

# Analisis Komparatif Arsitektur *Deep Learning* untuk Klasifikasi Citra Rempah-Rempah Indonesia

## *Comparative Analysis of Deep Learning Architectures for Indonesian Spice Image Classification*

<sup>1</sup>Fatma Meylinda Putri\*, <sup>2</sup>Muhammad Imam Ghozali, <sup>3</sup>Wibowo Harry Sugiharto

<sup>1,2,3</sup>Program Studi Teknik Informatika, Fakultas Teknik, Universitas Muria Kudus

<sup>1,2,3</sup>Jl. Lkr. Utara, Kayuapu Kulon, Gondangmanis, Kec. Bae, Kabupaten Kudus, Jawa Tengah 59327  
Indonesia

\*e-mail: [202251179@std.umk.ac.id](mailto:202251179@std.umk.ac.id)

(received: 15 December 2025, revised: 31 January 2026, accepted: 2 February 2026)

### Abstrak

Rempah-rempah merupakan komoditas penting di Indonesia, namun identifikasi visualnya masih menjadi tantangan karena kemiripan bentuk antar jenis rempah. Penelitian ini mengembangkan sistem klasifikasi citra rempah menggunakan *transfer learning* dengan membandingkan empat arsitektur Convolutional Neural Network (CNN), VGG16, ResNet50, EfficientNetB0, dan MobileNetV2. Dataset Indonesian Spices terdiri dari 31 kelas dengan total 6.510 citra yang dibagi secara *stratified* ke dalam data latih, validasi, dan uji. Proses pelatihan dilakukan melalui dua tahap: *head layer training* dan *fine-tuning*, dengan penerapan regularisasi seperti *dropout*, *batch normalization*, dan L2 regularization. Hasil menunjukkan bahwa ResNet50 memberikan performa terbaik dengan akurasi uji 95,80%, diikuti VGG16 dengan 95,70%. EfficientNetB0 memberikan keseimbangan optimal antara akurasi (94,17%) dan waktu inferensi tercepat (5,51 ms), sedangkan MobileNetV2 mencatat waktu *inference* 6,07 ms dengan akurasi 92,63% yang cocok untuk perangkat *mobile*. Penelitian ini membuktikan efektivitas *transfer learning* dalam klasifikasi rempah-rempah Indonesia.

**Kata kunci:** klasifikasi citra, rempah Indonesia, *transfer learning*, *convolutional neural network*, *deep learning*

### Abstract

Spices are an important commodity in Indonesia; however, visual identification remains challenging due to the similarity in appearance among different types of spices. This study develops an image classification system for Indonesian spices using transfer learning by comparing four Convolutional Neural Network (CNN) architectures: VGG16, ResNet50, EfficientNetB0, and MobileNetV2. The Indonesian Spices dataset consists of 31 classes with a total of 6,510 images, which were stratified and divided into training, validation, and testing sets. The training process was conducted in two stages: head-layer training and fine-tuning, with the application of regularization techniques such as dropout, batch normalization, and L2 regularization. The results show that ResNet50 achieved the best performance with a test accuracy of 95.80%, followed by VGG16 with 95.70%. EfficientNetB0 provided an optimal balance between accuracy (94.17%) and the fastest inference time (5.51 ms), while MobileNetV2 achieved an inference time of 6.07 ms with an accuracy of 92.63%, making it suitable for mobile devices. This study demonstrates the effectiveness of transfer learning for Indonesian spice image classification.

**Keywords:** image classification, Indonesian spices, transfer learning, convolutional neural network, deep learning

## 1 Pendahuluan

Indonesia merupakan salah satu negara penghasil rempah-rempah terbesar di dunia dengan tingkat keanekaragaman hayati yang tinggi, serta memiliki nilai ekonomi penting di sektor kuliner, kesehatan, dan perdagangan [1]. Namun, proses identifikasi visual rempah-rempah masih menjadi tantangan bagi masyarakat umum, pedagang pemula, maupun peneliti karena adanya kemiripan

bentuk, tekstur, dan warna antar jenis rempah tertentu [2]. Kondisi ini menunjukkan perlunya sistem berbasis teknologi yang mampu membantu proses identifikasi secara lebih cepat, konsisten, dan akurat.

Perkembangan teknologi *deep learning*, khususnya Convolutional Neural Network (CNN), telah membuka peluang besar dalam pengembangan sistem identifikasi melalui sistem klasifikasi citra [3]. CNN terbukti efektif dalam mengekstraksi fitur visual dan melakukan pengenalan objek pada berbagai domain. Selain itu, pendekatan *transfer learning* yang memanfaatkan model pra-latih pada dataset besar seperti ImageNet dapat meningkatkan performa model, terutama pada kondisi jumlah data latih yang terbatas [4]. Dengan memanfaatkan keunggulan tersebut, sistem klasifikasi citra berbasis CNN berpotensi memberikan solusi efektif untuk mengatasi permasalahan identifikasi rempah-rempah.

Penerapan teknologi klasifikasi citra pada rempah-rempah memiliki manfaat yang luas. Sistem ini dapat membantu pedagang dan pelaku usaha dalam mengidentifikasi jenis rempah secara cepat dan konsisten, mendukung proses sortasi dan inventarisasi produk, serta mempermudah konsumen maupun masyarakat umum dalam mengenali jenis rempah yang memiliki kemiripan visual. Selain itu, sistem klasifikasi citra juga dapat dimanfaatkan sebagai media pembelajaran dan pendukung penelitian dalam bidang botani maupun pertanian. Dengan demikian, teknologi berbasis CNN tidak hanya memberikan solusi teknis, tetapi juga meningkatkan pemahaman dan akses informasi terkait keragaman rempah-rempah Indonesia.

Beberapa penelitian sebelumnya telah menggunakan CNN untuk klasifikasi produk pertanian maupun rempah-rempah dan menunjukkan bahwa model berbasis *deep learning* mampu menghasilkan performa yang lebih baik dibandingkan metode *machine learning* tradisional [5], [6], [7]. Namun, sebagian penelitian masih terbatas pada jumlah kelas yang relatif sedikit, belum melakukan perbandingan menyeluruh terhadap berbagai arsitektur CNN, atau belum mempertimbangkan aspek efisiensi komputasi, khususnya dari sisi waktu inferensi, yang penting untuk implementasi pada perangkat *mobile* atau aplikasi berbasis *web* [8].

Berdasarkan analisis kesenjangan penelitian yang telah diidentifikasi, penelitian ini dirumuskan untuk menjawab beberapa pertanyaan penelitian. Pertama, bagaimana performa komparatif arsitektur CNN populer (VGG16, ResNet50, EfficientNetB0, dan MobileNetV2) dalam mengklasifikasikan citra rempah-rempah Indonesia dengan 31 kelas. Kedua, seberapa efektif pendekatan *transfer learning* dengan *two-stage training* dalam meningkatkan akurasi klasifikasi dibandingkan penelitian sebelumnya pada dataset yang sama. Ketiga, bagaimana *trade-off* antara akurasi klasifikasi dan efisiensi komputasi (waktu *inference*) pada masing-masing arsitektur untuk mendukung pemilihan model berdasarkan skenario aplikasi. Keempat, arsitektur manakah yang memberikan keseimbangan optimal antara performa klasifikasi dan efisiensi komputasi untuk implementasi praktis sistem identifikasi rempah.

Penelitian ini bertujuan untuk mengembangkan dan membandingkan performa keempat arsitektur CNN tersebut dalam klasifikasi citra rempah-rempah Indonesia. Evaluasi dilakukan menggunakan metrik akurasi, *precision*, *recall*, *F1-score*, serta analisis waktu *inference* untuk menilai efisiensi komputasi masing-masing arsitektur. Penelitian ini diharapkan dapat memberikan analisis komparatif yang komprehensif terhadap berbagai arsitektur CNN pada dataset rempah dengan jumlah kelas yang lebih besar, serta menawarkan model yang berpotensi diterapkan dalam pengembangan sistem identifikasi rempah yang praktis dan efisien.

## 2 Tinjauan Literatur

Penelitian klasifikasi citra berbasis *deep learning* berkembang pesat sejak keberhasilan AlexNet pada kompetisi ImageNet tahun 2012, yang menandai penggunaan Convolutional Neural Network (CNN) secara luas dalam berbagai aplikasi *computer vision* [9]. Salah satu pendekatan yang banyak digunakan adalah *transfer learning*, yaitu pemanfaatan model pra-latih berskala besar seperti ImageNet untuk meningkatkan performa pada dataset yang lebih kecil dan spesifik. Sejumlah studi menunjukkan bahwa *transfer learning* mampu mengurangi kebutuhan data latih dan tetap mempertahankan akurasi yang tinggi, sehingga metode ini dinilai efektif untuk permasalahan dengan ketersediaan data terbatas [4].

Berbagai arsitektur CNN telah berkembang dengan karakteristik dan keunggulan masing-masing. VGG16 dikenal karena arsitekturnya yang sederhana dan stabil, namun memiliki jumlah parameter yang besar sehingga membutuhkan komputasi tinggi [3]. ResNet memperkenalkan konsep *residual connection* untuk mengatasi permasalahan *vanishing gradient* pada jaringan yang lebih dalam, dan terbukti menghasilkan performa unggul di banyak tugas klasifikasi [10]. EfficientNet menawarkan compound scaling yang menyeimbangkan kedalaman, lebar, dan resolusi secara bersamaan sehingga mencapai akurasi tinggi dengan jumlah parameter yang lebih efisien [11]. Sementara itu, MobileNetV2 dirancang untuk perangkat mobile melalui penggunaan *depthwise separable convolution* dan *inverted residual blocks* yang memungkinkan peningkatan efisiensi komputasi tanpa penurunan akurasi yang signifikan [8].

Pemanfaatan CNN untuk klasifikasi rempah-rempah telah dieksplorasi dalam beberapa penelitian dengan berbagai pendekatan dan hasil yang beragam. Saputro *et al.* [12] menggunakan CNN untuk klasifikasi 5 jenis rempah dengan 625 data, mencapai akurasi 96%. Nisa dan Candra [13] mengklasifikasikan 4 jenis rempah rimpang dengan 1.000 citra menggunakan CNN dengan empat lapisan konvolusi, mencapai akurasi *testing* 90%. Suandana *et al.* [6] memanfaatkan MobileNetV2 untuk klasifikasi 16 jenis rempah lokal dengan 5.111 citra, mencapai akurasi validasi 99,96%. Putra *et al.* [5] membandingkan enam arsitektur CNN pada sepuluh jenis rempah, di mana model Xception mencapai nilai F1-Score sebesar 96,99%. Sementara itu, Maulana *et al.* [7] menggunakan VGG16 untuk klasifikasi 31 jenis rempah Indonesia dengan akurasi 84,02% pada data uji setelah 50 *epoch training*.

Penelitian yang menggunakan metode tradisional menunjukkan keterbatasan dalam menangani klasifikasi multi-kelas kompleks. Simanjuntak *et al.* [14] menggunakan HOG dengan Gaussian Naïve Bayes mencapai akurasi 94,6% untuk lima kelas, namun menurun signifikan menjadi 63,3% pada 15 kelas. Muhathir *et al.* [15] menggunakan HOG dengan XGBoost mencapai akurasi sebesar 80,6% untuk 15 jenis rempah. Hasil-hasil tersebut menunjukkan bahwa performa metode tradisional masih berada di bawah metode deep learning, yang menegaskan keunggulan CNN dalam mengekstraksi fitur visual yang kompleks.

Selain pemilihan arsitektur, berbagai teknik peningkatan performa juga berperan dalam hasil klasifikasi. Augmentasi data seperti rotasi, translasi, zoom, dan *flipping* terbukti mampu meningkatkan generalisasi model pada dataset terbatas [16]. Teknik regularisasi seperti *dropout*, *batch normalization*, dan regulasi L2 efektif dalam mengurangi *overfitting* [17]. Selain itu, *label smoothing* dapat membantu menstabilkan distribusi prediksi dan meningkatkan *robustness* model [18]. Pendekatan *transfer learning* dengan skema *two-stage training* yang membekukan *base model* pada tahap awal kemudian melakukan *fine-tuning* dengan *learning rate* yang lebih rendah, telah menunjukkan peningkatan performa yang konsisten pada berbagai aplikasi klasifikasi citra [19].

Tabel 1 menyajikan perbandingan penelitian terdahulu yang relevan dan memperlihatkan posisi kontribusi penelitian ini terhadap *body of knowledge* yang telah ada.

**Tabel 1 Perbandingan dengan penelitian terdahulu**

| Penelitian                | Dataset           | Jumlah Kelas | Model                   | Akurasi            |
|---------------------------|-------------------|--------------|-------------------------|--------------------|
| Maulana <i>et al.</i>     | Indonesian Spices | 31           | VGG16                   | 84,02%             |
| Suandana <i>et al.</i>    | Rempah Lokal      | 16           | MobileNetV2             | 99,96%<br>(val)    |
| Putra <i>et al.</i>       | Custom Spices     | 10           | 6 arsitektur (Xception) | 97%                |
| Simanjuntak <i>et al.</i> | Custom            | 15           | HOG + Naïve Bayes       | 94,6%<br>(5 kelas) |
| Muhathir <i>et al.</i>    | Custom            | 15           | HOG + XGBoost           | 80,6%              |

Berdasarkan tinjauan literatur tersebut, masih terdapat beberapa kesenjangan penelitian. Pertama, terbatasnya penelitian yang membandingkan berbagai arsitektur CNN populer pada dataset rempah-rempah Indonesia dengan jumlah kelas yang besar. Kedua, evaluasi waktu *inference* sebagai indikator efisiensi komputasi masih jarang dilakukan, meskipun aspek ini krusial untuk implementasi pada

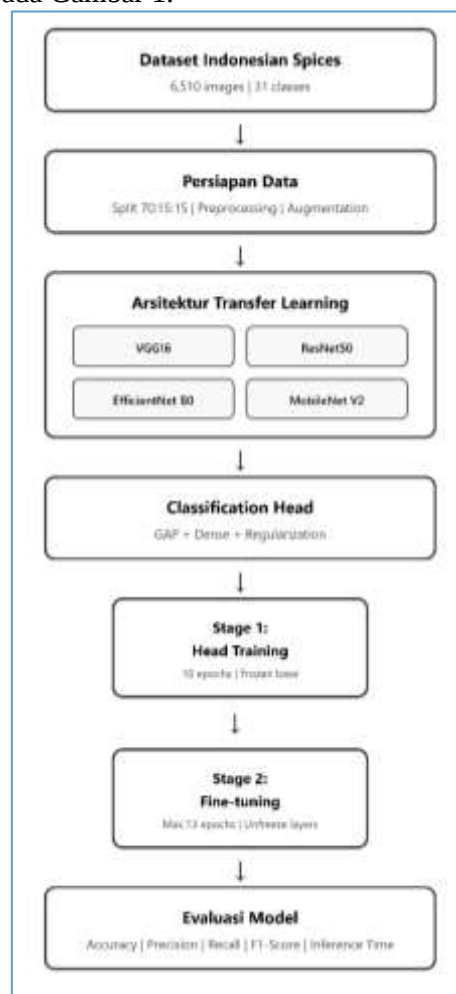
<http://sistemasi.ftik.unisi.ac.id>

perangkat *mobile* atau sistem berbasis *web*. Ketiga, meskipun penelitian Maulana *et al.* [7] telah menggunakan dataset dengan 31 kelas, akurasi yang dicapai masih memiliki ruang perbaikan signifikan. Keempat, belum ada penelitian yang menerapkan strategi *two-stage transfer learning* dengan *fine-tuning* adaptif pada dataset Indonesian Spices dengan 31 kelas. Kelima, analisis *trade-off* antara akurasi dan efisiensi komputasi untuk mendukung pemilihan model berdasarkan skenario aplikasi praktis belum tersedia secara komprehensif.

Penelitian ini bertujuan untuk mengisi kesenjangan tersebut melalui analisis komparatif empat arsitektur CNN populer, yaitu VGG16, ResNet50, EfficientNetB0, dan MobileNetV2, dengan implementasi *two-stage transfer learning*, evaluasi komprehensif terhadap akurasi dan efisiensi *inference*, serta analisis *trade-off* untuk mendukung pengembangan sistem klasifikasi rempah yang lebih aplikatif. Dengan menggunakan dataset Indonesian Spices yang terdiri dari 31 kelas dan 6.510 citra, penelitian ini diharapkan dapat memberikan *baseline* yang kuat dan rekomendasi praktis untuk implementasi sistem identifikasi rempah-rempah Indonesia di berbagai skenario *deployment*.

### 3 Metode Penelitian

Penelitian ini menggunakan pendekatan eksperimental untuk membandingkan performa empat arsitektur CNN dalam klasifikasi citra rempah-rempah Indonesia. Metodologi penelitian disusun secara sistematis, mulai dari persiapan dataset, *preprocessing*, pembangunan model, strategi pelatihan, hingga evaluasi komprehensif berdasarkan akurasi dan efisiensi komputasi. Kerangka umum metodologi penelitian disajikan pada Gambar 1.



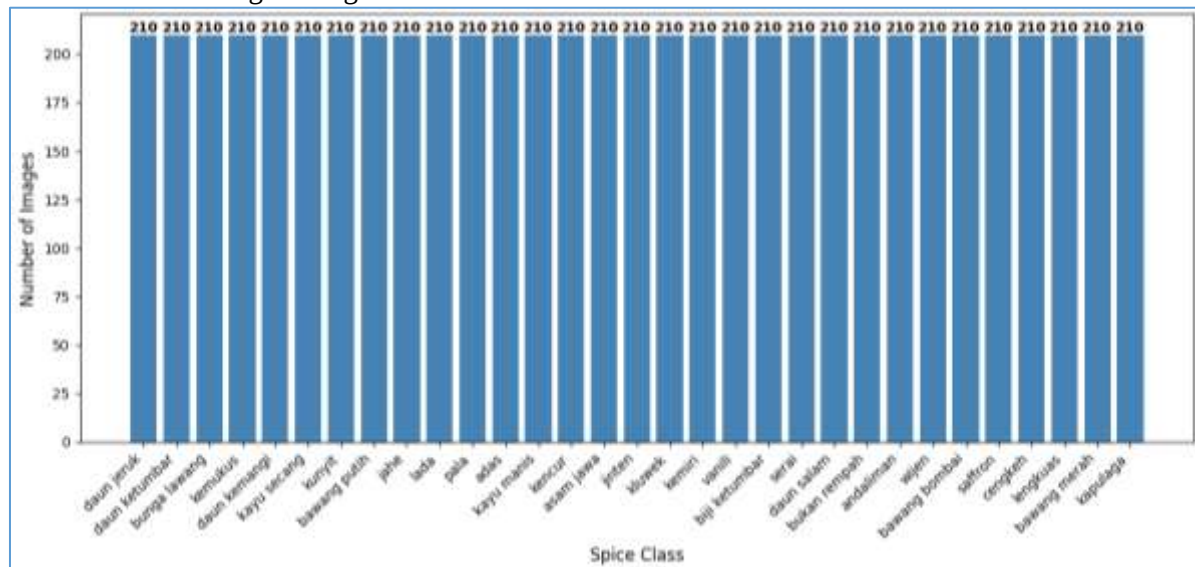
Gambar 1 Kerangka kerja metodologi penelitian

#### 3.1 Dataset dan Analisis Eksploratori

Penelitian ini menggunakan Indonesian Spices Dataset yang diperoleh dari Kaggle melalui kagglehub API [20]. Dataset terdiri dari 6.510 citra berformat JPG yang terbagi secara merata ke

<http://sistemasi.ftik.unisi.ac.id>

dalam 31 kelas rempah Indonesia. Analisis eksploratori dilakukan untuk memahami karakteristik awal dataset [21]. Distribusi jumlah citra per kelas ditampilkan pada Gambar 2 dan menunjukkan bahwa seluruh kelas memiliki jumlah citra yang sama yaitu 210 citra per kelas. Selain itu, visualisasi contoh citra dari setiap kelas disajikan pada Gambar 3 menggunakan visualisasi grid yang menampilkan satu citra acak dari masing-masing kelas.



**Gambar 2 Distribusi kelas dataset**



.....

### 3.2 Preprocessing dan Augmentasi Data

Augmentasi data diterapkan pada subset *training* untuk meningkatkan kemampuan generalisasi model. Transformasi yang digunakan meliputi rotasi hingga  $\pm 20^\circ$ , translasi horizontal dan vertikal sebesar 0,15, zoom sebesar 0,15, horizontal flip, serta penyesuaian kecerahan dalam rentang 0,8 hingga 1,2. Teknik augmentasi ini terbukti efektif meningkatkan generalisasi model dan mengurangi overfitting pada dataset citra terbatas. Sementara itu, data validasi dan uji hanya mengalami normalisasi tanpa augmentasi untuk menjaga konsistensi evaluasi [25].

### 3.3 Arsitektur Model

Penelitian ini membandingkan empat arsitektur Convolutional Neural Network (CNN) pra-latih pada ImageNet, yaitu VGG16, ResNet50, EfficientNetB0, dan MobileNetV2. Setiap arsitektur digunakan tanpa lapisan klasifikasi bawaan dengan ukuran input  $224 \times 224 \times 3$  piksel. Di atas *base model* ditambahkan *classification head* yang terdiri dari lapisan Global Average Pooling 2D untuk mengurangi dimensi spasial [26], diikuti dua lapisan Dense dengan masing-masing 256 dan 128 unit serta fungsi aktivasi ReLU. Untuk meningkatkan stabilitas dan generalisasi model, digunakan kombinasi *batch normalization* [24], *dropout* [17], dan regularisasi L2 sebesar 0,0001 [27]. Lapisan keluaran berupa Dense dengan 31 unit dan aktivasi softmax digunakan untuk melakukan klasifikasi multikelas [28].

Detail konfigurasi *hyperparameter* untuk setiap arsitektur disajikan pada Tabel 2. Kombinasi regularisasi ini dipilih berdasarkan *best practices* dalam *deep learning* untuk mencegah *overfitting* sambil mempertahankan kapasitas model [25].

**Tabel 2 Konfigurasi hyperparameter detail untuk setiap arsitektur**

| Arsitektur     | Total Parameters | Dropout Rate (Layer 1 / Layer 2) | L2 Regularization | Fine-tune Layers | LR Stage 1         | LR Stage 2         |
|----------------|------------------|----------------------------------|-------------------|------------------|--------------------|--------------------|
| VGG16          | 14.884.447       | 0,4 / 0,2                        | 0,0001            | 12 (last layers) | $5 \times 10^{-4}$ | $5 \times 10^{-5}$ |
| ResNet50       | 24.150.687       | 0,4 / 0,2                        | 0,0001            | 48 (last layers) | $5 \times 10^{-4}$ | $3 \times 10^{-5}$ |
| EfficientNetB0 | 4.415.938        | 0,4 / 0,2                        | 0,0001            | 50 (last layers) | $5 \times 10^{-4}$ | $3 \times 10^{-5}$ |
| MobileNetV2    | 2.624.351        | 0,4 / 0,2                        | 0,0001            | 40 (last layers) | $5 \times 10^{-4}$ | $5 \times 10^{-5}$ |

### 3.4 Prosedur Latihan

Proses pelatihan dilakukan menggunakan pendekatan *two-stage transfer learning* untuk memaksimalkan pemanfaatan fitur pra-latih sambil tetap memperoleh generalisasi yang optimal [19]. Pada tahap pertama (*Head Layer Training*), seluruh *base model* dibekukan sehingga hanya *classification head* yang dilatih selama 10 *epoch*. Tahap ini menggunakan optimizer Adam [29] dengan learning rate  $5 \times 10^{-4}$ , serta fungsi kerugian Categorical Crossentropy dengan label smoothing 0,05 untuk meningkatkan regularisasi dan mengurangi overconfidence prediksi [18].

Pada tahap kedua (*Fine-tuning*), sejumlah lapisan akhir dari *base model* dibuka untuk menyesuaikan fitur tingkat tinggi terhadap dataset target [30]. Jumlah lapisan yang dibuka menyesuaikan kedalaman masing-masing arsitektur, seperti tercantum pada Tabel 2. Proses *fine-tuning* dilakukan dengan maksimum 13 *epoch* dengan *learning rate* yang lebih rendah untuk mencegah terjadinya *catastrophic forgetting* [31]. Selain itu, digunakan callback ReduceLROnPlateau untuk menurunkan *learning rate* secara adaptif ketika *validation loss* tidak menunjukkan peningkatan [32], serta EarlyStopping dengan *patience* empat *epoch* untuk menghentikan pelatihan lebih awal dan mengembalikan bobot terbaik [33].

### 3.5 Evaluasi Model

Evaluasi performa dilakukan secara komprehensif melalui berbagai metrik, meliputi *accuracy*, *precision*, *recall*, dan *F1-score* berbobot (*weighted*) untuk mengukur performa klasifikasi secara keseluruhan [34]. Selain itu, metrik per kelas digunakan untuk mendapatkan gambaran performa lebih mendetail, sedangkan *confusion matrix* memberikan visualisasi terhadap pola kesalahan klasifikasi [35].

Efisiensi komputasi dievaluasi melalui pengukuran waktu *inference* per citra dalam satuan milidetik, serta waktu pelatihan total untuk kedua tahap [8]. Pengukuran waktu *inference* dilakukan dengan mengevaluasi seluruh test set (977 citra) menggunakan *batch processing*, kemudian waktu total dibagi dengan jumlah citra untuk mendapatkan rata-rata waktu per citra. Evaluasi menggunakan *batch size* 32 yang konsisten dengan proses *training*. Seluruh eksperimen dijalankan menggunakan TensorFlow versi 2.x dan Keras API [36] pada lingkungan Google Colab dengan akselerasi GPU NVIDIA Tesla T4, menggunakan *batch size* 32 serta *random seed* 42 untuk memastikan reproduktibilitas hasil eksperimen [37].

## 4 Hasil dan Pembahasan

Bagian ini menyajikan hasil eksperimen dari empat arsitektur CNN, meliputi analisis performa, dinamika pelatihan, *trade-off* akurasi-efisiensi, serta perbandingan dengan penelitian sebelumnya. Pembahasan mencakup rekomendasi pemilihan model berdasarkan skenario aplikasi praktis yang berbeda.

### 4.1 Kinerja Model Secara Keseluruhan

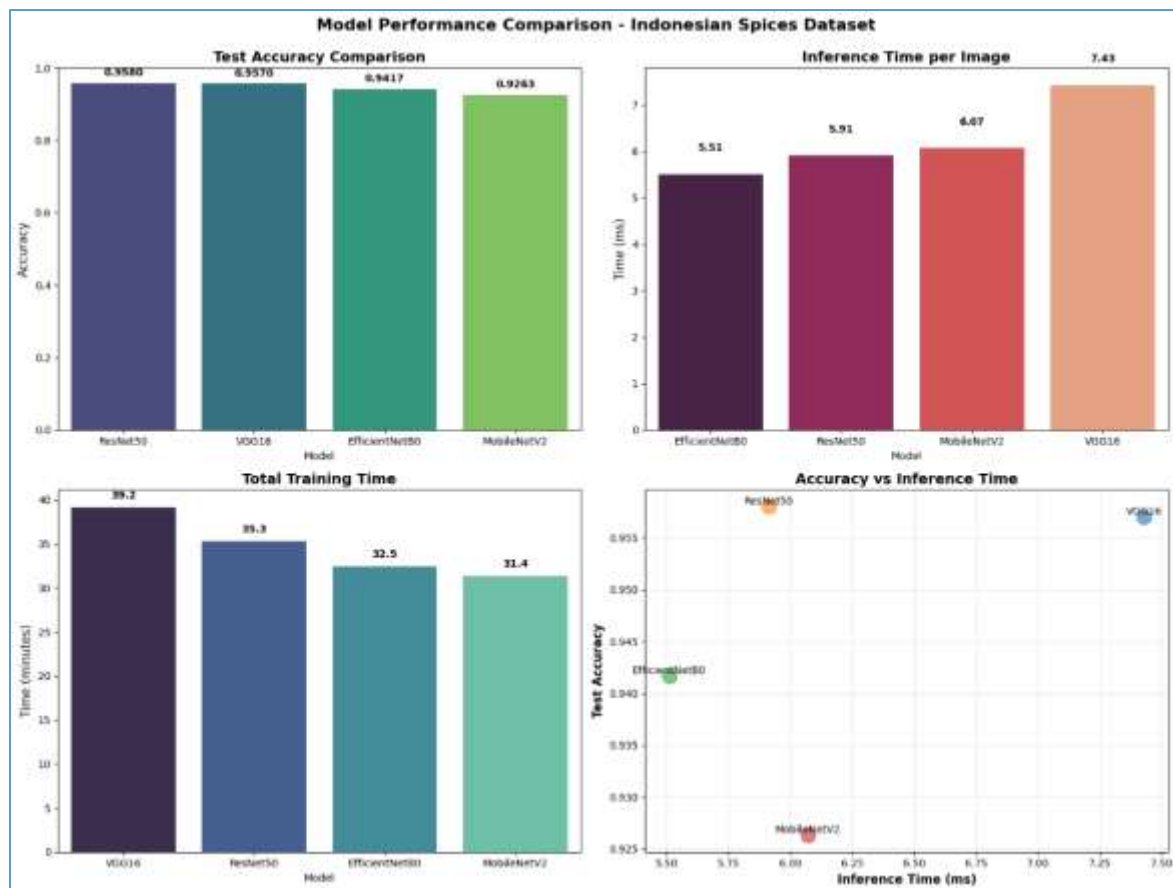
Evaluasi performa keempat arsitektur CNN pada dataset Indonesian Spices menunjukkan hasil yang sangat kompetitif dengan akurasi di atas 92% pada *test set*. ResNet50 mencapai performa terbaik dengan *test accuracy* 95,80%, diikuti VGG16 (95,70%), EfficientNetB0 (94,17%), dan MobileNetV2 (92,63%). Tingginya nilai *precision*, *recall*, dan *F1-score* yang seimbang untuk semua model menunjukkan bahwa klasifikasi tidak hanya akurat secara keseluruhan, tetapi juga konsisten di seluruh kelas tanpa bias signifikan terhadap kelas tertentu. Performa ini mencerminkan efektivitas pendekatan *transfer learning* dalam mengadaptasi fitur pre-trained dari ImageNet ke domain rempah-rempah Indonesia. Tabel 3 merangkum perbandingan performa, termasuk metrik akurasi, efisiensi komputasi, dan waktu pelatihan.

**Tabel 3 Perbandingan performa model pada test set**

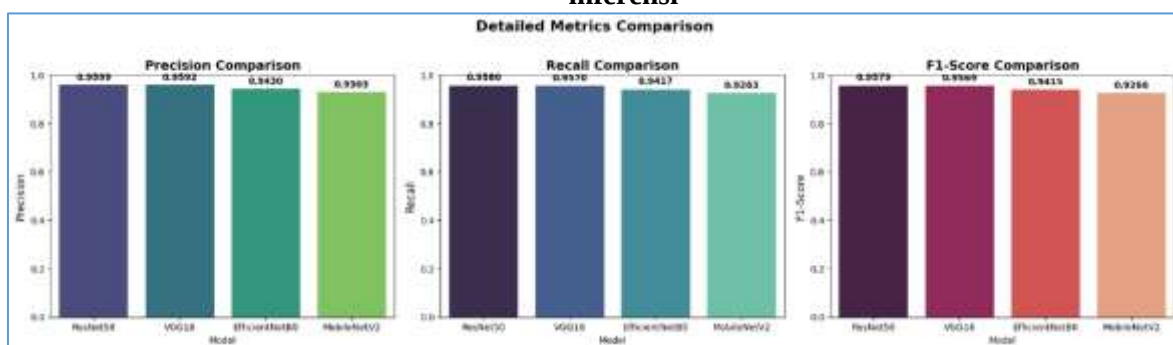
| Model          | Train Acc | Test Acc | Precision | Recall | F1-Score | Test Loss | Inference (ms) | Train Time (min) |
|----------------|-----------|----------|-----------|--------|----------|-----------|----------------|------------------|
| VGG16          | 98,33%    | 95,70%   | 95,92%    | 95,70% | 95,69%   | 24,18%    | 7,43           | 39,2             |
| ResNet50       | 99,03%    | 95,80%   | 95,99%    | 95,80% | 95,79%   | 23,53%    | 5,91           | 35,3             |
| EfficientNetB0 | 93,55%    | 94,17%   | 94,30%    | 94,17% | 94,15%   | 26,57%    | 5,51           | 32,5             |
| MobileNetV2    | 95,39%    | 92,63%   | 93,03%    | 92,63% | 92,66%   | 31,97%    | 6,07           | 31,4             |

Analisis generalisasi model berdasarkan selisih antara *training accuracy* dan *test accuracy* menunjukkan bahwa EfficientNetB0 memiliki generalisasi terbaik dengan selisih terkecil hanya 0,62 poin persentase, bahkan dengan *test accuracy* lebih tinggi dari *training accuracy*. VGG16 menunjukkan selisih 2,63 poin persentase, MobileNetV2 dengan selisih 2,76 poin persentase, dan ResNet50 dengan selisih terbesar 3,23 poin persentase namun tetap menghasilkan akurasi test tertinggi. Selisih yang relatif kecil untuk semua model mengindikasikan bahwa regularisasi yang diterapkan efektif dalam mencegah *overfitting* [17].

Perbandingan visual performa ditampilkan pada Gambar 4, menunjukkan bahwa ResNet50 dan VGG16 unggul dalam akurasi, sementara EfficientNetB0 menawarkan efisiensi inferensi terbaik (5,51 ms per citra). *Trade-off* antara akurasi dan kecepatan inferensi jelas terlihat, dengan ResNet50 memberikan kombinasi optimal. Evaluasi metrik detail pada Gambar 5 menunjukkan konsistensi performa antar berbagai metrik evaluasi, dengan ResNet50 unggul di semua metrik (*precision* 95,99%, *recall* 95,80%, *F1-score* 95,79%).



Gambar 4 Perbandingan komprehensif performa keempat model: (a) *test accuracy*, (b) waktu inferensi per citra, (c) waktu pelatihan total, (d) *trade-off* akurasi vs kecepatan inferensi



Gambar 5 Perbandingan metrik detail untuk keempat model: (a) *precision*, (b) *recall*, (c) *f1-score*

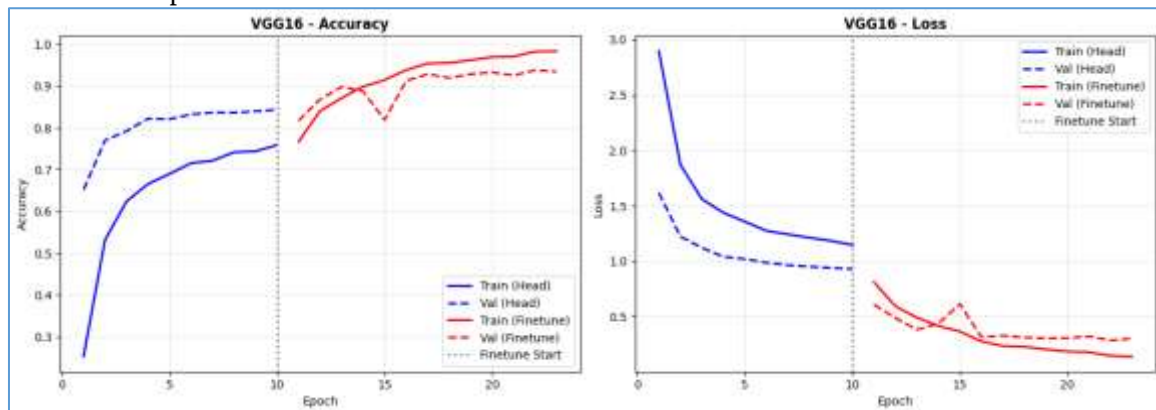
## 4.2 Dinamika Pelatihan

Proses pelatihan dilakukan dalam dua tahap untuk memaksimalkan *transfer learning* sambil mencegah *catastrophic forgetting* [31]. Pada stage pertama dengan *base model* dibekukan, EfficientNetB0 mencapai performa terbaik dengan *validation accuracy* 92,02%, diikuti ResNet50 (90,38%), MobileNetV2 (89,36%), dan VGG16 (84,34%). Perbedaan performa ini mengindikasikan tingkat *transferability* yang berbeda dari fitur pre-trained ImageNet terhadap domain rempah-rempah Indonesia. EfficientNetB0 menunjukkan selisih *training-validation* terkecil (2,58 poin persentase), sementara VGG16 dengan selisih terbesar (8,46 poin persentase) memerlukan *fine-tuning* lebih intensif.

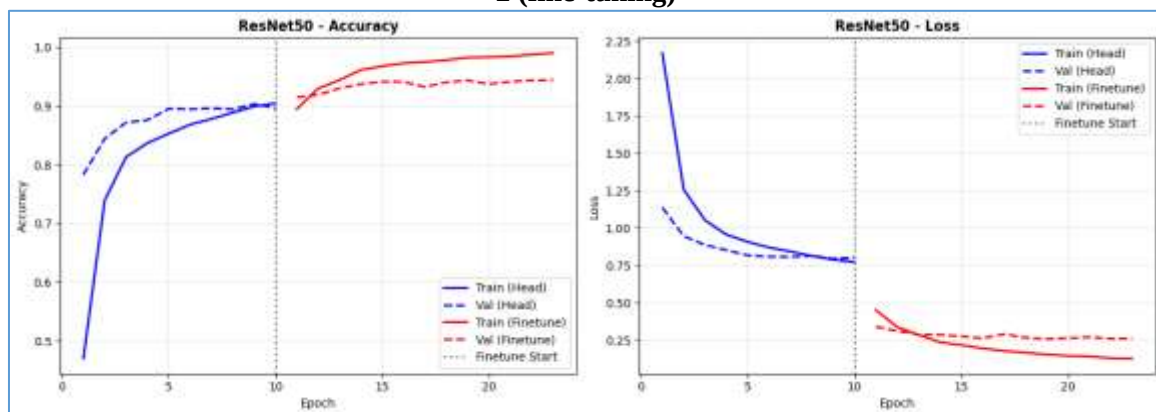
Tahap kedua memberikan peningkatan signifikan melalui *fine-tuning* lapisan akhir dengan *learning rate* lebih rendah. VGG16 meningkat paling signifikan menjadi *validation accuracy* 93,76% (peningkatan 9,42 poin persentase), ResNet50 mencapai 94,47% (peningkatan 4,09 poin persentase), MobileNetV2 mencapai 92,43% (peningkatan 3,07 poin persentase), dan EfficientNetB0 mencapai

<http://sistemasi.ftik.unisi.ac.id>

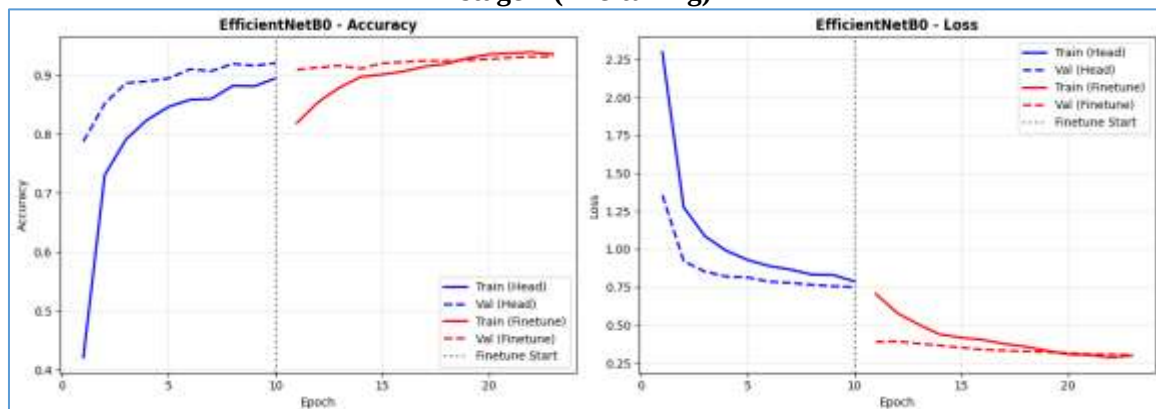
93,14% (peningkatan 1,12 poin persentase). Grafik pelatihan untuk keempat model ditampilkan pada Gambar 6a-6d, menunjukkan pola konvergensi yang stabil dengan peningkatan bertahap dari tahap pertama ke tahap kedua.



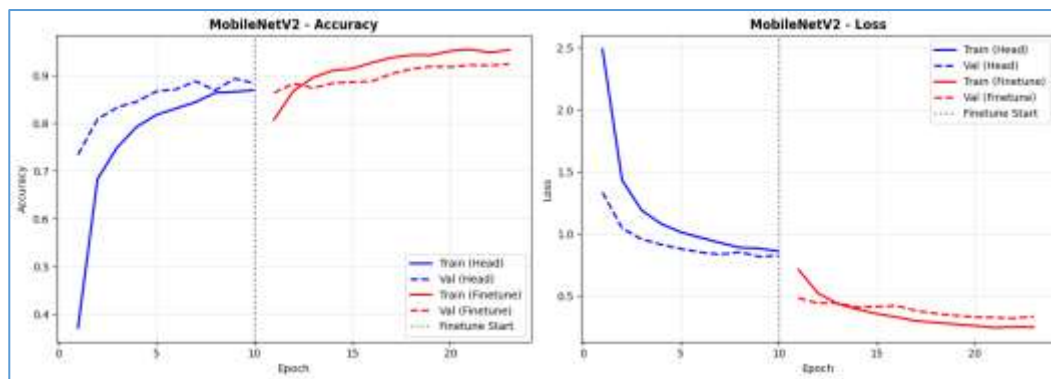
**Gambar 6a Grafik pelatihan VGG16: accuracy dan loss untuk stage 1 (head training) dan stage 2 (fine-tuning)**



**Gambar 6b Grafik pelatihan ResNet50: accuracy dan loss untuk stage 1 (head training) dan stage 2 (fine-tuning)**



**Gambar 6c Grafik Pelatihan EfficientNetB0: Accuracy dan Loss untuk Stage 1 (Head Training) dan Stage 2 (Fine-tuning)**



**Gambar 6d Grafik pelatihan MobileNetV2: accuracy dan loss untuk stage 1 (head training) dan stage 2 (fine-tuning)**

Callback ReduceLROnPlateau teraktivasi pada berbagai *epoch* untuk setiap model, membantu konvergensi yang lebih halus dengan menurunkan *learning rate* adaptif ketika validation loss plateau [32]. ResNet50 mengalami pengurangan learning rate paling agresif dengan 3 kali penurunan di tahap 2 (*epoch* 8, 11, dan 13), dari  $3 \times 10^{-5}$  turun hingga  $3,75 \times 10^{-6}$ . VGG16 mengalami 2 kali penurunan (*epoch* 5 dan 11), turun dari  $5 \times 10^{-5}$  menjadi  $1,25 \times 10^{-5}$ . MobileNetV2 juga mengalami penurunan di *epoch* 6 dari  $5 \times 10^{-5}$  menjadi  $2,5 \times 10^{-5}$ . Sementara EfficientNetB0 menunjukkan konvergensi paling stabil tanpa aktivasi ReduceLROnPlateau sama sekali di tahap 2, mengindikasikan *learning rate*  $3 \times 10^{-5}$  sudah optimal sepanjang *fine-tuning*.

#### 4.3 Analisis Trade-off Akurasi vs Efisiensi

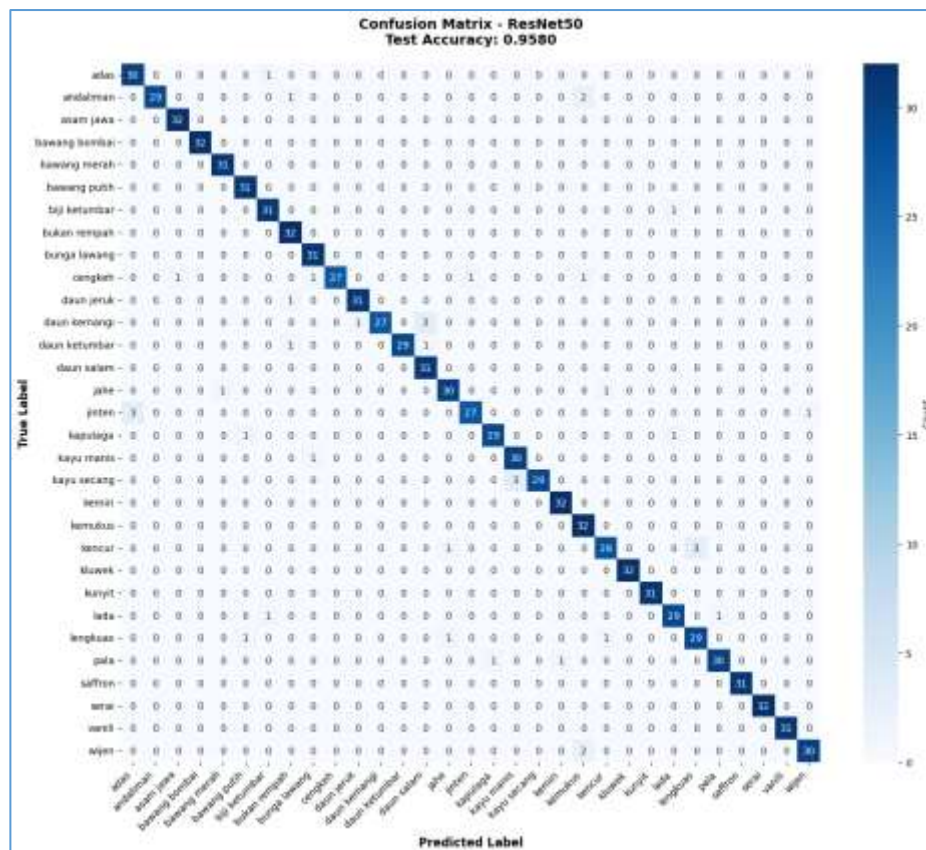
Analisis *trade-off* antara akurasi dan efisiensi komputasi memberikan wawasan penting untuk pemilihan model berdasarkan skenario penerapan. ResNet50 dan VGG16 menunjukkan performa superior dalam akurasi (95,80% dan 95,70%), namun berbeda signifikan dalam efisiensi. ResNet50 mencapai waktu inferensi 5,91 ms per citra dan waktu pelatihan 35,3 menit, sementara VGG16 memerlukan 7,43 ms per citra dan 39,2 menit untuk pelatihan. Perbedaan ini disebabkan struktur residual ResNet50 yang lebih efisien meskipun memiliki lebih banyak parameter (23,59 juta vs 14,88 juta).

EfficientNetB0 menunjukkan keseimbangan optimal antara akurasi dan efisiensi dengan *test accuracy* sebesar 94,17%, waktu inferensi tercepat 5,51 ms per citra, dan waktu pelatihan 32,5 menit. Model ini mencapai akurasi hanya 1,63 poin persentase di bawah ResNet50 dengan efisiensi inferensi 6,8% lebih cepat dan kemampuan generalisasi yang lebih baik, ditunjukkan oleh selisih *training-test* sebesar 0,62 poin persentase. MobileNetV2 menawarkan *trade-off* terbaik untuk penerapan pada perangkat *mobile* dengan *test accuracy* 92,63%, waktu inferensi 6,07 ms per citra, dan ukuran model terkecil 2,62 juta parameter.

Berdasarkan analisis *trade-off* tersebut, rekomendasi penerapan untuk berbagai skenario aplikasi adalah sebagai berikut: ResNet50 sesuai untuk sistem *quality control* industri yang memprioritaskan akurasi tinggi, EfficientNetB0 cocok untuk sistem *edge computing* atau *embedded system* yang memerlukan keseimbangan antara akurasi dan efisiensi, MobileNetV2 direkomendasikan untuk aplikasi *mobile* dengan keterbatasan sumber daya, sedangkan VGG16 lebih sesuai untuk pemrosesan sisi server, di mana sumber daya komputasi tidak menjadi kendala.

#### 4.4 Analisis Kinerja per Model

*Confusion matrix* menunjukkan bahwa sebagian besar prediksi berada pada diagonal, mengindikasikan klasifikasi yang tepat. ResNet50 memiliki *confusion matrix* paling bersih dengan kesalahan minimal seperti ditampilkan pada Gambar 7. Kesalahan yang terjadi umumnya pada kelas dengan kemiripan visual tinggi seperti berbagai jenis daun atau antara kencur dan lengkuas. Beberapa kelas seperti vanili, kapulaga, dan bunga lawang mencapai *precision* dan *recall* mendekati 100%, mengindikasikan *distinctive visual features* yang kuat.



Gambar 7 Confusion matrix ResNet50 pada test set (accuracy: 95.80%)

Classification report menunjukkan performa seimbang di seluruh kelas untuk semua model tanpa bias signifikan terhadap kelas tertentu. ResNet50 dengan *precision* 95,99% dan *recall* 95,80% menunjukkan keseimbangan optimal, mengindikasikan model tidak hanya presisi tetapi juga *comprehensive* dalam mendeteksi *instance* dari setiap kelas. VGG16 cenderung lebih konservatif dengan *false negative rate* sedikit lebih tinggi namun *false positive rate* sangat rendah. EfficientNetB0 unggul dalam generalisasi dengan selisih *training-test* terkecil, menunjukkan pembelajaran yang benar-benar memahami karakteristik visual pembeda setiap kelas. MobileNetV2 menunjukkan performa mengesankan untuk arsitektur *lightweight* dengan selisih *precision-recall* kecil (0.40 poin), membuktikan bahwa performa tinggi dapat dicapai dengan *constraint resource* ketat.

#### 4.5 Efektivitas Transfer Learning

Perbandingan dengan penelitian sebelumnya menunjukkan peningkatan performa yang signifikan, yang membuktikan efektivitas pendekatan transfer learning pada klasifikasi citra rempah-rempah Indonesia. Penelitian oleh Maulana dkk [7] menggunakan VGG16 pada dataset 31 kelas yang sama menghasilkan *test accuracy* 84,02%, sementara penelitian ini berhasil mencapai 95,70%, menunjukkan peningkatan 11,68 poin persentase. Selain itu, model ResNet50 pada penelitian ini mencapai akurasi tertinggi sebesar 95,80% menunjukkan peningkatan yang bahkan lebih mengesankan.

Peningkatan performa ini tidak hanya disebabkan oleh penggunaan model pra-latih, tetapi juga oleh strategi pelatihan yang diterapkan. Pendekatan *two-stage transfer learning* memungkinkan model memanfaatkan fitur umum dari ImageNet pada tahap awal, kemudian menyesuaikan fitur tingkat tinggi secara bertahap melalui *fine-tuning* dengan *learning rate* yang lebih rendah. Selama proses pelatihan, penulis mengamati bahwa strategi ini menghasilkan konvergensi yang lebih stabil dan mengurangi fluktuasi *validation loss*.

Selain itu, kombinasi teknik regularisasi yang digunakan berperan penting dalam meningkatkan kemampuan generalisasi model. Penerapan *dropout* dengan dua tingkat berbeda, regularisasi L2, *batch normalization*, serta *label smoothing* membantu menekan *overfitting*, yang tercermin dari selisih *training-test accuracy* yang relatif kecil (0,62–3,23 poin persentase) pada seluruh model. Augmentasi

data yang seimbang juga berkontribusi dalam memperkaya variasi citra latih tanpa mengganggu stabilitas pembelajaran.

Hasil penelitian ini menunjukkan bahwa *transfer learning* sangat efektif untuk klasifikasi rempah-rempah Indonesia, bahkan dengan ukuran dataset yang relatif terbatas (6.510 citra untuk 31 kelas). Model pra-latih ImageNet mampu mentransfer pengetahuan visual dasar secara efektif ke domain rempah-rempah, sehingga mengurangi kebutuhan data latih dalam jumlah besar. Temuan ini memperkuat bahwa dengan strategi pelatihan dan regularisasi yang tepat, *transfer learning* dapat menghasilkan performa yang kompetitif dan aplikatif untuk sistem identifikasi rempah di dunia nyata.

#### 4.6 Kontribusi terhadap Pengembangan Pengetahuan

Penelitian ini memberikan kontribusi teoretis yang relevan terhadap pengembangan pengetahuan dalam bidang *computer vision* dan *transfer learning*, khususnya pada klasifikasi citra rempah-rempah Indonesia. Secara konseptual, penelitian ini memperluas pemahaman mengenai *transferability* fitur visual dari domain umum (ImageNet) ke domain spesifik produk pertanian dengan karakteristik visual yang kompleks dan tingkat kemiripan antar kelas yang tinggi. Hasil empiris menunjukkan bahwa fitur *low-level* dan *mid-level* yang dipelajari dari objek natural dapat diadaptasi secara efektif untuk klasifikasi multi-kelas rempah-rempah.

Selain itu, penelitian ini memberikan bukti empiris mengenai efektivitas *strategi two-stage transfer learning* dengan *fine-tuning* adaptif dalam meningkatkan kinerja model sekaligus meminimalkan *catastrophic forgetting*, sehingga memperkuat teori *transfer learning* sebagaimana dikemukakan oleh Yosinski *et al.* [19]. Temuan ini juga menunjukkan bahwa penerapan kombinasi teknik regularisasi secara sinergis berkontribusi dalam meningkatkan kemampuan generalisasi model, yang tercermin dari perbedaan performa *training* dan *testing* yang relatif kecil pada dataset dengan variasi visual tinggi.

#### 4.7 Implikasi Praktis Penelitian

Secara praktis, hasil penelitian ini berimplikasi pada pengembangan sistem identifikasi rempah-rempah berbasis citra yang dapat diterapkan dalam berbagai konteks nyata. Model klasifikasi yang dikembangkan berpotensi dimanfaatkan sebagai sistem pendukung keputusan dalam proses *quality control* dan sortasi produk pada industri pangan, serta membantu pedagang rempah dalam mengidentifikasi jenis rempah secara otomatis dan konsisten. Bagi masyarakat umum, model ini dapat diintegrasikan ke dalam aplikasi *mobile* untuk mempermudah pengenalan rempah-rempah Indonesia.

Dari sisi implementasi, hasil evaluasi efisiensi komputasi menunjukkan bahwa setiap arsitektur memiliki karakteristik yang sesuai untuk kebutuhan *deployment* yang berbeda. ResNet50 lebih sesuai untuk sistem *server-side* dengan kebutuhan akurasi tinggi, EfficientNetB0 ideal untuk sistem *edge* dengan keterbatasan sumber daya, sementara MobileNetV2 cocok untuk aplikasi *mobile* karena ukuran model yang ringan. Temuan ini memberikan panduan praktis dalam pemilihan arsitektur CNN sesuai dengan skenario penerapan sistem di dunia nyata.

#### 4.8 Keterbatasan Penelitian

Penelitian ini memiliki beberapa keterbatasan yang perlu diperhatikan. Pertama, dataset yang digunakan berasal dari satu sumber dengan kondisi pengambilan citra yang relatif terkontrol, sehingga variasi kondisi dunia nyata seperti pencahayaan ekstrem, latar belakang kompleks, dan kondisi rempah yang telah diolah belum sepenuhnya terwakili. Hal ini berpotensi memengaruhi kemampuan generalisasi model pada lingkungan nyata.

Kedua, evaluasi efisiensi komputasi dilakukan pada satu konfigurasi perangkat keras tertentu, sehingga hasil waktu inferensi dapat berbeda pada platform lain. Ketiga, aspek robustness terhadap degradasi kualitas citra serta *interpretability* model belum dievaluasi secara mendalam. Terakhir, meskipun distribusi kelas seimbang, ukuran dataset per kelas masih relatif terbatas untuk arsitektur CNN yang kompleks.

#### 4.9 Arah Penelitian Masa Depan

Berdasarkan keterbatasan tersebut, penelitian selanjutnya dapat difokuskan pada pengembangan dataset yang lebih besar dan beragam dengan variasi kondisi pengambilan citra yang lebih realistis. Eksplorasi arsitektur *deep learning* yang lebih modern, termasuk model berbasis *attention*, juga dapat dilakukan untuk membandingkan efektivitasnya dengan CNN konvensional.

Selain itu, penerapan teknik optimisasi model seperti *quantization* dan *pruning* perlu dikaji untuk mendukung *deployment* pada perangkat dengan sumber daya terbatas. Integrasi metode *explainable AI* juga penting untuk meningkatkan transparansi dan kepercayaan pengguna terhadap sistem. Ke depan, pengembangan sistem *end-to-end* yang terintegrasi dengan aplikasi *web* atau *mobile*, serta evaluasi langsung melalui studi pengguna, dapat dilakukan untuk memvalidasi kegunaan sistem dalam konteks penggunaan nyata.

## 5 Kesimpulan

Penelitian ini melakukan analisis komparatif terhadap empat arsitektur *Convolutional Neural Network* (CNN), yaitu *VGG16*, *ResNet50*, *EfficientNetB0*, dan *MobileNetV2*, untuk klasifikasi citra rempah-rempah Indonesia menggunakan pendekatan *transfer learning*. Dataset Indonesian Spices yang terdiri dari 31 kelas digunakan untuk mengevaluasi kemampuan model dalam menangani klasifikasi multi-kelas dengan tingkat kemiripan visual tinggi. Hasil eksperimen menunjukkan bahwa *ResNet50* memberikan performa terbaik dengan akurasi pengujian sebesar 95,80%, diikuti oleh *VGG16* sebesar 95,70%. Sementara itu, *EfficientNetB0* dan *MobileNetV2* menawarkan keseimbangan yang baik antara akurasi dan efisiensi komputasi, sehingga lebih sesuai untuk skenario *deployment* dengan keterbatasan sumber daya. Penerapan strategi *two-stage transfer learning* yang dikombinasikan dengan *fine-tuning* adaptif, augmentasi data, dan berbagai teknik regularisasi terbukti efektif dalam meningkatkan generalisasi model serta menekan *overfitting*, yang ditunjukkan oleh selisih *training-test accuracy* yang relatif kecil. Berdasarkan hasil tersebut, *ResNet50* direkomendasikan untuk aplikasi yang memerlukan akurasi tinggi seperti *quality control* industri, *EfficientNetB0* untuk sistem *edge computing*, dan *MobileNetV2* untuk implementasi pada perangkat *mobile*. Temuan penelitian ini diharapkan dapat menjadi referensi dalam pengembangan sistem klasifikasi citra produk pertanian, khususnya rempah-rempah Indonesia, yang akurat dan efisien.

## Referensi

- [1] H. Anggrasari, P. Perdana, and J. H. Mulyo, "Keunggulan Komparatif dan Kompetitif Rempah-Rempah Indonesia di Pasar Internasional," *JURNAL AGRICA*, Vol. 14, No. 1, pp. 9–19, Apr. 2021, DOI: 10.31289/agrica.v14i1.4396.
- [2] A. Adnan et al., "Diversity of Herbs and Spices Plants and Their Importance in Traditional Medicine in the South Aceh District, Indonesia," *Biodiversitas*, Vol. 23, No. 7, Jul. 2022, DOI: 10.13057/biodiv/d230761.
- [3] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," Apr. 2015.
- [4] S. J. Pan and Q. Yang, "A Survey on Transfer Learning," *IEEE Trans. Knowl. Data Eng.*, Vol. 22, No. 10, pp. 1345–1359, Oct. 2010, DOI: 10.1109/TKDE.2009.191.
- [5] A. E. Putra, M. F. Naufal, and V. R. Prasetyo, "Klasifikasi Jenis Rempah menggunakan *Convolutional Neural Network* dan *Transfer Learning*," *Jurnal Edukasi dan Penelitian Informatika*, Vol. 9, No. 1, pp. 12–17, 2023.
- [6] I. N. Suandana, Asriyanik, and W. Apriandari, "Pemanfaatan CNN (*Convolution Neural Network*) dan *MobileNetV2* dalam Klasifikasi Rempah-Rempah Lokal di Indonesia," Oct. 2024.
- [7] R. Maulana, R. D. Z. Putri, T. A. Amelia, H. Syahputra, and F. Ramadhani, "Identifikasi Jenis Rempah-Rempah Indonesia dengan *Convolutional Neural Network* (CNN) menggunakan Arsitektur *VGG16*," Aug. 2024.
- [8] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted Residuals and Linear Bottlenecks," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, IEEE, Jun. 2018, pp. 4510–4520. DOI: 10.1109/CVPR.2018.00474.
- [9] X. Zhao, L. Wang, Y. Zhang, X. Han, M. Deveci, and M. Parmar, "A Review of *Convolutional Neural Networks* in *Computer Vision*," *Artif. Intell. Rev.*, Vol. 57, No. 4, Apr. 2024, DOI: 10.1007/s10462-024-10721-6.
- [10] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, Jun. 2016, pp. 770–778. DOI: 10.1109/CVPR.2016.90.

- [11] M. Tan and Q. V. Le, “EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks,” Sep. 2020.
- [12] M. N. A. Saputra, F. Liantoni, and D. Maryono, “Application of Convolutional Neural Network (CNN) using TensorFlow as a Learning Medium for Spice Classification,” *Ultimatics : Jurnal Teknik Informatika*, Vol. 16, No. 1, 2024.
- [13] C. Nisa and F. Candra, “Klasifikasi Jenis Rempah-Rempah menggunakan Algoritma Convolutional Neural Network,” *MALCOM: Indonesian Journal of Machine Learning and Computer Science*, Vol. 4, No. 1, pp. 78–84, Dec. 2023, DOI: 10.57152/malcom.v4i1.1018.
- [14] T. I. Simanjuntak, Muhathir, Fadlisyah, and I. Safira, “Performance Analysis of Naive Bayes Variation Method in Spice Image Classification using Histogram of Gradient Oriented (HOG) Feature Extraction,” *Journal of Informatics and Telecommunication Engineering*, Vol. 7, No. 1, pp. 282–291, Jul. 2023, DOI: 10.31289/jite.v7i1.7957.
- [15] Muhathir, R. T. Pangestu, I. Safira, and Melisah, “Performance Comparison of Boosting Algorithms in Spices Classification using Histogram of Oriented Gradient Feature Extraction,” *Journal of Computer Science, Information Technology and Telecommunication Engineering*, Vol. 4, No. 1, pp. 342–349, Mar. 2023, DOI: 10.30596/jcositte.v4i1.13710.
- [16] C. Shorten and T. M. Khoshgoftaar, “A Survey on Image Data Augmentation for Deep Learning,” *J. Big Data*, Vol. 6, No. 1, Dec. 2019, DOI: 10.1186/s40537-019-0197-0.
- [17] N. Srivastava, G. Hinton, A. Krizhevsky, and R. Salakhutdinov, “Dropout: A Simple Way to Prevent Neural Networks from Overfitting,” 2014.
- [18] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, “Rethinking the Inception Architecture for Computer Vision,” in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, Jun. 2016, pp. 2818–2826. DOI: 10.1109/CVPR.2016.308.
- [19] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, “How Transferable are Features in Deep Neural Networks?,” Nov. 2014, [Online]. Available: <http://arxiv.org/abs/1411.1792>
- [20] A. Nathaniel, “Indonesian Spices Dataset,” Kaggle Dataset. Accessed: Dec. 01, 2025. [Online]. Available: <https://www.kaggle.com/datasets/albertnathaniel12/indonesian-spices-dataset>
- [21] J. D. Hunter, “Matplotlib: A 2D Graphics Environment,” *Comput. SCI. Eng.*, Vol. 9, No. 3, pp. 90–95, 2007, DOI: 10.1109/MCSE.2007.55.
- [22] F. Pedregosa et al., “Scikit-Learn: Machine Learning in Python,” 2011. [Online]. Available: <http://scikit-learn.sourceforge.net>.
- [23] C. Szegedy et al., “Going Deeper with Convolutions,” Sep. 2014.
- [24] S. Ioffe and C. Szegedy, “Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift,” Mar. 2015.
- [25] F. Chollet, *Deep Learning with Python*. 2017.
- [26] M. Lin, Q. Chen, and S. Yan, “Network in Network,” Mar. 2014.
- [27] A. Krogh and J. A. Hertz, “A Simple Weight Decay can Improve Generalization.”
- [28] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge: MIT Press, 2016.
- [29] D. P. Kingma and J. Ba, “Adam: A Method for Stochastic Optimization,” Jan. 2017, [Online]. Available: <http://arxiv.org/abs/1412.6980>
- [30] A. S. Razavian, H. Azizpour, J. Sullivan, and S. Carlsson, “CNN Features off-the-Shelf: an Astounding Baseline for Recognition,” May 2014.
- [31] M. McCloskey and N. J. Cohen, “Catastrophic Interference in Connectionist Networks: The Sequential Learning Problem,” 1989, pp. 109–165. DOI: 10.1016/S0079-7421(08)60536-8.
- [32] L. N. Smith, “Cyclical Learning Rates for Training Neural Networks,” Apr. 2017.
- [33] L. Prechelt, “Early Stopping — But When?,” 2012, pp. 53–67. DOI: 10.1007/978-3-642-35289-8\_5.
- [34] D. M. W. Powers, “Evaluation: From Precision, Recall and F-Measure to Roc, Informedness, Markedness and Correlation,” Oct. 2020.
- [35] J. Brownlee, “What is a Confusion Matrix in Machine Learning,” Machine Learning Mastery.
- [36] M. Abadi et al., “TensorFlow: A system for Large-Scale Machine Learning,” May 2016.
- [37] M. Abadi et al., “TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems,” Mar. 2016, [Online]. Available: <http://arxiv.org/abs/1603.04467>