukan evaluasi untuk mengukur seberapa baik model bekerja terhadap data uji atau data validasi. Hasil evaluasi ini digunakan untuk melihat apakah model mampu menggeneralisasi pola dari data pelatihan ke data yang belum pernah dilihat sehingga dalam prosesnya menggunakan data uji. Evaluasi dilakukan dengan confusion matrix yang memungkinkan untuk mengidentifikasi jumlah prediksi yang tepat (True Positives dan True Negatives) serta prediksi yang tidak tepat (False Positives dan False Negatives), dengan begitu dapat mengetahui nilai dari performa metrik tertentu seperti akurasi, presisi, recall, dan F1-score [15]. Tujuannya adalah untuk mengetahui seberapa baik model dapat mengklasifikasikan gambar dengan benar.

## 4. HASIL DAN PEMBAHASAN

### 4.1. Distribusi Dataset

Data yang digunakan dalam penelitian ini berupa citra dua jenis wadah minuman, yaitu wadah reusable (seperti tumbler atau botol minum pakai ulang) dan wadah non-reusable (seperti gelas plastik dan botol plastik sekali pakai). Tujuan dari pengumpulan data dengan menggabungkan beberapa sumber ini adalah untuk menyediakan dataset yang beragam guna melatih dan menguji model klasifikasi berbasis computer vision.

Untuk mengetahui jumlah citra pada masing-masing kelas (tumbler dan botol), peneliti melakukan pemindaian terhadap direktori dataset menggunakan modul os. Setiap file citra dihitung dan hasilnya disimpan ke dalam sebuah DataFrame agar mudah divisualisasikan. Sehingga dapat diketahui dataset terdiri atas 1.200 citra, dimana 790 diantaranya merupakan kelas tumbler, sedangkan sisanya yaitu 410 merupakan kelas botol. Jumlah masing-masing kelas tidak seimbang, dengan dominasi pada kelas tumbler.

Gambar tersebut diperoleh melalui dua metode, yaitu pencarian daring, khususnya dari laman Google Images dan sumber dataset terbuka (open source), serta dokumentasi manual menggunakan kamera handphone dengan rasio gambar 1:1. Pengambilan gambar dilakukan dari berbagai sudut, jarak, serta kondisi pencahayaan untuk meningkatkan keberagaman citra dalam

dataset. Visualisasi distribusi ditampilkan dalam grafik batang (Tabel 1).

Tabel 1. Contoh Penulisan Tabel

Kelas	Jumlah Data
Tumbler	790
Botol	410

Setelah proses pelabelan, dataset dibagi menjadi tiga subset yaitu data pelatihan (training), data validasi (validation), dan data pengujian (testing). Pembagian ini dilakukan untuk memastikan bahwa proses pelatihan model dapat dilakukan secara terstruktur dan untuk mencegah overfitting dan bias evaluasi.

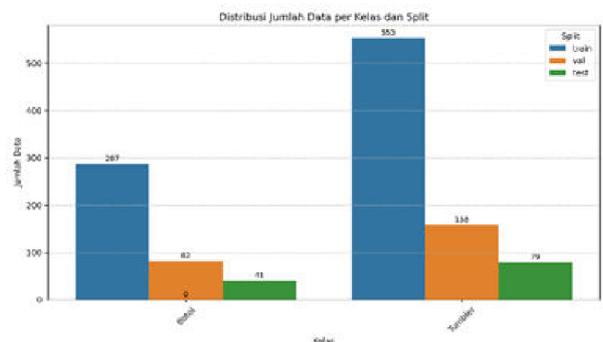
Dengan menggunakan pustaka splitfolders, proses pembagian secara otomatis dilakukan. Dalam penelitian ini, dataset dibagi dengan rasio 70% untuk pelatihan, 20% untuk validasi, dan 10% untuk pengujian. Selain itu, untuk memastikan pembagian data yang konsisten dan dapat direplikasi, seed=42 digunakan. Sehingga dapat diketahui pembagian dataset ke dalam tiga subset utama, yaitu data pelatihan (70%), validasi (20%), dan pengujian (10%) dari total 1.200 citra pada Tabel 2

Tabel 1. Contoh Penulisan Tabel

Subset	Jumlah Data
Pelatihan	840
Validasi	240
Pengujian	120

Selanjutnya, Untuk memastikan bahwa proses pembagian dataset telah menghasilkan distribusi yang seimbang antara masing-masing kelas dalam tiap subset (train, validation, dan test), dilakukan perhitungan jumlah citra berdasarkan kelas dan jenis subset-nya.

Proses ini dilakukan dengan mendefinisikan daftar nama subset (train, val, dan test) serta direktori utama hasil splitting (data\_deteksi). Menggunakan fungsi count\_files\_per\_class() yang sebelumnya telah didefinisikan untuk menghitung jumlah file gambar pada masing-masing subfolder (kelas). Lalu data jumlah file dikumpulkan dalam struktur dictionary bertingkat all\_class\_counts, di mana setiap entri menyimpan jumlah gambar untuk satu kelas, lengkap dengan jumlahnya pada masing-masing subset akan menghasilkan grafik yang terlihat pada gambar 2 di



bawah

Gambar 2 Distribusi Per Kelas Di Setiap Subset

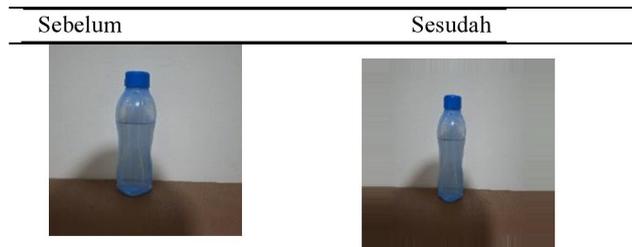
### 4.2. Augmentasi Data

Pada penelitian ini, augmentasi diterapkan hanya pada dataset pelatihan (training set), menggunakan berbagai metode

transformasi citra melalui pustaka ImageDataGenerator dari Keras. Teknik augmentasi yang digunakan antara lain:

4.2.1. *Zooming*

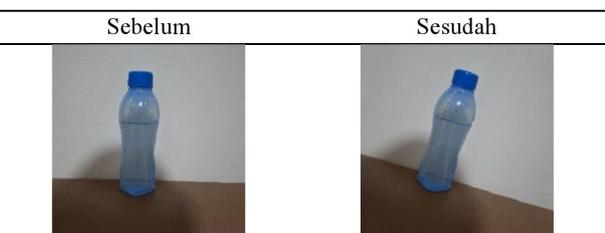
Yaitu memperbesar atau memperkecil sebagian area citra secara acak. Pada penelitian ini diterapkan perbesaran dengan nilai 0.5 atau gambar akan diperbesar atau diperkecil hingga 50% dari ukuran aslinya. Berikut ditampilkan hasil visualisasi sebelum dan sesudah diterapkan zooming pada gambar 3 di bawah.



Gambar 3. Contoh Augmentasi *Zooming* Pada Citra Tumbler

4.2.2. *Rotating*

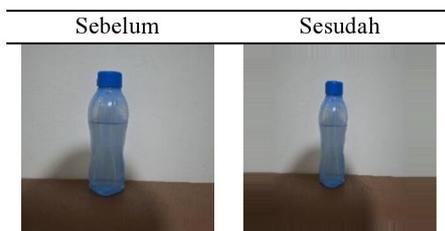
Yaitu memutar citra beberapa derajat ke kiri atau kanan. Pada penelitian ini diterapkan pemutaran citra sebesar  $-40^\circ$  sampai  $40^\circ$  baik ke kiri ataupun ke kanan. Berikut ditampilkan hasil visualisasi sebelum dan sesudah diterapkan rotating pada gambar 4 di bawah.



Gambar 4. Contoh Augmentasi *Rotating* Pada Citra Tumbler

4.2.3. *Width Shift*

Yaitu menggeser posisi objek dalam citra secara horizontal atau secara lebar. Pada penelitian ini diterapkan penggeseran posisi citra sebesar 0.3 secara acak ke kanan atau kiri sampai 30% dari ukuran. Berikut ditampilkan hasil visualisasi sebelum dan sesudah diterapkan width shift pada Gambar 5 di bawah.

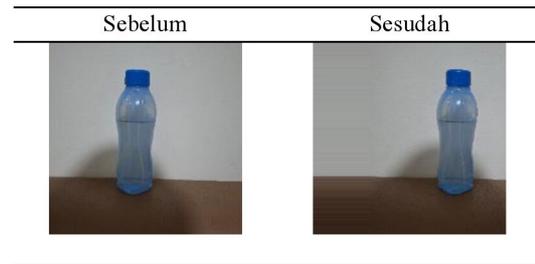


Gambar 5. Contoh Augmentasi *Width Shift* Pada Citra Tumbler

4.2.4. *Height Shift*

Yaitu menggeser posisi objek dalam citra secara vertikal atau secara tinggi. Pada penelitian ini diterapkan penggeseran posisi citra sebesar 0.3 secara acak ke atas atau ke bawah sampai 30%

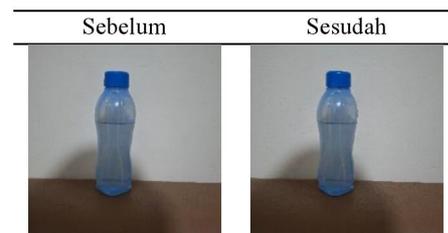
dari ukuran gambar. Berikut ditampilkan hasil visualisasi sebelum dan sesudah diterapkan height shift pada Gambar 6 di bawah.



Gambar 6. Contoh Augmentasi *Height Shift* Pada Citra Tumbler

4.2.5. *Shear*

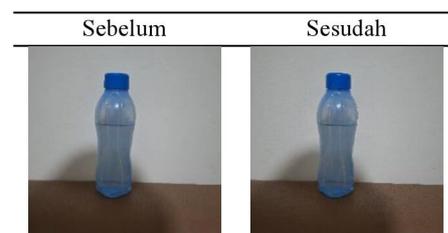
Yaitu mengubah bentuk objek dengan distorsi sudut miring. Pada penelitian ini diterapkan distorsi citra sebesar 0.3. Berikut ditampilkan hasil visualisasi sebelum dan sesudah diterapkan height shift pada Gambar 7 di bawah.



Gambar 7. Contoh Augmentasi *Shear* Pada Citra Tumbler

4.2.6. *Horizontal Flip*

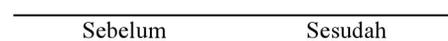
Yaitu membalik gambar dari sisi kanan ke kiri. Berikut ditampilkan hasil visualisasi sebelum dan sesudah diterapkan Horizontal Flip pada Gambar 8 di bawah.



Gambar 8. Contoh Augmentasi *Shear* Pada Citra Tumbler

4.2.7. *Vertical Flip*

Yaitu membalik gambar dari sisi atas ke bawah. Pada penelitian ini diterapkan pembalikan citra. Berikut ditampilkan hasil visualisasi sebelum dan sesudah diterapkan Vertikal Flip pada Gambar 9 di bawah.

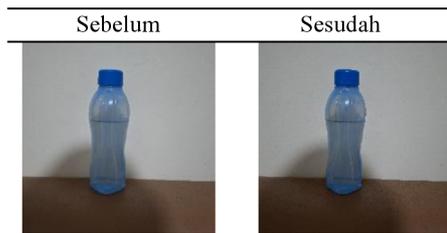




Gambar 9. Contoh Augmentasi *Vertical Flip* Pada Citra Tumbler

#### 4.2.8. Channel Shift

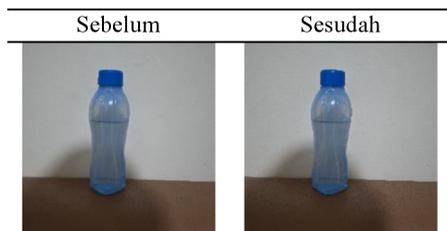
Yaitu menggeser intensitas warna secara acak. Pada penelitian ini nilai warna di setiap channel (R, G, B) bisa digeser nilainya hingga  $\pm 30$ . Berikut ditampilkan hasil visualisasi sebelum dan sesudah diterapkan Channel Shift pada Gambar 10 di bawah.



Gambar 10. Contoh Augmentasi *Channel Shift* Pada Citra Tumbler

#### 4.2.9. Fill Mode

Saat gambar diputar atau digeser, akan muncul sebuah ruang kosong karena canvas tidak ikut berpindah, karenanya dengan fill mode='nearest' artinya ruang kosong diisi dengan warna piksel tetangga terdekat supaya tidak ada pixel yang kosong atau hitam. Berikut ditampilkan hasil visualisasi sebelum dan sesudah diterapkan Fill Mode pada Gambar 11 di bawah.



Gambar 11. Contoh Augmentasi *Fill Mode* Pada Citra Tumbler

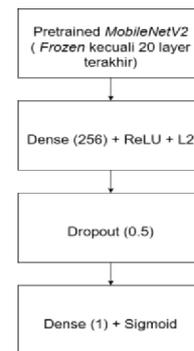
#### 4.3. Arsitektur Model

Dalam penelitian ini, digunakan pendekatan transfer learning dengan memanfaatkan arsitektur *MobileNetV2* yang telah dilatih sebelumnya (pretrained) pada dataset ImageNet. Model dimuat tanpa lapisan klasifikasi akhir bawaan (include\_top=False) dan dengan skema average pooling pada bagian akhir konvolusinya. Untuk memaksimalkan proses pembelajaran, hanya 20 lapisan terakhir dari *MobileNetV2* yang diaktifkan agar dapat diperbarui bobotnya (trainable=True), sementara lapisan lainnya dibekukan (frozen). Strategi ini bertujuan agar fitur-fitur umum yang telah dipelajari dari ImageNet tetap digunakan, namun model tetap mampu menyesuaikan diri dengan karakteristik data baru dalam penelitian ini.

Selanjutnya, model juga ditambahkan lapisan fully connected (dense) sebanyak dua buah dan satu lapisan dropout yaitu:

1. Dense (256, activation='relu') dengan regularisasi L2 untuk mencegah overfitting,
2. Dropout (0.5) untuk menonaktifkan 50% neuron secara acak selama training, juga sebagai bentuk regularisasi tambahan,
3. Dense(1, activation='sigmoid') sebagai lapisan Output karena tugas klasifikasi bersifat biner (antara tumbler dan botol).

Sehingga dapat kita sederhanakan bentuk akhir dari arsitektur model pada penelitian ini dapat dilihat pada Gambar 12 di bawah.



Gambar 12 Arsitektur Model

#### 4.4. Proses Pelatihan

Pelatihan model dilakukan melalui tiga tahap utama: compile, callback, dan fitting. Model dikompilasi menggunakan fungsi binary\_crossentropy sebagai loss function karena klasifikasi bersifat biner, dengan optimizer Adam dan learning rate sebesar 0.0001. Metrik evaluasi utama yang digunakan adalah akurasi.

Untuk menghindari overfitting dan menghemat waktu pelatihan, diterapkan mekanisme callback berupa early stopping kustom. Callback ini secara otomatis menghentikan proses pelatihan jika akurasi pelatihan dan validasi mencapai lebih dari 95%.

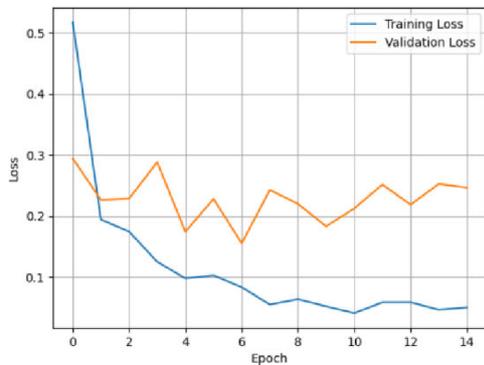


Gambar 13 Grafik Akurasi Tiap Epoch

Proses fitting dilakukan selama maksimal 25 epoch menggunakan fungsi model.fit(), dengan data yang telah dibagi sebelumnya untuk pelatihan dan validasi. Informasi historis pelatihan seperti nilai akurasi dan loss per epoch disimpan dalam objek history untuk keperluan visualisasi dan evaluasi kinerja model. Pada Gambar 13 menunjukkan perbandingan akurasi pada data pelatihan (training akurasi) dan data validasi (validasi akurasi) selama 30 epoch.

Training akurasi mengalami peningkatan secara konsisten hingga mencapai nilai di atas 0.85 pada akhir pelatihan. Sementara itu, validasi akurasi menunjukkan fluktuasi yang cukup signifikan, dengan nilai tertinggi berada di sekitar epoch ke-15, yaitu

Gambar 14 Grafik Loss Tiap Epoch



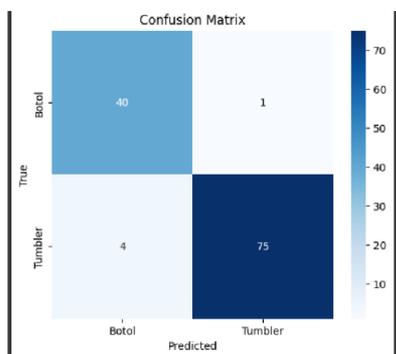
mendekati 0.80, kemudian mengalami penurunan dan naik-turun hingga akhir pelatihan. Secara keseluruhan, akurasi pelatihan lebih tinggi dibandingkan akurasi validasi pada sebagian besar epoch.

Gambar 14 menunjukkan grafik perbandingan loss pada data pelatihan (training loss) dan data validasi (validasi loss) selama 30 epoch. Terlihat bahwa nilai loss pada kedua data mengalami penurunan secara bertahap seiring bertambahnya epoch. Pada awal pelatihan, baik training loss maupun validasi loss berada pada kisaran 4.5 hingga 5.0. Seiring waktu, keduanya menurun dan mencapai nilai di bawah 2.0 pada akhir pelatihan. Meskipun terdapat sedikit fluktuasi pada validasi loss setelah epoch ke-15, secara umum pola penurunannya tetap terlihat. Training loss cenderung lebih rendah dibandingkan validasi loss di sebagian besar epoch.

#### 4.5. Evaluasi Kinerja Model

Evaluasi model dilakukan menggunakan confusion matrix dan metrik performa klasifikasi seperti precision, recall, f1-score, dan akurasi.

Gambar 15 Hasil Pengujian Confusion Matrix



Gambar 5 memperlihatkan confusion matrix hasil pengujian model terhadap 120 data uji. Model berhasil mengklasifikasikan:

- 40 dari 41 citra botol dengan benar (True Negative),
- 75 dari 79 citra tumbler dengan benar (True Positive),
- dengan 1 kesalahan klasifikasi botol menjadi tumbler (False Positive) dan

- kesalahan klasifikasi tumbler menjadi botol (False Negative).

Berdasarkan classification report (gambar 15), diperoleh metrik performa sebagai berikut:

- Akurasi keseluruhan: 96%
- Precision: 91% (botol), 99% (tumbler)
- Recall: 98% (botol), 95% (tumbler)
- F1-score: 0.94 (botol), 0.97 (tumbler)

Nilai-nilai tersebut menunjukkan bahwa model memiliki kemampuan klasifikasi yang tinggi dan seimbang antar kelas, dengan precision dan recall yang saling mendekati. Performa model terhadap kelas tumbler lebih stabil, sedangkan kesalahan pada tumbler yang diklasifikasikan sebagai botol kemungkinan disebabkan oleh kemiripan visual pada desain botol reusable tertentu.

### 5. KESIMPULAN DAN SARAN

Penelitian ini berhasil mengembangkan sistem klasifikasi citra untuk membedakan wadah minuman reusable (seperti tumbler berbahan PP, stainless steel, atau Tritan bebas BPA) dan non-reusable (seperti botol plastik sekali pakai berbahan PET) secara otomatis menggunakan arsitektur *MobileNetV2* dan pendekatan transfer learning. Model dilatih menggunakan dataset yang telah diproses dengan teknik normalisasi dan augmentasi, serta dievaluasi melalui metrik performa klasifikasi.

Hasil evaluasi menunjukkan bahwa model memiliki kinerja klasifikasi yang sangat baik, dengan akurasi 96%, precision 95%, recall 96%, dan f1-score 95%. Performa tersebut menandakan bahwa sistem mampu mengenali pola visual antar kelas dengan baik meskipun dataset terbatas.

Dengan arsitektur yang ringan dan efisien, model ini dapat diimplementasikan pada antarmuka sederhana berbasis web untuk memudahkan pengguna melakukan klasifikasi secara langsung. Ke depan, pengembangan sistem dapat diarahkan pada penambahan kelas, integrasi object detection, serta pengujian lebih lanjut dalam lingkungan nyata untuk meningkatkan akurasi dan relevansi sistem secara umum.

### DAFTAR PUSTAKA

- [1] D. L. Fithri, R. Setiawan, B. C. Wibowo, F. Nugraha, and N. Latifah, "Pengelolaan Bank Sampah Muria Berseri berbasis Digital Desa Gondangmanis Kabupaten Kudus," vol. 4, no. 1, pp. 51–58, 2024.
- [2] C. Ari Rahmayani, "Efektivitas Pengendalian Sampah Plastik Untuk Mendukung Kelestarian Lingkungan Hidup Di Kota Semarang," 2021.
- [3] N. Maulidah, R. Supriyadi, D. Y. Utami, F. N. Hasan, A. Fauzi, and A. Christian, "Prediksi Penyakit Diabetes Melitus Menggunakan Metode Support Vector Machine dan Naive Bayes," *Indonesian Journal on Software Engineering (IJSE)*, vol. 7, no. 1, pp. 63–68, 2021, [Online]. Available: <http://ejournal.bsi.ac.id/ejurnal/index.php/ijse63>

- [4] D. Y. Utami, E. Nurlelah, and F. N. Hasan, "Comparison of Neural Network Algorithms, Naive Bayes and Logistic Regression to predict diabetes," *JOURNAL OF INFORMATICS AND TELECOMMUNICATION ENGINEERING*, vol. 5, no. 1, pp. 53–64, Jul. 2021, doi: 10.31289/jite.v5i1.5201.
- [5] S. Agustiani, R. Aryanti, S. Khotimatul Wildah, Y. T. Arifin, S. Marlina, and T. Misriati, "Optimisasi Model Deep Learning untuk Deteksi Penyakit Daun Tebu dengan Fine-Tuning MobileNetV2," *Journal of Informatics Management and Information Technology*, vol. 4, no. 4, pp. 150–157, 2024, doi: 10.47065/jimat.v4i4.411.
- [6] U. Ramadhan *et al.*, "Deteksi Sampah Botol Plastik di Perairan Menggunakan YOLO v4-Tiny," vol. 7, no. 1, 2025, doi: 10.47233/jteksis.v5i1.1744.
- [7] Y. Ginting, K. Hantoro, A. Yunizar Pratama Yusuf, and U. Bhayangkara Jakarta Raya, "Deteksi Jenis Sampah Plastik Berbasis Mobile Menggunakan Model Transfer Learning," 2024, doi: 10.37817/tekinfo.v25i2.
- [8] D. A. Rismayadi, M. A. Muharam, F. I. Kreatif, D. Teknik Informatika, and U. T. Bandung, "PEMANFAATAN MACHINE LEARNING UNTUK OPTIMALISASI LIMBAH DENGAN MODEL MOBILENETV2 PADA APLIKASI ANDROID," vol. 06, 2024.
- [9] R. Theofilus and R. Kurniawan, "Deteksi Sampah di Permukaan Sungai menggunakan Convolutional Neural Network dengan Algoritma YOLOv8 Studi Kasus: Sungai Ciliwung," 2024.
- [10] R. Muhammad, V. Pramudika, and M. Hablul Barri, "Sistem Pemilahan Sampah Berbasis Deep Learning dengan Algoritma SSD-MobileNet v2," 2024.
- [11] T. Sutisna, A. Rachmat Raharja, E. Hariyadi, and V. Hafizh Cahaya Putra, "Penggunaan Computer Vision untuk Menghitung Jumlah Kendaraan dengan Menggunakan Metode SSD (Single Shoot Detector)," *INNOVATIVE: Journal Of Social Science Research*, vol. 4, pp. 6060–6067, 2024.
- [12] Rohman Dijaya, *Buku Ajar Pengolahan Citra Digital*. 2023.
- [13] O. Saputra, D. Iskandar Mulyana, and M. B. Yel, "Implementasi Algoritma Convolutional Neural Network (CNN) Untuk Klasifikasi Senjata Tradisional Di Jawa Tengah Dengan Metode Transfer Learning," 2022.
- [14] N. Hikmatia and M. I. Zul, "Aplikasi Penerjemah Bahasa Isyarat Indonesia menjadi Suara berbasis Android menggunakan Tensorflow," 2021. [Online]. Available: <https://jurnal.pcr.ac.id/index.php/jkt/>
- [15] K. Azmi, S. Defit, and U. Putra Indonesia YPTK Padang Jl Raya Lubuk Begalung-Padang-Sumatera Barat, "Implementasi Convolutional Neural Network (CNN) Untuk Klasifikasi Batik Tanah Liat Sumatera Barat," vol. 16, no. 1, 2023.

## BIODATA PENULIS



### Penulis Pertama

Dea Ramanda merupakan Mahasiswa di Universitas Bina Sarana Informatika, Jurusan Informatika.



### Penulis Kedua

Fuad Nur Hasan, M.Kom merupakan Dosen di Universitas Bina Sarana Informatika.



### Penulis Ketiga

Antonius Yadi Kuntoro, M.M, M.Kom Merupakan Dosen di Universitas Bina Sarana Informatika.