

# Perbandingan Akurasi Arsitektur MobileNet dan EfficientNet dalam Mendeteksi Produk Kemasan

<sup>1\*</sup>**Lu'ay Baehaqie, <sup>2</sup>Daniel Swanjaya, <sup>3</sup>Danar Putra Pamungkas**

<sup>123</sup> Teknik Informatika, Universitas Nusantara PGRI Kediri

E-mail: \*[baehaqie820@gmail.com](mailto:baehaqie820@gmail.com), <sup>2</sup>[daniel@unpkediri.ac.id](mailto:daniel@unpkediri.ac.id), <sup>3</sup>[danar@unpkediri.ac.id](mailto:danar@unpkediri.ac.id)

*Penulis Korespondens : Lu'ay Baehaqie*

**Abstrak**— Penelitian ini membahas perbandingan performa dua arsitektur *deep learning*, yaitu *MobileNet* dan *EfficientNet*, dalam mendeteksi produk kemasan berbasis citra. Dataset yang digunakan terdiri dari 150 gambar tiga jenis produk kemasan, yaitu Le Minerale, Isoplus, dan Sprite, yang telah melalui proses anotasi dan dibagi menjadi data latih, validasi, dan uji. Proses pelatihan dilakukan selama 10 *epoch* dengan ukuran citra 448x448 piksel. Evaluasi dilakukan menggunakan metrik *accuracy*, *precision*, *recall*, dan *f1-score*. Hasil penelitian menunjukkan bahwa *MobileNet* memiliki performa sangat baik dengan akurasi 100% pada ketiga kelas produk, serta nilai *precision*, *recall*, dan *f1-score* sebesar 1.00 pada semua kategori. Sementara itu, *EfficientNet* menunjukkan hasil yang kurang optimal dengan akurasi hanya 33%, disertai bias prediksi terhadap satu kelas saja. Berdasarkan hasil tersebut, *MobileNet* direkomendasikan sebagai arsitektur yang lebih efisien dan andal dalam kasus deteksi produk kemasan berdasarkan dari hasil dan penelitian yang dilakukan.

**Kata Kunci**— Deep Learning, EfficientNet, MobileNet, Object Detection, Produk Kemasan

**Abstract**— This study discusses the performance comparison of two deep learning architectures, *MobileNet* and *EfficientNet*, in detecting packaged products based on image classification. The dataset consists of 150 images of three types of packaged products: Le Minerale, Isoplus, and Sprite, which were annotated and divided into training, validation, and testing sets. The models were trained for 10 epochs using 448x448 pixel image sizes. Evaluation was conducted using accuracy, precision, recall, and f1-score metrics. The results showed that *MobileNet* achieved excellent performance with 100% accuracy on all three product classes, and precision, recall, and f1-score values of 1.00 across all categories. On the other hand, *EfficientNet* produced suboptimal results with only 33% accuracy and a strong prediction bias toward a single class. Based on these findings, *MobileNet* is recommended as a more efficient and reliable architecture for packaged product detection tasks based on study that have been did.

**Keywords**— Deep Learning, EfficientNet, MobileNet, Object Detection, Packaged Products

This is an open access article under the CC BY-SA License.



## I. PENDAHULUAN

Bidang keilmuan komputer merupakan salah satu disiplin ilmu yang sangat luas dan terus berkembang seiring kemajuan teknologi. Salah satu cabang yang saat ini sangat populer dan banyak digunakan dalam berbagai aplikasi adalah *deep learning*. *Deep learning* merupakan teknologi yang memungkinkan komputer untuk belajar secara mandiri dari data tanpa harus diprogram secara eksplisit oleh manusia[1]. Proses pembelajaran ini didasarkan pada analisis data yang besar dan kompleks dengan menggunakan algoritma yang telah dikembangkan. Salah satu

teknik dalam *deep learning* adalah deteksi objek yang digunakan untuk mengidentifikasi dan mendeteksi objek dalam citra atau video[2].

Deteksi objek adalah salah satu bagian dari *deep learning* yang terus berkembang dengan berbagai arsitektur baru untuk meningkatkan akurasi dan efisiensi[3]. Dua arsitektur yang sering digunakan saat ini adalah *MobileNetV2* dan *EfficientNet*. Keduanya memiliki cara berbeda dalam menyeimbangkan antara kinerja dan penggunaan sumber daya. *MobileNetV2* dibuat khusus untuk perangkat dengan kemampuan terbatas, seperti ponsel atau perangkat *IoT*, dengan menggunakan teknik khusus seperti *depthwise separable convolution* dan *inverted residual blocks*. Teknik ini membuat jumlah parameter yang digunakan menjadi lebih sedikit, sekitar 30% lebih kecil dari versi sebelumnya, namun tetap menjaga akurasi yang baik. Sementara itu, *EfficientNet* menggunakan pendekatan yang disebut *compound scaling*, yaitu menyesuaikan kedalaman, lebar, dan resolusi model secara bersamaan agar bisa mendapatkan akurasi tinggi tanpa membuang-buang sumber daya komputasi[4].

*MobileNet* dan *EfficientNet* adalah dua arsitektur jaringan saraf konvolisional yang sering digunakan untuk tugas deteksi objek, namun keduanya memiliki keunggulan dan karakteristik yang berbeda. *MobileNet* khususnya dirancang agar ringan dan efisien sehingga sangat cocok digunakan pada perangkat dengan sumber daya terbatas. Arsitektur ini menggunakan teknik *depthwise separable convolution* dan *inverted residual blocks* yang membuat jumlah parameter dan beban komputasi menjadi sangat kecil, tanpa mengorbankan akurasi secara signifikan. Dalam berbagai penelitian, *MobileNet* terbukti sangat cepat dalam proses inferensi dan membutuhkan daya komputasi yang rendah, sehingga sangat efisien untuk aplikasi *real-time* atau pada perangkat dengan keterbatasan hardware[5].

Berdasarkan perkembangan ilmu komputer dan kemajuan teknologi *deep learning*, khususnya dalam bidang *object detection*, dapat disimpulkan bahwa teknologi ini memegang peranan sangat penting dalam berbagai aplikasi modern. *Object detection* tidak hanya memungkinkan identifikasi dan pengenalan objek dalam citra atau video secara akurat, tetapi juga memberikan kontribusi besar dalam meningkatkan efisiensi otomatisasi proses di berbagai sektor industri, seperti klasifikasi produk botol kemasan. Dengan dukungan arsitektur yang beragam seperti *MobileNet* dan *EfficientNet*. Selain itu, kemampuan *deep learning* dalam mengolah data skala besar dan melakukan automasi sistem membuka peluang besar untuk inovasi yang dapat mempercepat operasional dan meningkatkan kualitas produk. Oleh karena itu, *object detection* berbasis *deep learning* merupakan inovasi yang dapat mendorong kemajuan teknologi dan transformasi digital yang berkelanjutan di berbagai bidang industri dan kehidupan sehari-hari[6].

## II. METODE

### A. Studi Literatur

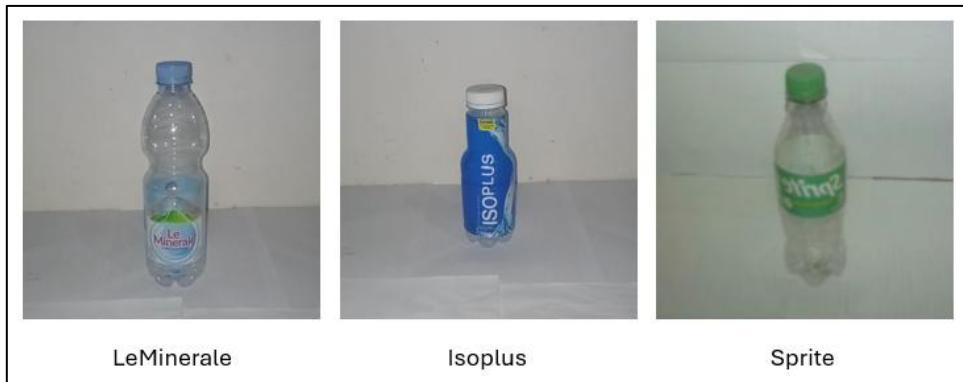
Tahap awal dilakukan dengan membaca berbagai referensi ilmiah terkait deteksi dan klasifikasi produk kemasan berbasis *deep learning*. Penelitian sebelumnya yang menggunakan *MobileNet*, *EfficientNet*, dan metode evaluasi seperti *mAp* (*mean Average Precision*), *accuracy*, *precision*, dan *recall* menjadi dasar dalam menyusun strategi pelatihan serta pengujian model baik dari segi efisiensi komputasi, kecepatan inferensi, maupun tingkat akurasi yang dihasilkan. Studi literatur juga dijadikan acuan dalam melakukan penelitian dengan membandingkan dua arsitektur tersebut, dengan hal itu penulis dapat membuat penelitian mengenai perbandingan dari dua arsitektur antara *MobileNet* dan *EfficientNet*.

## B. Pengumpulan Data

Secara keseluruhan, jumlah dataset yang digunakan dalam penelitian ini adalah 150 gambar. Seluruh gambar telah dianotasi kemudian dilakukan proses pelatihan data menggunakan dua arsitektur antara *MobileNet* dan *EfficientNet* sehingga dapat dijadikan perbandingan oleh penulis. Dataset yang digunakan dalam penelitian ini berasal dari foto produk dengan berbagai sisi guna untuk memudahkan data latih. Proses anotasi dan pelabelan dilakukan dengan membagi tiap kelas ke dalam satu folder , data yang telah dilabeli kemudian dibagi menjadi tiga bagian:

Tabel 1. Pembagian Dataset

Folder	Jumlah
Train	105
Test	24
Valid	21



Gambar 1. Contoh Dataset

## C. PreProcessing Data

Pada tahap ini, dilakukan beberapa proses penting seperti *resizing*, *augmentasi*, dan normalisasi untuk menyiapkan data sebelum digunakan dalam pelatihan model. Pertama, seluruh gambar diubah ukurannya menjadi 448x448 piksel agar memiliki dimensi yang seragam, sesuai dengan kebutuhan arsitektur model. Dengan cara ini, model dapat belajar mengenali keunikan setiap kemasan produk yang memiliki berbagai bentuk dan warna yang beragam dan lebih tahan terhadap *overfitting*.

## D. Arsitektur

Dalam penelitian ini digunakan dua model arsitektur yaitu *MobileNet* dan *EfficientNet*.

### 1. MobileNet

*MobileNet* merupakan salah satu arsitektur jaringan saraf konvolusional (CNN) yang dirancang khusus untuk aplikasi dengan keterbatasan sumber daya komputasi, seperti perangkat *mobile* dan *edge device*[7]. Arsitektur ini menggunakan teknik *depthwise separable convolution* yang memisahkan proses konvolusi menjadi dua tahap, sehingga mengurangi jumlah parameter dan beban komputasi secara signifikan tanpa mengorbankan akurasi. Dalam konteks deteksi objek, *MobileNet* sering dipadukan dengan arsitektur *Single Shot MultiBox Detector* (SSD) menjadi *MobileNet*, yang

memungkinkan deteksi objek secara real-time dengan kecepatan tinggi dan akurasi yang cukup baik[8].

## 2. *EfficientNet*

*EfficientNet* adalah arsitektur *convolutional neural network* (CNN) yang dikembangkan oleh Mingxing Tan dan Quoc V. Le pada tahun 2019 dengan tujuan menciptakan model yang efisien dalam penggunaan sumber daya komputasi sekaligus memiliki akurasi tinggi dalam tugas pengolahan citra, termasuk deteksi objek produk kemasan[9]. Arsitektur ini menggunakan pendekatan *compound scaling*, yaitu penskalaan serentak pada tiga dimensi utama model kedalaman (*depth*), lebar (*width*), dan resolusi input sehingga model dapat dioptimalkan secara proporsional untuk mencapai performa terbaik tanpa pemborosan komputasi. *EfficientNet* juga mengadaptasi blok MBCConv (*Mobile Inverted Bottleneck Convolution*) yang sebelumnya digunakan pada *MobileNetV2*, yang membantu mengurangi jumlah parameter dan mempercepat proses pelatihan tanpa mengurangi kemampuan representasi fitur[10].

## E. Evaluasi

Pada proses evaluasi diukur tingkat akurasi dalam mendekripsi produk kemasan. Untuk mengukur tingkat akurasi deteksi objek, maka akan dihitung berapa banyak jumlah objek produk yang terdeteksi oleh sistem. Berikut merupakan parameter-parameter yang digunakan dalam pengukuran tingkat akurasi deteksi :

### a) *Accuracy*

*Accuracy* adalah hasil perhitungan tingkat akurasi *object detection* terhadap objek produk pada keseluruhan frame yang dilakukan proses deteksi. Rumus dari perhitungan *accuracy* dapat dilihat pada Algoritma (1).

$$\text{Accuracy} = \frac{\text{Jumlah Objek Yang Terdeteksi Benar}}{\text{Jumlah Keseluruhan Objek Yang Terdeteksi}} \quad (1)$$

### b) *Precision*

*Precision* adalah jumlah prediksi yang benar dibandingkan keseluruhan hasil yang terprediksi oleh sistem. Dalam hal ini *precision* akan menjawab berapa jumlah objek yang terdeteksi benar dari keseluruhan jumlah objek yang terdeteksi oleh sistem. Rumus dari perhitungan *precision* dapat dilihat pada (2)

$$\text{Precision} = \frac{TP}{(TP+FP)} \quad (2)$$

### c) *Recall*

*Recall* adalah jumlah prediksi yang benar dibandingkan dengan keseluruhan hasil yang sebenarnya. Dalam hal ini *recall* akan menjawab berapa jumlah objek yang terdeteksi benar dari keseluruhan jumlah objek sebenarnya. Rumus dari perhitungan *recall* dapat dilihat pada (3)

$$\text{Recall} = \frac{TP}{(TP+FP)} \quad (3)$$

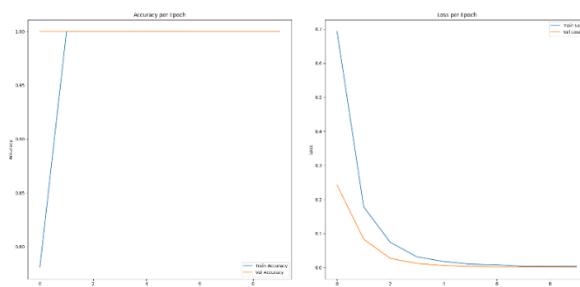
Melalui proses ini, dapat diketahui kelebihan dan kekurangan antara kedua model, serta dijadikan acuan untuk melakukan penyempurnaan, seperti peningkatan data, parameter pelatihan, atau penggantian strategi pelatihan.

### III. HASIL DAN PEMBAHASAN

Berdasarkan percobaan yang dilakukan oleh penulis dalam penggunaan dua arsitektur antara arsitektur *MobileNet* dan *EfficientNet* memberikan hasil perbandingan yang signifikan dalam mendeteksi produk kemasan, untuk menguraikan hasil yang sudah penulis lakukan maka rincian prosesnya sebagai berikut :

#### A. Training Model

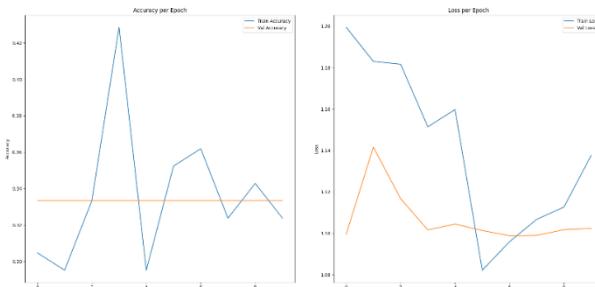
Proses *training* model dalam penelitian ini menggunakan dua arsitektur yaitu *MobileNet* dan *EfficientNet*, model dilatih menggunakan ukuran gambar sebesar 448x448 pixel. Proses pelatihan dilakukan selama 10 epoch dengan batch size sebesar 32. Pada Gambar 2 merupakan hasil *Training Loop* dari arsitektur *MobileNet*, grafik yang ditampilkan memberikan hasil yang bagus dimana *Accuracy per Epoch* meningkat dan stabil di angka 100%, sedangkan *Loss per Epoch* semakin turun pada akhir epoch dengan angka 0%. Hal ini memaparkan bahwa arsitektur *MobileNet* dapat mendeteksi objek dan belajar dari data dengan sangat akurat.



Gambar 2. Grafik *Training Loop* *MobileNet*

Terlihat pada Gambar 3 bahwa akurasi pada data pelatihan mengalami fluktuasi yang cukup besar, naik turun secara tidak stabil, sementara akurasi pada data validasi cenderung stagnan di sekitar angka 0,33 tanpa peningkatan yang signifikan. Hal ini memberikan kesimpulan dimana hasil *Training Data* dari arsitektur *EfficientNet* tidak stabil dalam mendeteksi produk kemasan. Hal tersebut juga didukung oleh grafik *Loss per Epoch* yang menampilkan nilai loss pada data pelatihan *Train Loss* dan data validasi *Val Loss per Epoch*. Nilai loss pada data pelatihan juga tampak tidak stabil, dengan penurunan tajam di tengah epoch namun kemudian kembali meningkat, sedangkan loss pada data validasi relatif stabil dan tidak menunjukkan penurunan.

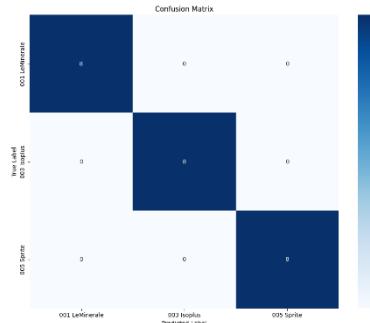
Pola ini mengindikasikan bahwa model mengalami kesulitan dalam belajar dari data, kemungkinan karena masalah pada arsitektur model, parameter pelatihan, atau kualitas data, sehingga belum mampu meningkatkan performa pada data validasi secara optimal.



Gambar 3. Grafik *Training Loop* *EfficientNet*

Hasil dari *Confusion Matrix* dari arsitektur *MobileNet* dapat dilihat pada Gambar 4 yang digunakan untuk mengevaluasi performa model klasifikasi dalam mendekripsi tiga jenis produk kemasan, yaitu 001 LeMinerale, 003 Isoplus, dan 005 Sprite. Setiap baris pada matriks menunjukkan *true label* dari data uji, sedangkan setiap kolom menunjukkan label hasil *predicted label*. Nilai diagonal utama, yaitu angka 8 pada masing-masing kelas, menunjukkan bahwa seluruh data uji untuk ketiga produk tersebut berhasil diklasifikasikan dengan benar oleh model. Tidak terdapat nilai di luar diagonal, yang berarti tidak ada data yang salah klasifikasi ke kelas lain.

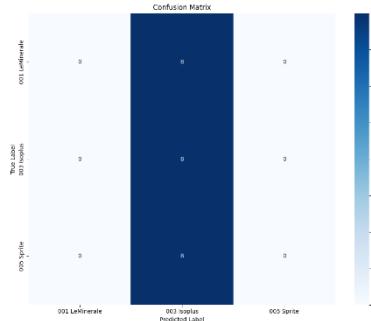
Dengan demikian, *confusion matrix* ini menunjukkan bahwa model memiliki akurasi sempurna dalam mengklasifikasikan ketiga jenis produk kemasan pada data uji yang digunakan.



Gambar 4. *Confusion Matrix MobileNet*

Kemudian pada Gambar 5 yang ditampilkan merupakan *confusion matrix* hasil evaluasi model klasifikasi arsitektur *EfficientNet* yang dirancang untuk mengklasifikasikan tiga jenis produk, yaitu 001 LeMinerale, 003 Isoplus, dan 005 Sprite. Setiap baris pada *confusion matrix* ini menunjukkan *true label* pada data uji, setiap kolom menunjukkan label hasil *predict label*. Pada hasil yang terlihat, seluruh data uji diprediksi sebagai 003 Isoplus. Dengan kata lain, baik sampel yang sebenarnya adalah 001 LeMinerale maupun 005 Sprite, semuanya salah diprediksi sebagai 003 Isoplus. Satu-satunya prediksi yang benar hanya pada kelas 003 Isoplus itu sendiri, namun prediksi ini tidak memberikan nilai tambah karena model gagal mengenali dua kelas lainnya.

Kondisi ini mengindikasikan bahwa model mengalami bias yang sangat kuat terhadap kelas 003 Isoplus, sehingga tidak mampu membedakan ketiga kategori produk yang ada. Secara keseluruhan, performa model ini sangat buruk karena tidak dapat mengenali dua dari tiga kelas produk, sehingga evaluasi dan perbaikan terhadap data maupun model sangat diperlukan agar model dapat berfungsi dengan baik dalam melakukan deteksi pada produk kemasan.



Gambar 5. *Confusion Matrix EfficientNet*

Nilai *classification report* yang menampilkan hasil evaluasi performa model dari arsitektur *MobileNet* terhadap tiga jenis produk kemasan, yaitu 001 LeMinerale, 003 Isoplus, dan 005 Sprite dapat dilihat pada tabel 2. Setiap kelas memperoleh nilai *precision*, *recall*, dan *f1-score* sebesar 1,00, yang berarti model mampu mengklasifikasikan seluruh sampel dengan benar tanpa ada kesalahan. Nilai *support* pada masing-masing kelas adalah 8, menunjukkan bahwa setiap kelas diuji dengan 8 sampel.

Tabel 2. *Classification Report MobileNet*

Kelas	Precision	Recall	F1-Score	Support
001 LeMinerale	1.00	1.00	1.00	8
003 Isoplus	1.00	1.00	1.00	8
005 Sprite	1.00	1.00	1.00	8

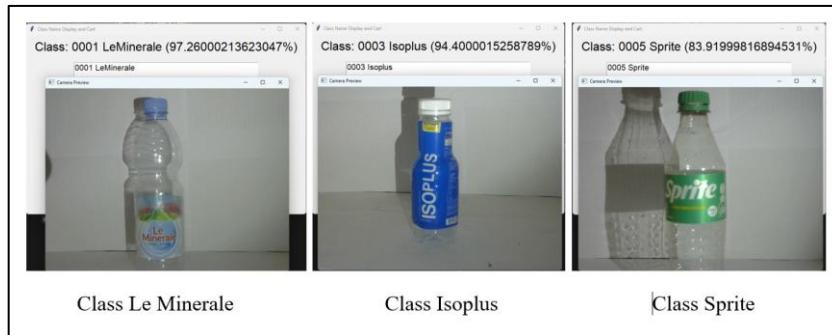
Sedangkan pada Tabel 3 merupakan hasil dari *classification report* dari arsitektur *EfficientNet* yang menampilkan hasil evaluasi kinerja model klasifikasi terhadap tiga produk kemasan 001 LeMinerale, 003 Isoplus, dan 005 Sprite. Nilai *precision*, *recall*, dan *f1-score* untuk kelas 001 LeMinerale dan 005 Sprite pada Tabel 3 adalah 0,00, yang berarti model sama sekali tidak mampu mengidentifikasi kedua kelas tersebut dengan benar. Sebaliknya, pada kelas 003 Isoplus, model memiliki *recall* sebesar 1,00, namun *precision* hanya 0,33 dan *f1-score* 0,50, menandakan bahwa model selalu memprediksi semua sampel sebagai 003 Isoplus, sehingga hanya benar ketika sampel tersebut memang benar-benar 003 Isoplus. Akurasi keseluruhan model hanya 0,33 atau 33%, ini menunjukkan bahwa model mengalami bias yang kuat terhadap kelas 003 Isoplus dan gagal mengenali dua kelas lainnya.

Tabel 3. Classification Report EfficientNet

Kelas	Precision	Recall	F1-Score	Support
001 LeMinerale	0.00	0.00	0.00	8
003 Isoplus	0.33	1.00	0.50	8
005 Sprite	0.00	0.00	0.00	8

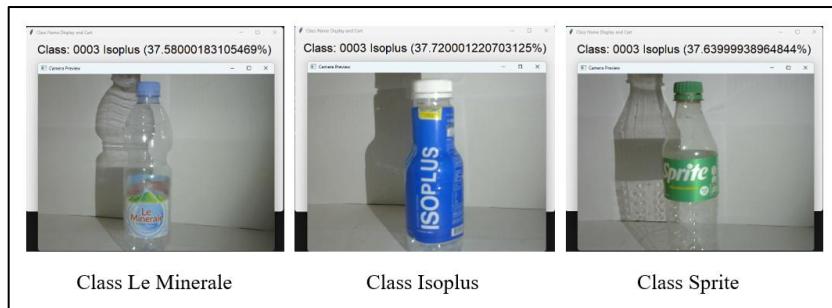
## B. Pengujian Model

Hasil deteksi yang dilakukan menggunakan model dari arsitektur *MobileNet* dapat dilihat pada Gambar 6 pada bagian atas masing-masing gambar, tercantum nama kelas hasil prediksi, seperti "Class: 0001 LeMinerale", "Class: 0003 Isoplus", dan "Class: 0005 Sprite", diikuti angka yang menunjukkan persentase keyakinan model terhadap prediksi tersebut. Sesuai dengan hasil dari *Confussion Matrix* yang sebelumnya sudah dilakukan maka dari ketiga produk kemasan semua kelasnya dapat terdeteksi dengan akurat dimana nilai *Confidence* setiap kelas berada diatas 80%, dengan hasil tersebut maka model sudah dikatakan cukup mumpuni untuk diimplementasikan.



Gambar 6. Hasil Pengujian *MobileNet*

Sedangkan hasil hasil deteksi yang dilakukan menggunakan model dari arsitektur *EfficientNet* dapat dilihat pada Gambar 7 dengan hasil prediksi model klasifikasi citra terhadap tiga jenis minuman, yaitu Le Minerale, Isoplus, dan Sprite. Masing-masing gambar menampilkan botol minuman dari kelas yang berbeda, namun model memberikan hasil prediksi yang sama untuk ketiganya. Botol Le Minerale dan Sprite yang seharusnya termasuk dalam kelas 001Le Minerale dan 005 Sprite justru diprediksi sebagai kelas 003 Isoplus. Hanya botol Isoplus yang diprediksi dengan benar oleh model sebagai kelas 003 Isoplus. Namun demikian, tingkat kepercayaan (*confidence*) model pada ketiga gambar tersebut tergolong rendah, yaitu sekitar 37%. Hal ini menunjukkan bahwa model mengalami bias yang cukup besar terhadap kelas Isoplus dan tidak mampu mengenali ciri khas dari dua kelas lainnya. Temuan ini juga konsisten dengan *confusion matrix* sebelumnya yang memperlihatkan bahwa semua data uji, tanpa memandang label aslinya, diprediksi sebagai kelas Isoplus.



Gambar 7. Hasil Pengujian *EfficientNet*

#### IV. KESIMPULAN

Berdasarkan hasil penelitian yang telah dilakukan, dapat disimpulkan bahwa pemanfaatan arsitektur *MobileNet* dan *EfficientNet* dalam mendeteksi produk kemasan memberikan performa yang berbeda secara signifikan. *MobileNet* menunjukkan hasil yang sangat baik, dengan *accuracy*, *precision*, *recall*, dan *f1-score* sebesar 100% pada seluruh kelas yang diuji, serta konsistensi pada hasil training dan pengujian. Hal ini membuktikan bahwa *MobileNet* mampu mengklasifikasikan objek produk kemasan secara tepat dan efisien, serta cocok untuk diterapkan pada sistem dengan keterbatasan sumber daya komputasi. Sebaliknya, *EfficientNet* menunjukkan performa yang kurang optimal, dengan *accuracy* hanya sebesar 33% dan kecenderungan bias kuat terhadap satu kelas tertentu, yaitu Isoplus. Model ini gagal dalam mengenali dua kelas lainnya, yaitu Le Minerale dan Sprite, serta memiliki tingkat *confidence* yang rendah pada proses prediksi.

Temuan ini mengindikasikan bahwa meskipun *EfficientNet* dikenal memiliki potensi akurasi tinggi, implementasinya dalam konteks dataset dan parameter pelatihan pada penelitian ini belum memberikan hasil yang memuaskan. Oleh karena itu, *MobileNet* dapat direkomendasikan sebagai arsitektur yang lebih andal dan efisien untuk sistem deteksi produk kemasan dalam skenario nyata, sedangkan pengembangan lebih lanjut masih diperlukan untuk meningkatkan kinerja *EfficientNet*, baik dari sisi data, arsitektur, maupun teknik pelatihan.

## DAFTAR PUSTAKA

- [1] E. Sihabudin, R. Kumalasari Niswatin, and L. S. Wahyuniar, “Penerjemah Bahasa Isyarat Menggunakan Tensorflow,” *Seminar Nasional Inovasi Teknologi*, Jul. 2022, doi: <https://doi.org/10.29407/inotek.v6i2.2576>.
- [2] R. Fajri and F. Fitria, “KLICK: Kajian Ilmiah Informatika dan Komputer Pengembangan Real-Time Object Detection System pada Perangkat Single-Board Computer,” *Media Online*), vol. 4, no. 2, pp. 1154–1162, 2023, doi: 10.30865/klck.v4i2.1224.
- [3] S. Arifin, “Klasifikasi Motif Batik Menggunakan Metode Convolutional Neural Network (CNN) Dengan Multi Class Clasification,” *Jurnal Ilmiah IT CIDA : Diseminasi Teknologi Informasi*, vol. 10, no. 1, 2024, doi: <https://doi.org/10.55635/jic.v10i1.206>.
- [4] Y. Miftahuddin and F. Zaelani, “Perbandingan Metode Efficientnet-B3 dan Mobilenet-V2 Untuk Identifikasi Jenis Buah-buahan Menggunakan Fitur Daun,” 2022. doi: <http://dx.doi.org/10.33197/jitter.vol9.iss1.2022.911>.
- [5] S. A. F. Harahap and I. Irmawan, “Performance Comparison of MobileNet, EfficientNet, and Inception for Predicting Crop Disease,” *Sriwijaya Electrical and Computer Engineering Journal*, vol. 1, no. 1, pp. 30–36, Feb. 2024, doi: <https://doi.org/10.62420/selco.v1i1.4>.
- [6] Albertus Adrian Susanto, “Implementasi Deteksi Objek Secara Real-Time,” Yogyakarta, Jul. 2023. doi: <http://repository.ukdw.ac.id/id/eprint/8058>.
- [7] B. Khasoggi, Ermatita, and Samsuryadi, “Efficient mobilenet architecture as image recognition on mobile and embedded devices,” *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 16, no. 1, pp. 389–394, Oct. 2019, doi: 10.11591/ijeecs.v16.i1.pp389-394.
- [8] T. Turahman, E. Hasmin, and K. Aryasa, “Analisis Perbandingan Metode Convolutional Neural Network (CNN) dan MobileNet dalam Klasifikasi Penyakit Daun Padi.” [Online]. Available: <https://www.kaggle.com/datasets/vbookshelf/rice-disease-classification>
- [9] M. Iqbal Burhanuddin, A. Syaifulah, S. Adeka Putra Jaya, M. Gabriel Somoal, and U. Muhammadiyah HAMKA, “Analisis Komparatif Model MobilenetV1 Dan EfficientnetB0 Dalam Klasifikasi Citra Empat Musim Menggunakan Transfer Learning,” 2025. doi: <https://doi.org/10.58794/jekin.v5i2.1378>.
- [10] Aryo Hardirega, Irsan Jaelani, and Minarto, “Implementasi Convolutional Neural Network (CNN) Klasifikasi Motif Batik Menggunakan Efficientnet-B1,” Kab. Purwakarta, Oct. 2024. doi: <http://dx.doi.org/10.36040/jati.v8i5.10901>.