

Comparative Analysis of CNN Architectures (ResNet152V2, InceptionV3, and Xception) in Animal Species Classification

Ferry Khusnil Arief^{1*}, Nurul Kholidiah², Rifki Mistahul Munir³, Rahma Yunita⁴
Teknik Informatika, Fakultas Sains dan Teknologi, Universitas Nahdlatul
Ulama Lampung

Corresponding Author: Ferry Khusnil Arief ceryover@gmail.com

ARTICLE INFO

Keywords : Animal
Classification, Convolutional
Neural Network, Transfer
Learning, InceptionV3,
Comparative Analysis

Received : 04 Juni 2025

Revised : 24 Juni 2025

Accepted: 26 Juli 2025

©2025 Arief, Kholidiah,
Munir, Yunita : This is an
open-access article distributed
under the terms of the [Creative
Commons Atribusi 4.0
Internasional](https://creativecommons.org/licenses/by/4.0/).



ABSTRACT

This research contributes to the field of computer vision by identifying the most effective Convolutional Neural Network (CNN) architecture for animal species classification, a fundamental task to support biodiversity conservation. A comparative analysis was conducted on three pre-trained models: ResNet152V2, InceptionV3, and Xception. The method used was transfer learning on the "Animals-10" dataset with a two-stage training strategy: feature extraction followed by fine-tuning. The results show that the InceptionV3 and Xception architectures achieved the highest performance with a validation accuracy of 97.30%. Consequently, InceptionV3 is recommended as the most optimal model due to its lowest loss value, indicating the best generalization ability for similar tasks.

Analisis Komparatif Arsitektur CNN (ResNet152V2, InceptionV3, dan Xception) dalam Klasifikasi Spesies Hewan

Ferry Khusnil Arief^{1*}, Nurul Kholidiah², Rifki Mistahul Munir³, Rahma Yunita⁴
Teknik Informatika, Fakultas Sains dan Teknologi, Universitas Nahdlatul
Ulama Lampung

Corresponding Author: Ferry Khusnil Arief ceryover@gmail.com

ARTICLE INFO

Kata Kunci: Klasifikasi Hewan, Convolutional Neural Network, Transfer Learning, InceptionV3, Analisis Komparatif

Received : 04 Juni 2025

Revised : 24 Juni 2025

Accepted: 26 Juli 2025

©2025 Arief, Kholidiah, Munir, Yunita : This is an open-access article distributed under the terms of the [Creative Commons Atribusi 4.0 Internasional](https://creativecommons.org/licenses/by/4.0/).



ABSTRACT

Penelitian ini memberikan kontribusi pada bidang visi komputer dengan mengidentifikasi arsitektur Convolutional Neural Network (CNN) paling efektif untuk klasifikasi spesies hewan, sebuah tugas fundamental untuk mendukung konservasi keanekaragaman hayati. Proses analisis komparatif dilakukan pada tiga model pra-terlatih: ResNet152V2, InceptionV3, dan Xception. Metode yang digunakan adalah *transfer learning* pada dataset "Animals-10" dengan strategi pelatihan dua tahap, yaitu *feature extraction* diikuti *fine-tuning*. Hasil menunjukkan bahwa arsitektur InceptionV3 dan Xception mencapai performa tertinggi dengan akurasi validasi 97.30%. Implikasinya, InceptionV3 direkomendasikan sebagai model paling optimal karena memiliki nilai *loss* terendah, yang menunjukkan kemampuan generalisasi terbaik untuk tugas serupa.

PENDAHULUAN

Perkembangan kecerdasan buatan (AI) dan ketersediaan data digital dalam volume besar telah mendorong kemajuan pesat dalam bidang visi komputer, khususnya pada tugas klasifikasi gambar. Kemampuan untuk mengidentifikasi dan mengkategorikan objek dalam gambar secara otomatis telah menjadi fundamental dalam berbagai aplikasi, mulai dari sistem keamanan, diagnosis medis, hingga penelitian lingkungan. Di antara berbagai teknik yang ada, *Deep Learning*—sebuah jenis *machine learning* yang menggunakan jaringan saraf tiruan untuk meniru otak manusia (Abdel-Jaber et al., 2022)—telah muncul sebagai pendekatan paling dominan. Secara khusus, *Convolutional Neural Network* (CNN) menjadi teknik unggulan karena kemampuannya mempelajari fitur secara otonom dari gambar-gambar spesifik domain (Dahr & Gaafar, 2024). Didukung oleh pesatnya perkembangan perangkat keras komputer dan terobosan dalam algoritma terkait, metode berbasis *deep learning* dan CNN telah menunjukkan hasil yang unggul di berbagai bidang pengenalan citra (Xu et al., 2022), dan dengan demikian merevolusi cara mesin "melihat" serta menafsirkan data visual.

Salah satu area aplikasi yang krusial dan menantang adalah klasifikasi spesies hewan secara otomatis. Identifikasi spesies yang akurat dan efisien sangat penting untuk pemantauan keanekaragaman hayati, studi ekologi, upaya konservasi, dan pemahaman dampak perubahan lingkungan terhadap fauna. Namun, tugas ini memiliki kompleksitas yang tinggi akibat variasi dalam gambar, seperti pose hewan yang beragam, kondisi pencahayaan yang tidak menentu, latar belakang yang kompleks, dan oklusi sebagian.

Untuk mengatasi tantangan ini, algoritma *Convolutional Neural Network* (CNN) telah terbukti menjadi solusi yang sangat andal. CNN dirancang untuk mengestimasi dan mengekstrak fitur di dalam gambar untuk meningkatkan presisi klasifikasi (Dahr & Gaafar, 2024). Dengan menggunakan algoritma CNN, objek dalam sebuah gambar dapat diklasifikasikan dengan tingkat presisi yang tinggi, menjadikannya solusi efektif untuk mengatasi berbagai masalah klasifikasi berbasis citra (Dwiatmoko et al., 2024). Daripada membangun arsitektur CNN dari awal (*from scratch*) yang membutuhkan sumber daya komputasi masif dan dataset yang sangat besar, pendekatan *transfer learning* menggunakan model pra-terlatih (*pre-trained models*) telah menjadi taktik yang umum dan standar industri. Filosofi di balik *transfer learning* adalah mentransfer pengetahuan dari tugas-tugas yang telah diselesaikan sebelumnya untuk meningkatkan pembelajaran pada tugas baru, sehingga model tidak selalu harus dilatih dari nol (Hosna et al., 2022). Konsep *transfer learning* memungkinkan model yang telah dilatih pada dataset umum berskala besar untuk dispesialisasikan pada tugas-tugas tertentu dengan menggunakan dataset yang jauh lebih kecil dan spesifik pada suatu masalah (Janiesch et al., 2021). Model-model ini, yang pada awalnya dilatih pada dataset masif untuk mempelajari statistik alami gambar, kemudian diadaptasi melalui proses *fine-tuning* pada dataset yang jauh lebih kecil untuk tugas-tugas spesifik. Pendekatan ini telah menunjukkan kinerja yang sangat baik dalam berbagai penerapan (Abdel-Jaber et al., 2022).

Meskipun banyak arsitektur pra-terlatih yang tersedia, seperti ResNet, Inception, dan Xception, pemilihan arsitektur yang paling optimal sering kali bergantung pada karakteristik dataset dan tugas yang dihadapi. Setiap arsitektur memiliki keunggulan dan kelemahan dalam hal kedalaman, efisiensi komputasi, dan kemampuan generalisasi. Oleh karena itu, analisis perbandingan langsung menjadi penting untuk mengidentifikasi model yang paling sesuai untuk domain masalah tertentu.

Penelitian ini bertujuan untuk melakukan analisis komparatif terhadap tiga arsitektur CNN pra-terlatih yang populer: ResNet152V2, InceptionV3, dan Xception. Kinerja ketiganya akan dievaluasi secara sistematis dalam tugas klasifikasi 10 spesies hewan yang berbeda menggunakan dataset "Animals-10". Perbandingan akan difokuskan pada metrik kinerja utama, yaitu akurasi dan *loss* pada data validasi, serta analisis terhadap kecepatan dan stabilitas konvergensi selama proses pelatihan. Hasil dari penelitian ini diharapkan dapat memberikan wawasan berharga mengenai keunggulan dan keterbatasan masing-masing arsitektur serta memberikan rekomendasi model terbaik untuk tugas klasifikasi spesies hewan.

TINJAUAN PUSTAKA

Klasifikasi Citra

Klasifikasi citra adalah proses inti dalam visi komputer untuk mengelompokkan piksel ke dalam kategori-kategori berdasarkan informasi yang dikandungnya. Sebuah piksel dikategorikan jika telah memenuhi syarat tertentu, di mana kategori tersebut bisa sudah ditentukan sebelumnya (*supervised*) atau ditemukan secara otomatis oleh sistem (*unsupervised*). Alur kerjanya terdiri dari dua langkah: mengekstraksi fitur-fitur penting dari citra, lalu mengklasifikasikannya. Oleh karena itu, kunci keberhasilan dari klasifikasi citra terletak pada seberapa baik fitur yang representatif dapat diekstraksi dan dianalisis (Mohamed et al., 2022).

Deep Learning

Deep Learning (DL) dan *machine learning* (ML) telah muncul sebagai teknologi revolusioner dalam bidang kecerdasan buatan. Keduanya menjadi semakin populer dalam beberapa tahun terakhir karena kemampuannya untuk menganalisis kumpulan data besar, membuat prediksi, dan memberikan wawasan yang sebelumnya tidak mungkin diperoleh. Kemajuan ini didorong oleh ketersediaan *big data* dan peningkatan kekuatan komputasi (Sharifani & Amini, 2023).

Meskipun istilah *machine learning* dan *deep learning* sering digunakan secara bergantian, keduanya merupakan cabang kecerdasan buatan yang berbeda. *Deep Learning* (DL) pada dasarnya adalah bagian atau himpunan bagian dari *machine learning* (ML). Secara umum, *machine learning* melibatkan penggunaan metode statistik untuk belajar dari data, memungkinkan komputer untuk belajar dan berkembang secara otomatis dari pengalaman tanpa diprogram secara eksplisit. Di sisi lain, *deep learning* secara khusus menggunakan jaringan saraf tiruan (*neural networks*) untuk belajar dari kumpulan data yang besar. Arsitektur *deep learning* terdiri dari banyak lapisan (*layers*) berisi *node* atau neuron, di mana setiap lapisan

dirancang untuk memodelkan pola yang semakin kompleks dalam data. Kemampuan DL untuk menangani data dalam jumlah besar dan komputasi yang rumit telah menunjukkan kinerja yang sangat baik di berbagai aplikasi (Mienye & Swart, 2024).

Convolutional Neural Network (CNN)

Convolutional Neural Network (CNN) adalah salah satu arsitektur *Deep Learning* (DL) yang paling populer dan umum digunakan, terutama dalam aplikasi yang melibatkan data visual. Arsitektur CNN terinspirasi dari cara kerja korteks visual pada otak manusia dan hewan, dengan keunggulan utamanya adalah kemampuan untuk secara otomatis mendeteksi dan mempelajari fitur-fitur yang relevan dari sebuah gambar tanpa memerlukan rekayasa fitur secara manual. Struktur CNN umumnya terdiri dari tiga jenis lapisan utama: lapisan Konvolusi (*Convolutional Layer*) yang berfungsi sebagai ekstraktor fitur menggunakan serangkaian filter atau kernel untuk menghasilkan peta fitur (*feature maps*), lapisan *Pooling* yang bertugas untuk mereduksi dimensi spasial fitur dan parameter, serta lapisan *Fully Connected* (FC) di bagian akhir yang berfungsi sebagai pengklasifikasi untuk menghasilkan *output* akhir. Efisiensi CNN terletak pada dua mekanisme kunci: pembagian bobot (*parameter sharing*), di mana satu filter yang sama digunakan di seluruh bagian gambar, dan interaksi renggang (*sparse interactions*), yang secara signifikan mengurangi jumlah total parameter yang perlu dilatih dan membantu mencegah *overfitting* (Alzubaidi et al., 2021).

Transfer Learning

Transfer Learning (TL) adalah sebuah metodologi dalam *machine learning* yang bertujuan untuk meningkatkan kinerja pada sebuah tugas target dengan mentransfer pengetahuan yang telah dipelajari dari tugas sumber (*source*) yang terkait. Pendekatan ini sangat efektif untuk mengatasi masalah di mana data pelatihan untuk tugas target mahal atau sulit untuk dikumpulkan, dengan mengabaikan asumsi ML tradisional bahwa data pelatihan dan pengujian harus berasal dari distribusi yang sama. Secara formal, TL bertujuan untuk meningkatkan fungsi prediktif target (f_{TPC}) di dalam *target domain* (D_T) dengan memanfaatkan pengetahuan dari *source domain* (D_S), terutama ketika $D_S \neq D_T$ atau tugasnya berbeda ($T_{SL} \neq T_{TL}$). Keberhasilan TL bergantung pada penentuan "apa, bagaimana, dan kapan" pengetahuan harus ditransfer, karena transfer dari sumber yang tidak relevan dapat menyebabkan penurunan kinerja, sebuah fenomena yang dikenal sebagai *negative transfer* (Agarwal et al., 2021).

Arsitektur CNN Pra-Terlatih

Dalam *deep learning*, arsitektur pra-terlatih didefinisikan sebagai arsitektur yang telah dilatih untuk mengekstraksi fitur-fitur paling signifikan dari gambar, lalu menggunakan fitur tersebut untuk mempelajari tugas baru. Umumnya, arsitektur ini dilatih menggunakan kumpulan data (*database*) ImageNet yang digunakan dalam ImageNet Large-Scale Visual Recognition Challenge (ILSVRC). Arsitektur tersebut dilatih dengan jutaan gambar yang terbagi ke dalam seribu kelas.

Tren penggunaan model pra-terlatih membantu membuat proses pengembangan menjadi lebih cepat dan mudah dibandingkan membangun model dari nol (*from scratch*). Arsitektur pra-terlatih umumnya digunakan untuk klasifikasi, ekstraksi fitur, dan *transfer learning*.

Dalam kasus klasifikasi, arsitektur pra-terlatih bisa langsung digunakan untuk mengklasifikasikan suatu gambar. Setiap arsitektur memiliki karakteristik yang berbeda, namun yang terpenting adalah akurasi, ukuran, dan kecepatan arsitektur. Pemilihan arsitektur didasarkan pada karakteristik-karakteristik ini. Sebuah arsitektur dianggap baik jika memiliki akurasi dan kecepatan yang tinggi, di mana akurasi model diukur berdasarkan ketepatan hasil klasifikasinya (Mathew et al., 2023).

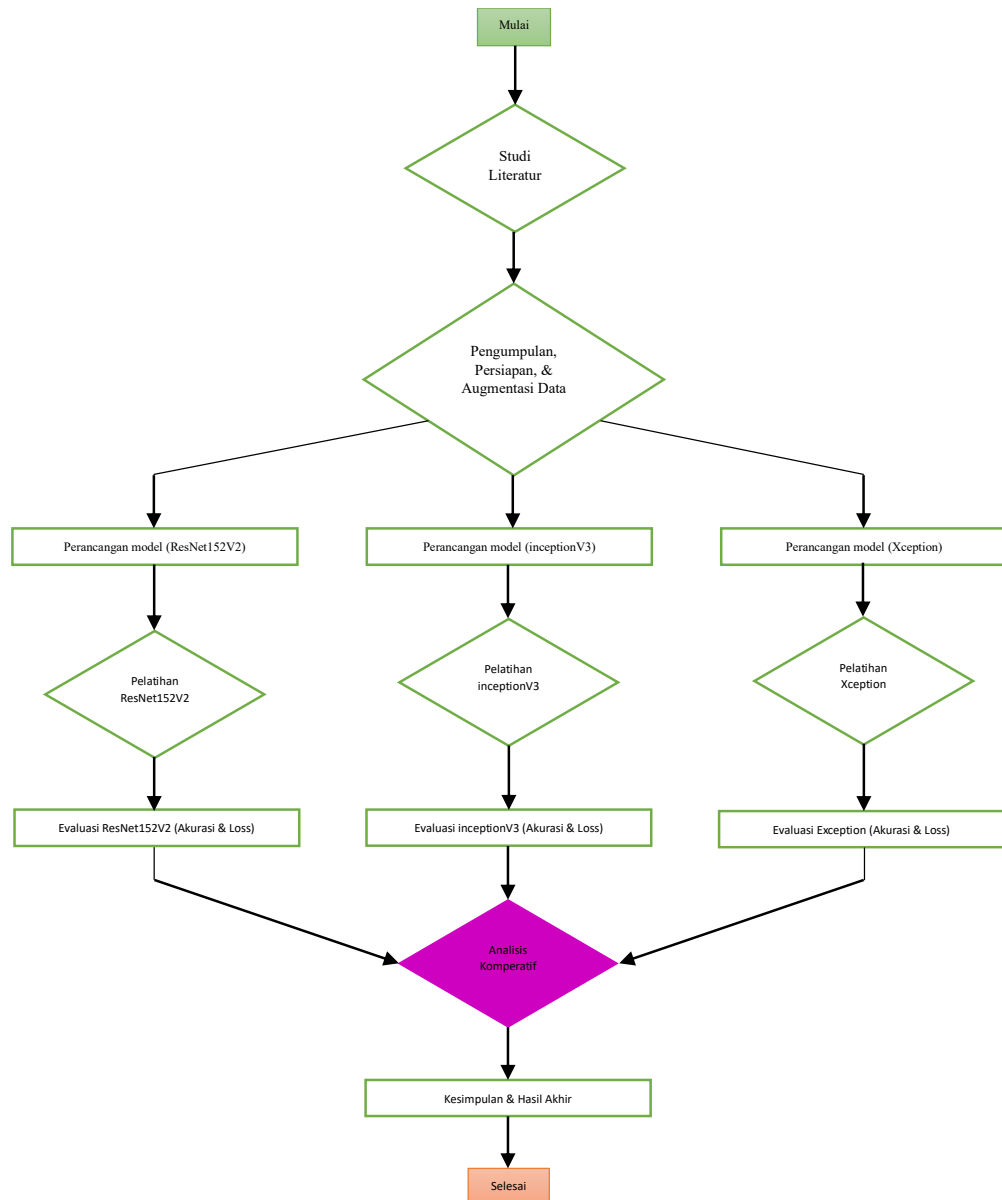
Penelitian ini menggunakan tiga arsitektur CNN pra-terlatih:

- ResNet152V2 : ResNet adalah model yang sudah dilatih sebelumnya (pre-trained model), sehingga tidak lagi membutuhkan pengaturan khusus untuk lapisan-lapisan di dalamnya. Cara kerja ResNet adalah dengan membuat jaringan yang lebih dalam dari jaringan pada umumnya, sekaligus mencari jumlah lapisan yang paling optimal untuk mengatasi masalah hilangnya gradien (*vanishing gradient*) (Ridhovan et al., 2022).
- InceptionV3 : Arsitektur Inception, yang dikenal sebagai GoogleNet (diperkenalkan 2014), adalah model *deep learning*. InceptionV3, generasi ketiga dari Google, membawa inovasi utama berupa metode faktorisasi. Metode ini bertujuan mengurangi parameter dan koneksi, sehingga meningkatkan efisiensi komputasi dan kinerja. Struktur InceptionV3 terdiri dari lapisan konvolusi, *average pooling*, *max pooling*, *dropout*, penggabungan fitur, dan *fully connected layers* (Huda et al., 2023).
- Xception : Arsitektur Xception mengusulkan modifikasi pada modul Inception dengan memperkenalkan *depth-wise separable convolutions* sebagai komponen utamanya. Berbeda dengan konvolusi standar, *depth-wise separable convolutions* secara signifikan mengurangi kompleksitas komputasi dan jumlah parameter dalam model. Proses ini terdiri dari dua tahap utama: pertama, *depthwise convolution* yang menerapkan satu filter secara terpisah untuk setiap kanal input, diikuti oleh *pointwise convolution* (sebuah konvolusi 1×1) untuk menggabungkan hasil dari tahap sebelumnya. Dengan mengurangi jumlah parameter, arsitektur ini tidak hanya menjadi lebih efisien dan ideal untuk aplikasi pada perangkat dengan daya pemrosesan terbatas seperti perangkat seluler, tetapi juga secara efektif membantu mengurangi masalah *overfitting* (Shafik et al., 2025).

METODOLOGI

Kerangka Penelitian

Penelitian ini dilakukan melalui beberapa tahapan utama yang terstruktur untuk memastikan perbandingan yang adil dan hasil yang valid. Alur kerja penelitian diilustrasikan dalam diagram berikut:



Gambar 1. Alur Penelitian

Alat dan Bahan Penelitian

- **Perangkat Keras dan Lunak**

Penelitian ini diimplementasikan dalam lingkungan Google Colaboratory yang menyediakan akses ke *Graphics Processing Unit* (GPU) untuk mempercepat proses pelatihan model. Perangkat lunak utama yang digunakan adalah:

- Bahasa Pemrograman: Python
- Library Deep Learning: TensorFlow v2.x dengan API Keras
- Library Pendukung: NumPy untuk operasi numerik, Matplotlib untuk visualisasi data, dan Pandas untuk manipulasi data.

- **Dataset**

Dataset yang digunakan dalam penelitian ini adalah dataset "Animals-10" yang diperoleh dari platform Kaggle. Dataset ini berisi citra dari 10 spesies hewan yang berbeda. Kelas-kelas tersebut adalah: anjing (*dog*), kuda (*horse*), gajah (*elephant*), kupu-kupu (*butterfly*), ayam (*chicken*), kucing (*cat*), sapi (*cow*), domba (*sheep*), laba-laba (*spider*), dan tupai (*squirrel*).

Tahapan Penelitian

- **Pra-pemrosesan dan Augmentasi Data**

Tahap ini bertujuan untuk menyiapkan data agar sesuai untuk pelatihan model deep learning.

Untuk memastikan perbandingan arsitektur yang adil dan menghindari potensi bias model, dilakukan proses penyeimbangan kelas menggunakan teknik random undersampling. Langkah pertama adalah mengidentifikasi kardinalitas (jumlah citra) dari masing-masing 10 kelas. Ditemukan bahwa jumlah citra paling sedikit pada salah satu kelas adalah 1.446. Angka ini kemudian ditetapkan sebagai titik acuan. Untuk sembilan kelas lainnya yang memiliki jumlah citra lebih banyak, dilakukan pengambilan sampel acak sebanyak 1.446 citra. Proses ini menghasilkan dataset yang seimbang sempurna, di mana setiap kelas memiliki representasi yang setara, dengan total 14.460 citra yang siap digunakan untuk pelatihan dan evaluasi.

Pembagian Dataset: Dataset yang telah diseimbangkan kemudian dibagi menjadi dua set secara otomatis menggunakan `validation_split`:

- a. Data Latih (Training Data): 80% dari total data.
- b. Data Validasi (Validation Data): 20% dari total data.

Augmentasi Data: Untuk meningkatkan variasi data latih dan mencegah overfitting, diterapkan teknik augmentasi citra secara real-time menggunakan `ImageDataGenerator`. Parameter augmentasi yang digunakan meliputi:

- a. Rotasi (`rotation_range=30`)
- b. Pergeseran horizontal (`width_shift_range=0.2`)
- c. Pergeseran vertikal (`height_shift_range=0.2`)
- d. Pergeseran geser (`shear_range=0.2`)
- e. Perbesaran (`zoom_range=0.2`)
- f. Balik horizontal (`horizontal_flip=True`)

Normalisasi dan Perubahan Ukuran Citra: Setiap citra diubah ukurannya sesuai dengan input yang dibutuhkan oleh masing-masing arsitektur:

- a. ResNet152V2: (224, 224) piksel.
- b. InceptionV3 & Xception: (299, 299) piksel. Nilai piksel juga dinormalisasi menggunakan fungsi `preprocess_input` spesifik dari setiap model, yang umumnya menskalakan nilai piksel ke rentang $[-1, 1]$.

- **Perancangan Arsitektur Model**

Penelitian ini memanfaatkan teknik transfer learning dengan menggunakan tiga model CNN yang telah dilatih pada dataset ImageNet.

- a. Model Dasar (Base Model): Digunakan tiga arsitektur berbeda: ResNet152V2, InceptionV3, dan Xception. Bobot pra-terlatih dari ImageNet dimuat, dan lapisan klasifikasi asli (*top layer*) dihilangkan (`include_top=False`).

- b. Lapisan Tambahan (Custom Head): Di atas *base model*, ditambahkan lapisan klasifikasi baru yang terdiri dari:
- GlobalAveragePooling2D: Untuk mereduksi dimensi fitur dari *base model*.
 - Dropout(0.5): Lapisan *dropout* dengan *rate* 50% untuk regularisasi dan mengurangi *overfitting*.
 - Dense: Lapisan *output* dengan 10 neuron (sesuai jumlah kelas) dan fungsi aktivasi Softmax untuk menghasilkan probabilitas klasifikasi. Regularisasi L2(0.001) juga ditambahkan pada lapisan ini.

- **Skenario Pelatihan Model**

Pelatihan untuk setiap model dilakukan dalam dua tahap untuk memaksimalkan performa.

1. Tahap 1: Feature Extraction

- Seluruh lapisan pada *base model* dibekukan (*trainable=False*), sehingga bobotnya tidak diperbarui.
- Hanya lapisan *custom head* yang dilatih.
- Model dikompilasi menggunakan *optimizer* Adam dengan *learning rate* awal sebesar $1e-3$ dan fungsi *loss* *categorical_crossentropy*.

2. Tahap 2: Fine-Tuning

- Setelah tahap pertama, beberapa lapisan teratas dari *base model* dibuka (*trainable=True*) untuk dilatih kembali. Titik pembekuan lapisan bervariasi untuk setiap model (*conv5_block1_out* untuk ResNet152V2, *mixed7* untuk InceptionV3, dan *block14_sepconv1* untuk Xception) untuk menyesuaikan parameter dengan dataset target.
- Model dikompilasi ulang dengan *optimizer* Adam dan *learning rate* yang jauh lebih kecil, yaitu $1e-5$, untuk penyesuaian bobot yang halus dan menghindari perubahan drastis yang dapat merusak fitur yang telah dipelajari.

Untuk mengoptimalkan proses pelatihan, tiga *callbacks* diimplementasikan:

- ModelCheckpoint: Menyimpan bobot model terbaik secara otomatis berdasarkan akurasi validasi (*val_accuracy*) tertinggi.
- EarlyStopping: Menghentikan pelatihan jika *loss* validasi (*val_loss*) tidak menunjukkan perbaikan selama 8 epoch berturut-turut.
- ReduceLROnPlateau: Mengurangi *learning rate* secara otomatis jika *loss validasi* tidak membaik, untuk membantu model keluar dari *local minima*.

- **Evaluasi Model**

Kinerja dari ketiga arsitektur model dievaluasi berdasarkan metrik kuantitatif pada data validasi. Model yang dievaluasi adalah model yang disimpan oleh ModelCheckpoint, yang merepresentasikan performa terbaik selama proses pelatihan. Metrik utama yang digunakan untuk perbandingan adalah:

- Akurasi (Accuracy): merupakan persentase prediksi yang tepat dibandingkan dengan seluruh data yang digunakan dalam proses

validasi. Nilai ini menunjukkan seberapa sering model menghasilkan prediksi yang benar terhadap data yang diuji.

- Loss (Categorical Cross-entropy): Mengukur seberapa baik model dalam memprediksi distribusi probabilitas kelas. Nilai *loss* yang lebih rendah menunjukkan performa model yang lebih baik.

Selain itu, grafik histori akurasi dan *loss* untuk data latih dan validasi per epoch dianalisis untuk mengamati perilaku konvergensi model dan mendeteksi adanya *overfitting*.

HASIL PENELITIAN

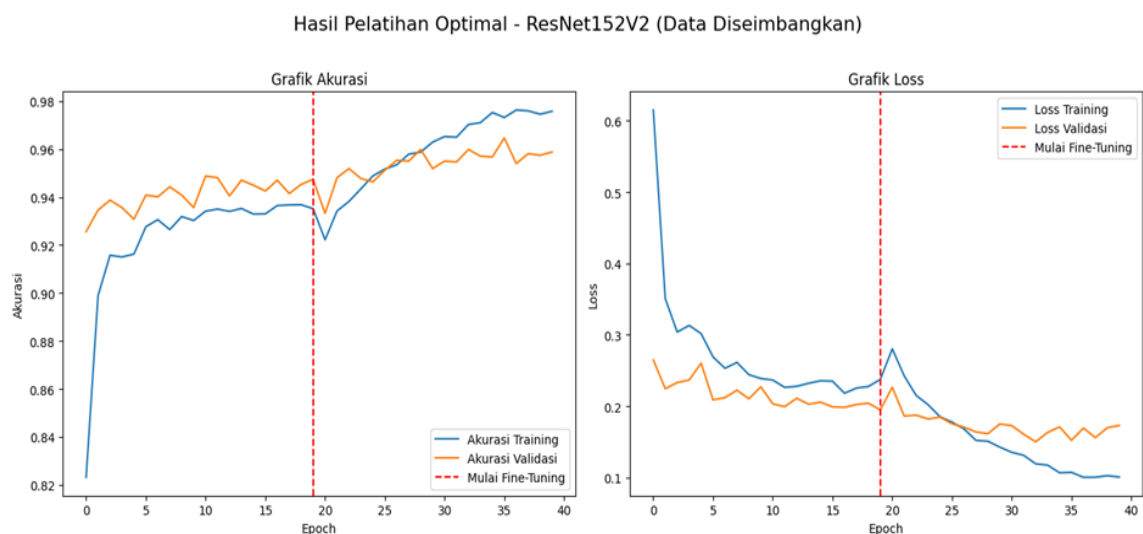
Hasil Pelatihan Model

Pelatihan untuk setiap model dilakukan dalam dua tahap: *feature extraction* dan *fine-tuning*. Kinerja model dievaluasi menggunakan data validasi (20% dari total dataset) dan model dengan bobot terbaik (berdasarkan *val_accuracy* tertinggi) disimpan. Berikut adalah rincian hasil untuk setiap model.

Model ResNet152V2

Model ResNet152V2 dilatih dengan ukuran citra input (224, 224). Setelah melalui tahap *feature extraction* dan *fine-tuning*, model terbaik yang disimpan oleh *callback* ModelCheckpoint dievaluasi pada set data validasi.

- Akurasi Validasi Terbaik: 96.06%
- Loss Validasi Terbaik: 0.1699



Gambar 2. Grafik Akurasi dan Loss ResNet152V2

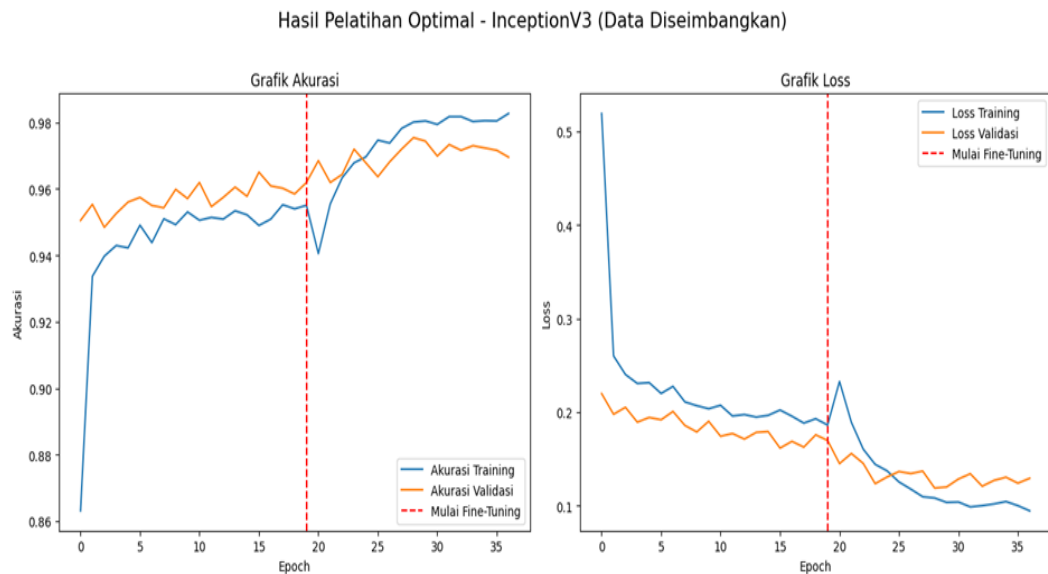
Analisis grafik histori pelatihan (Gambar 2) menunjukkan bahwa akurasi pelatihan dan validasi meningkat secara konsisten, sementara nilai loss menurun. Tahap *fine-tuning* yang dimulai setelah 20 epoch membantu model meningkatkan kinerjanya lebih lanjut, meskipun peningkatannya tidak

signifikan. Penggunaan *callback* berhasil mencegah *overfitting* yang parah dan menjaga model tetap konvergen.

Model InceptionV3

Model InceptionV3 menggunakan ukuran citra input (299, 299). Proses pelatihan dua tahap juga diterapkan pada arsitektur ini.

- Akurasi Validasi Terbaik: 97.30%
- Loss Validasi Terbaik: 0.1252



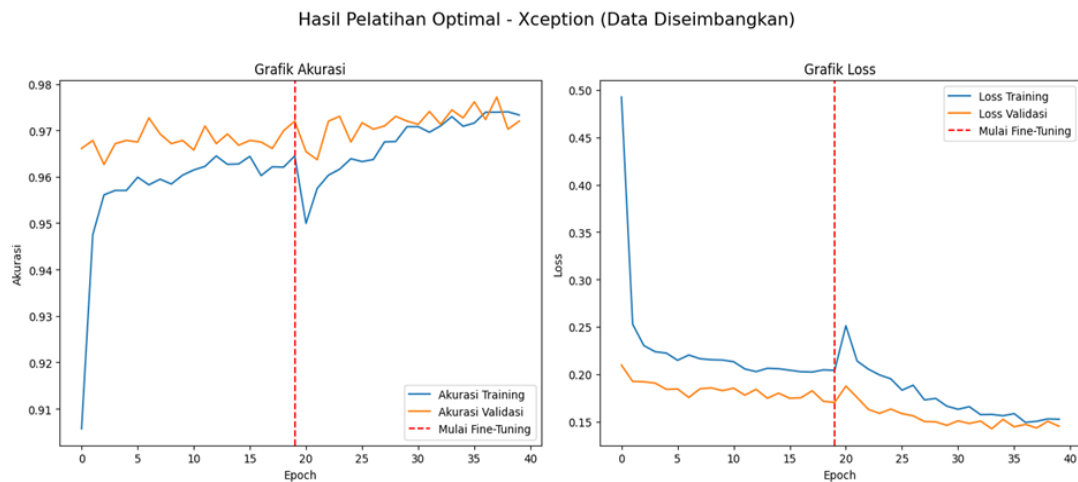
Gambar 3. Grafik Akurasi dan Loss InceptionV3

Grafik histori pelatihan untuk InceptionV3 (Gambar 3) menunjukkan konvergensi yang sangat baik. Setelah performa yang kuat di tahap *feature extraction*, model menunjukkan peningkatan lebih lanjut selama tahap *fine-tuning*. Hal ini dibuktikan dengan kurva akurasi validasi yang terus menanjak dan berhasil menyentuh nilai tertinggi di 97.54% pada salah epoch ke-29. Adapun model terbaik yang disimpan dan dievaluasi secara final menghasilkan akurasi validasi 97.30% dengan nilai loss yang sangat rendah. Ini mengindikasikan bahwa model memiliki kemampuan generalisasi yang kuat dan stabil.

Model Xception

Seperti InceptionV3, model Xception juga menggunakan ukuran citra input (299, 299). Pelatihan dilakukan dengan skenario yang sama.

- Akurasi Validasi Terbaik: 97.30%
- Loss Validasi Terbaik: 0.1453



Gambar 4. Grafik Akurasi dan Loss Xception

Dari log output, model Xception menunjukkan performa yang sangat kompetitif. Model ini mencapai akurasi validasi puncak di 97.72% pada epoch ke-38, meskipun model yang dievaluasi pada akhir proses memberikan akurasi 97.30% (kemungkinan karena *callback* *restore_best_weights* dari *EarlyStopping* berdasarkan *val_loss*). Grafik pelatihannya (Gambar 4) mengindikasikan bahwa model ini sangat efektif, dengan kurva akurasi yang menanjak tajam dan kurva loss yang menurun secara stabil, terutama setelah memasuki tahap *fine-tuning*.

Analisis Komparatif Kinerja Model

Untuk membandingkan kinerja ketiga arsitektur secara langsung, metrik utama (akurasi dan loss validasi) dari model terbaik dirangkum dalam tabel berikut.

Table 1. Perbandingan Kinerja Akhir Model

Arsitektur	Akurasi Validasi Terbaik	Loss Validasi Terbaik	Ukuran Input
Xception	97.30%	0.1453	(299, 299)
InceptionV3	97.30%	0.1252	(299, 299)
ResNet152V2	96.06%	0.1699	(224, 224)

Berdasarkan hasil pada Tabel 1, dapat ditarik beberapa kesimpulan utama:

- Performa Unggul: Arsitektur InceptionV3 dan Xception menunjukkan kinerja yang superior dan hampir identik, keduanya mencapai akurasi validasi sebesar 97.30%. Keduanya secara signifikan mengungguli ResNet152V2 yang mencapai akurasi 96.06%.
- Efisiensi Generalisasi: Meskipun memiliki akurasi yang sama, InceptionV3 berhasil mencapai loss validasi terendah (0.1252). Nilai loss yang lebih rendah mengindikasikan bahwa model InceptionV3 memiliki tingkat "kepastian" yang lebih tinggi dalam membuat prediksi yang benar dan menunjukkan kemampuan generalisasi yang sedikit lebih baik dibandingkan Xception.

- c. Pengaruh Arsitektur: Keunggulan InceptionV3 dan Xception berasal dari pendekatan arsitektur mereka yang efisien dalam melakukan konvolusi. InceptionV3 mencapai efisiensi melalui faktorisasi konvolusi, sedangkan Xception mengimplementasikan *depthwise separable convolutions*. Kedua teknik canggih ini memungkinkan model untuk belajar fitur yang kaya dengan parameter lebih sedikit, yang terbukti sangat efektif untuk dataset citra yang digunakan dalam penelitian ini..

PEMBAHASAN

Hasil penelitian ini menegaskan efektivitas metode transfer learning untuk tugas klasifikasi citra. Dengan memanfaatkan bobot pra-terlatih dari ImageNet, ketiga model mampu mencapai akurasi di atas 96% pada dataset yang relatif kecil (14.460 citra).

Strategi pelatihan dua tahap (*feature extraction* diikuti *fine-tuning*) terbukti sangat berhasil. Tahap *feature extraction* memungkinkan lapisan klasifikasi baru untuk beradaptasi dengan dataset target tanpa merusak fitur-fitur umum yang telah dipelajari oleh base model. Tahap *fine-tuning* dengan *learning rate* yang sangat kecil ($1e-5$) memungkinkan penyesuaian halus pada lapisan-lapisan top-level dari base model, yang secara konsisten meningkatkan performa validasi di ketiga skenario.

Implementasi *callbacks* seperti *ModelCheckpoint*, *EarlyStopping*, dan *ReduceLROnPlateau* memainkan peran krusial dalam mengoptimalkan proses pelatihan. *ModelCheckpoint* memastikan bahwa versi terbaik dari model disimpan, sementara *EarlyStopping* dan *ReduceLROnPlateau* secara efektif mencegah *overfitting* dan membantu model keluar dari local minima, sehingga menghemat waktu komputasi dan menghasilkan model yang lebih robust. Dari ketiga arsitektur yang diuji, InceptionV3 muncul sebagai model dengan keseimbangan terbaik antara akurasi tinggi dan loss terendah, menjadikannya pilihan paling optimal untuk studi kasus klasifikasi 10 spesies hewan ini. Xception memberikan performa yang sangat sebanding, sementara ResNet152V2, meskipun merupakan arsitektur yang sangat kuat, sedikit tertinggal dalam skenario perbandingan ini.

KESIMPULAN DAN REKOMENDASI

Berdasarkan analisis komparatif yang telah dilakukan terhadap arsitektur ResNet152V2, InceptionV3, dan Xception, penelitian ini menyimpulkan bahwa arsitektur InceptionV3 dan Xception menunjukkan performa yang superior dan sangat kompetitif, keduanya berhasil mencapai akurasi validasi terbaik sebesar 97.30%. Keduanya secara signifikan mengungguli arsitektur ResNet152V2 yang mencatatkan akurasi 96.06%. Meskipun memiliki akurasi yang identik dengan Xception, InceptionV3 menjadi model yang paling optimal dalam penelitian ini karena berhasil mencapai nilai loss validasi terendah (0.1252), yang mengindikasikan kemampuan generalisasi yang sedikit lebih baik. Selain itu, pendekatan pelatihan dua tahap yang menggabungkan *feature extraction* dan *fine-tuning* terbukti sangat efektif, dan implementasi *callbacks* seperti *ModelCheckpoint*, *EarlyStopping*, dan *ReduceLROnPlateau* memainkan peran vital dalam mencegah *overfitting* dan mencapai hasil yang robust.

Bagi para praktisi atau peneliti yang bekerja pada masalah klasifikasi citra hewan dengan karakteristik dataset yang serupa, arsitektur InceptionV3 direkomendasikan sebagai titik awal yang kuat karena keseimbangan optimal antara akurasi tinggi dan kemampuan generalisasi yang unggul. Arsitektur Xception juga merupakan alternatif yang sangat valid dengan kinerja yang hampir setara.

PENELITIAN LANJUTAN

Setiap penelitian memiliki keterbatasan. Penelitian ini terbatas pada dataset "Animals-10" yang telah diseimbangkan, yang mungkin tidak sepenuhnya mencerminkan tantangan dunia nyata seperti distribusi data yang tidak seimbang. Studi ini juga hanya menguji tiga arsitektur populer dan menggunakan *hyperparameter* yang diatur berdasarkan praktik umum, bukan melalui optimisasi sistematis. Untuk penelitian selanjutnya, disarankan untuk menggunakan dataset yang lebih besar dan lebih kompleks dengan jumlah kelas yang lebih banyak (misalnya, iNaturalist) dan mengeksplorasi teknik penanganan data tidak seimbang. Analisis komparatif dapat diperluas dengan menyertakan arsitektur yang lebih modern seperti EfficientNet atau Vision Transformer (ViT). Selain itu, penerapan teknik optimisasi *hyperparameter* secara sistematis (misalnya, *Bayesian Optimization*) dapat membantu memaksimalkan potensi setiap model. Terakhir, analisis efisiensi komputasi – seperti kecepatan inferensi, jumlah parameter, dan penggunaan memori – perlu dipertimbangkan untuk memberikan evaluasi yang lebih holistik, terutama untuk potensi penerapan pada perangkat dengan sumber daya terbatas.

DAFTAR PUSTAKA

- Abdel-Jaber, H., Devassy, D., Al Salam, A., Hidaytallah, L., & El-Amir, M. (2022). A Review of Deep Learning Algorithms and Their Applications in Healthcare. In *Algorithms* (Vol. 15, Issue 2). MDPI. <https://doi.org/10.3390/a15020071>
- Agarwal, N., Sondhi, A., Chopra, K., & Singh, G. (2021). Transfer learning: Survey and classification. *Advances in Intelligent Systems and Computing*, 1168, 145–155. https://doi.org/10.1007/978-981-15-5345-5_13
- Alzubaidi, L., Zhang, J., Humaidi, A. J., Al-Dujaili, A., Duan, Y., Al-Shamma, O., Santamaría, J., Fadhel, M. A., Al-Amidie, M., & Farhan, L. (2021). Review of deep learning: concepts, CNN architectures, challenges, applications, future directions. *Journal of Big Data*, 8(1). <https://doi.org/10.1186/s40537-021-00444-8>
- Dahr, J. M., & Gaafar, A. S. (2024). Performance Evaluation of the Convolutional Neural Networks for Object Identification Using RGB and Binary Images. *Informatica (Slovenia)*, 48(21), 177–188. <https://doi.org/10.31449/inf.v48i21.6568>
- Dwiatmoko, F., Utami, D., Sivi, N. A., Nahdlatul, U., & Lampung, U. (2024). *Klasifikasi Citra Sampah Organik dan Non Organik Menggunakan Algoritma CNN (Convolutional Neural Network)*.
- Hosna, A., Merry, E., Gyalmo, J., Alom, Z., Aung, Z., & Azim, M. A. (2022). Transfer learning: a friendly introduction. *Journal of Big Data*, 9(1). <https://doi.org/10.1186/s40537-022-00652-w>
- Huda, N., Mahiruna, A., Sulistijanti, W., Chandra Noor Santi, R., Teknologi Statistika dan Bisnis Muhammadiyah Semarang Jl DrHamka, I. K., Kota Semarang Jawa Tengah, N., & Stikubank Semarang Jl Tri Lomba Juang Mugassari Kota Semarang Jawa Tengah, U. (2023). Analisis Performa Inceptionv3 Convolutional Network Pada Klasifikasi Varietas Daun Grapevine Performance Analysis of InceptionV3 Convolutional Network Used for Grapevine Leaves Varieties Classification. In *Jurnal Sains Komputer dan Teknologi Informasi e-issn* (Vol. 5, Issue 2).
- Janiesch, C., Zschech, P., & Heinrich, K. (2021). *Machine learning and deep learning*. <https://doi.org/10.1007/s12525-021-00475-2/Published>
- Mathew, M. P., .M, S. E., VP, J., & Mahesh, T. Y. (2023). *Pretrained CNN Architectures: A Detailed Analysis Using Bell Pepper Image Datasets*. <https://doi.org/10.21203/rs.3.rs-3146418/v1>

- Mienye, I. D., & Swart, T. G. (2024). A Comprehensive Review of Deep Learning: Architectures, Recent Advances, and Applications. *Information (Switzerland)*, 15(12). <https://doi.org/10.3390/info15120755>
- Mohamed, A. H., Refaat, M., & Hemeida, A. M. (2022). Image classification based deep learning: A Review. *Aswan University Journal of Science and Technology*, 2(1). <https://journals.aswu.edu.eg/stjournal>
- Ridhovan, A., Suharso, A., Fakultas,), Komputer, I., Karawang, S., Ronggo Waluyo, J. H., Timur, T., & Karawang, K. (2022). *Penerapan Metode Residual Network (Resnet) Dalam Klasifikasi Penyakit Pada Daun Gandum*.
- Shafik, W., Tufail, A., Liyanage De Silva, C., & Awg Haji Mohd Apong, R. A. (2025). A novel hybrid inception-xception convolutional neural network for efficient plant disease classification and detection. *Scientific Reports*, 15(1), 3936. <https://doi.org/10.1038/s41598-024-82857-y>
- Sharifani, K., & Amini, M. (2023). Machine Learning and Deep Learning: A Review of Methods and Applications. In *World Information Technology and Engineering Journal* (Vol. 10). <https://ssrn.com/abstract=4458723>
- Xu, X., Zhao, M., Shi, P., Ren, R., He, X., Wei, X., & Yang, H. (2022). Crack Detection and Comparison Study Based on Faster R-CNN and Mask R-CNN. *Sensors*, 22(3). <https://doi.org/10.3390/s22031215>