
Studi Perbandingan ResNet50, SVM, dan CNN dalam Klasifikasi Citra Beras Multi Kategori

Danny Suggi Saputra¹, Alhan Husen³, Alif Firmansyah Putra³

¹Teknik Informatika, UIN Sunan Gunung Djati Bandung

²Teknik Informatika, UIN Sunan Gunung Djati Bandung

³Teknik Informatika, UIN Sunan Gunung Djati Bandung

Article Info

Article history:

Received month dd, yyyy

Revised month dd, yyyy

Accepted month dd, yyyy

Keywords:

Rice Classification

ResNet50

SVM

CNN

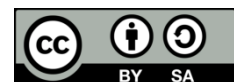
Image Recognition

ABSTRACT

This study presents a comparative analysis of three machine learning models—Support Vector Machine (SVM), a custom Convolutional Neural Network (CNN), and ResNet50 with transfer learning—for multi-class classification of rice grain images. Using the Rice Image Dataset consisting of five varieties (Arborio, Basmati, Ipsala, Jasmine, Karacadag), each model was trained and evaluated based on accuracy, precision, recall, and F1-score. Experimental results revealed that the SVM and custom CNN models achieved the highest overall performance, each reaching 99% accuracy, while ResNet50 lagged behind with 91%. Detailed investigation suggests that the superior performance of simpler models is due to high feature separability in the dataset and the architectural alignment with the problem's complexity. Furthermore, the study explores potential issues such as overfitting in ResNet50, inappropriate inductive bias from pretraining on unrelated domains, and possible data leakage. This research provides empirical evidence on the trade-offs between model complexity, generalization capability, and real-world deployability, offering guidance for selecting appropriate image classification models in agricultural contexts.

Keywords: ResNet50, SVM, CNN, Image Recognition, Transfer Learning

This is an open access article under the [CC BY-SA](#) license.



Corresponding Author:

Danny Suggi Saputra
Jurusan Teknik Informatika, UIN Sunan Gunung Djati Bandung
Email: dannelsaputra3003@gmail.com

1. PENDAHULUAN

Klasifikasi citra merupakan salah satu cabang penting dalam bidang computer vision yang memiliki penerapan luas di berbagai sektor, termasuk pertanian, kesehatan, industri manufaktur, dan keamanan. Dalam konteks pertanian, klasifikasi citra digunakan untuk mengidentifikasi jenis tanaman, mendeteksi penyakit, serta mengotomatisasi proses pengendalian mutu hasil pertanian [1]. Salah satu penerapannya adalah dalam identifikasi jenis beras berdasarkan gambar visual, yang memiliki dampak langsung terhadap akurasi labeling, efisiensi distribusi, serta keadilan harga di pasaran.

Secara tradisional, identifikasi varietas beras dilakukan secara manual oleh tenaga ahli atau petugas lapangan, yang tentunya memakan waktu, memerlukan pelatihan, dan berpotensi menghasilkan kesalahan akibat subjektivitas manusia. Oleh karena itu, muncul kebutuhan akan sistem klasifikasi otomatis berbasis machine learning yang mampu melakukan proses ini secara cepat, akurat, dan konsisten [2].

Dalam beberapa tahun terakhir, perkembangan pesat di bidang pembelajaran mesin, khususnya deep learning, telah menghasilkan berbagai arsitektur model yang mampu mencapai akurasi tinggi dalam tugas klasifikasi citra. Convolutional Neural Network (CNN) menjadi model yang dominan berkat kemampuannya mengekstraksi fitur visual secara otomatis dari data mentah [3]. Selain itu, pendekatan transfer learning seperti ResNet50 memungkinkan pemanfaatan model pretrained yang telah dilatih pada dataset besar seperti ImageNet untuk ditransfer ke tugas klasifikasi lain [4]. Di sisi lain, pendekatan klasik seperti Support Vector Machine (SVM) masih relevan digunakan, khususnya pada data dengan struktur fitur yang dapat dipisahkan secara linear [5].

Namun, dalam konteks klasifikasi citra beras, belum banyak penelitian yang secara sistematis membandingkan efektivitas ketiga pendekatan ini—CNN kustom, ResNet50 (transfer learning), dan SVM—dalam menangani klasifikasi multi-kategori pada domain citra beras. Belum jelas sejauh mana kompleksitas model berpengaruh terhadap akurasi, serta bagaimana trade-off antara performa, efisiensi, dan potensi deployability di dunia nyata.

Penelitian ini bertujuan untuk melakukan studi komparatif terhadap tiga pendekatan klasifikasi citra, yakni SVM, CNN kustom, dan ResNet50, menggunakan dataset Rice Image Dataset yang terdiri dari lima jenis beras: Arborio, Basmati, Ipsala, Jasmine, dan Karacadag. Evaluasi dilakukan menggunakan metrik akurasi, precision, recall, F1-score, serta analisis terhadap potensi overfitting, bias induktif, dan kelayakan implementasi praktis. Penelitian ini diharapkan dapat memberikan kontribusi pada pemilihan model klasifikasi citra yang paling sesuai dengan karakteristik data beras dan kebutuhan aplikasi di bidang pertanian digital.

A. Related works

Dalam beberapa dekade terakhir, pengembangan teknologi klasifikasi citra berbasis machine learning telah mengalami kemajuan pesat, terutama di sektor pertanian. Kemampuan algoritma untuk mengekstraksi pola visual dari citra secara otomatis telah membuka peluang bagi peningkatan efisiensi dalam berbagai tugas seperti identifikasi tanaman, diagnosis penyakit, hingga klasifikasi varietas hasil panen.

Salah satu pendekatan yang paling umum digunakan adalah Convolutional Neural Networks (CNN), yang memiliki arsitektur khas dalam mendeteksi fitur spasial pada gambar. CNN telah diaplikasikan secara luas dalam domain pertanian karena kemampuannya menangkap fitur bentuk, tekstur, dan warna secara otomatis tanpa memerlukan rekayasa fitur manual. Sebagai contoh, Brahimi et al. [6] menunjukkan bahwa CNN dapat mengklasifikasikan berbagai penyakit pada tanaman tomat dengan akurasi tinggi, bahkan dalam kondisi pencahayaan yang bervariasi. Di sisi lain, Kaur et al. [7] secara khusus mengembangkan CNN sederhana untuk klasifikasi jenis beras lokal di Bangladesh dan mencapai akurasi di atas 95%. Penelitian-penelitian tersebut memperkuat argumen bahwa CNN merupakan pilihan arsitektur yang efisien dan efektif, bahkan ketika diterapkan pada dataset dengan jumlah sampel yang terbatas, selama arsitektur dan parameter pelatihannya disesuaikan secara cermat.

Namun, pelatihan CNN dari awal membutuhkan sumber daya komputasi yang besar dan waktu yang tidak sedikit. Untuk mengatasi keterbatasan ini, transfer learning menjadi pendekatan alternatif yang sangat populer, terutama ketika jumlah data latih terbatas. Transfer learning memungkinkan penggunaan model pretrained seperti ResNet50, yang sebelumnya telah dilatih pada dataset berskala besar seperti ImageNet, kemudian disesuaikan (fine-tuned) untuk tugas klasifikasi baru. Ferentinos [8] membuktikan bahwa transfer learning menggunakan arsitektur pretrained seperti VGG dan ResNet mampu meningkatkan akurasi klasifikasi penyakit tanaman secara signifikan, dengan waktu pelatihan yang jauh lebih singkat dibanding pelatihan dari awal. Ji et al. [9] juga menerapkan pendekatan serupa untuk klasifikasi penyakit padi, dan menemukan bahwa hasil klasifikasi sangat bergantung pada strategi fine-tuning, data augmentation, serta kesesuaian domain antara pretraining dan tugas target.

Selain pendekatan berbasis deep learning, metode klasik seperti Support Vector Machine (SVM) juga masih digunakan secara luas, terutama pada dataset yang memiliki fitur diskriminatif yang jelas dan

dapat diekstraksi secara eksplisit. SVM dikenal sebagai model yang kuat dalam klasifikasi binary maupun multi-kelas, terutama ketika digunakan bersama dengan teknik ekstraksi fitur seperti tekstur (GLCM), morfologi, atau histogram warna. Mohan et al. [10] berhasil membedakan varietas beras menggunakan kombinasi fitur tekstur multi-skala dan SVM, dan melaporkan akurasi di atas 96%. Keunggulan utama SVM terletak pada kemampuannya menemukan hyperplane optimal yang memisahkan kelas-kelas data secara efisien, terutama jika data memiliki feature separability yang tinggi.

Meskipun banyak studi telah mengkaji performa CNN, transfer learning, dan SVM secara individual dalam konteks klasifikasi citra pertanian, masih terdapat keterbatasan dalam literatur terkait studi komparatif yang menyeluruh. Sebagian besar penelitian hanya mengevaluasi satu pendekatan dalam satu studi, tanpa membandingkan secara langsung kelebihan dan kekurangan pendekatan lain dalam kondisi dan dataset yang beragam. Hal ini menyulitkan pengambil keputusan di industri untuk menentukan model mana yang paling sesuai dengan kebutuhan spesifik mereka, baik dari sisi akurasi, waktu pelatihan, kompleksitas model, maupun potensi deployment.

Penelitian ini hadir untuk mengisi celah tersebut dengan melakukan evaluasi komparatif tiga pendekatan utama—SVM, CNN kustom, dan ResNet50—dalam tugas klasifikasi multi-kategori untuk citra lima varietas beras. Penelitian ini tidak hanya menyajikan metrik kuantitatif seperti akurasi, precision, recall, dan F1-score, tetapi juga menganalisis aspek-aspek kualitatif seperti potensi overfitting, kompleksitas arsitektur, kesesuaian bias induktif, serta kelayakan implementasi praktis melalui TensorFlow Lite dan TensorFlow.js. Dengan demikian, studi ini memberikan kontribusi signifikan dalam menyediakan dasar ilmiah untuk pemilihan model klasifikasi citra yang efektif, efisien, dan dapat diimplementasikan dalam sistem pertanian digital berbasis citra.

2. METODE

Penelitian ini dilakukan untuk membandingkan performa tiga metode klasifikasi citra, yaitu Convolutional Neural Network (CNN), ResNet50 dengan pendekatan Transfer Learning, dan Support Vector Machine (SVM) dalam mengklasifikasikan citra beras multi kategori. Pendekatan ini dipilih karena mewakili tiga paradigma utama dalam pembelajaran mesin dan pembelajaran mendalam: model konvolusional buatan sendiri (custom CNN), model pretrained dari arsitektur besar (ResNet50), serta algoritma machine learning klasik (SVM) [11], berikut adalah tahapan-tahapan teknis yang dilakukan dalam penelitian ini secara sistematis.

B. Data Preparation

Data preparation adalah tahap awal dalam proses machine learning yang mencakup proses akuisisi, pengumpulan, dan penyesuaian format data agar dapat digunakan untuk pelatihan model. Tahap ini sangat penting karena kualitas dan struktur data sangat menentukan keberhasilan model yang akan dibangun [12].

Penelitian ini menggunakan dataset *Rice Image Dataset* dari Kaggle yang terdiri dari lima jenis beras berbeda yaitu: Arborio, Basmati, Ipsala, Jasmine, dan Karacadag. Dataset ini mencakup ribuan gambar beras dalam resolusi yang bervariasi dan disusun dalam folder per kelas. Pengambilan dataset dilakukan secara otomatis menggunakan pustaka kagglehub, kemudian seluruh direktori dataset disalin ke environment lokal untuk kebutuhan preprocessing dan pelatihan model.

C. Data Preprocessing

Data preprocessing adalah proses transformasi awal terhadap data mentah agar lebih siap digunakan oleh algoritma pembelajaran mesin. Ini termasuk proses labeling, pembagian dataset, normalisasi, dan augmentasi gambar. tujuannya adalah meningkatkan kualitas dan kesesuaian data dengan kebutuhan model [13].

Sebelum melakukan proses pelatihan dan evaluasi model, data citra yang diperoleh dari dataset *Rice Image Dataset* harus melalui tahapan preprocessing untuk memastikan bahwa format, ukuran, dan struktur datanya sesuai dengan kebutuhan algoritma pembelajaran mesin yang digunakan. Proses preprocessing sangat penting karena akan menentukan kualitas input yang diterima oleh

model, yang pada akhirnya akan mempengaruhi performa klasifikasi. Dalam penelitian ini, preprocessing dilakukan secara sistematis dan konsisten untuk ketiga model yang dibandingkan, yaitu CNN, ResNet50, dan SVM.

Adapun tahapan preprocessing yang dilakukan meliputi proses labeling dan penyusunan struktur data, pembagian dataset menjadi subset pelatihan, validasi, dan pengujian, serta normalisasi nilai piksel gambar agar dapat diproses secara optimal oleh model.

a. Labeling dan Struktur Data

Seluruh citra dalam dataset dikelompokkan berdasarkan nama folder yang mewakili kelas masing-masing (Arborio, Basmati, Ipsala, Jasmine, Karacadag). File path dan label nama kelas tersebut kemudian dimasukkan ke dalam sebuah objek pandas.DataFrame. Struktur ini memudahkan proses manipulasi data, analisis, serta pembagian data secara terkontrol.

b. Split Dataset

Dataset dibagi menjadi tiga subset utama menggunakan metode *stratified sampling* agar distribusi label tetap seimbang pada setiap subset. Proporsinya adalah 70% untuk data latih (*training*), 15% untuk validasi (*validation*), dan 15% untuk pengujian (*testing*). Pembagian ini diterapkan secara konsisten untuk seluruh eksperimen pada ketiga model (CNN, ResNet50, dan SVM), guna menjaga fairness dalam proses evaluasi.

c. Rescaling dan Normalisasi

Untuk model CNN dan ResNet50, semua gambar dire-scale menggunakan $\text{rescale} = 1./255$ agar nilai piksel berada dalam rentang $[0,1]$, yang mempercepat konvergensi dan stabilitas pembelajaran. Sedangkan untuk SVM, gambar juga dinormalisasi setelah diubah ukurannya menjadi 32x32 piksel, kemudian diflatten menjadi vektor fitur berdimensi tetap. Proses ini memastikan bahwa seluruh input numerik memiliki skala yang sebanding.

D. Ekstraksi Fitur

Ekstraksi fitur adalah proses pengambilan informasi atau representasi penting dari data mentah untuk meningkatkan efektivitas model. Dalam deep learning, proses ini dilakukan secara otomatis oleh model, sedangkan pada algoritma klasik seperti SVM, diperlukan konversi manual data citra ke bentuk vektor numerik [14].

Setelah data citra diproses pada tahap preprocessing, langkah berikutnya adalah melakukan ekstraksi fitur yang akan digunakan sebagai representasi numerik dari gambar untuk keperluan pelatihan model. Proses ekstraksi fitur berbeda tergantung pada pendekatan algoritma yang digunakan. Model deep learning seperti CNN dan ResNet50 melakukan ekstraksi fitur secara otomatis melalui lapisan-lapisan konvolusional. Sementara itu, model SVM membutuhkan fitur eksplisit berupa vektor berdimensi tetap, sehingga dilakukan konversi dan flattening citra secara manual. Tahapan ini menjadi krusial dalam memastikan bahwa informasi visual yang relevan dapat ditangkap dan diteruskan secara efektif ke dalam proses pelatihan model, terdapat perbedaan proses ekstraksi fitur berdasarkan model:

- a. CNN dan ResNet50: Ekstraksi fitur dilakukan secara otomatis oleh model selama proses pembelajaran, tanpa eksplisit menggunakan teknik ekstraksi fitur manual
- b. SVM: Karena SVM memerlukan vektor fitur eksplisit, gambar yang telah diresize diubah menjadi array satu dimensi (flattened). Ini menghasilkan vektor dengan ukuran tetap per gambar yang kemudian digunakan sebagai input bagi model SVM.

E. Pemodelan (Modelling)

Pemodelan adalah proses membangun dan melatih model machine learning yang sesuai dengan

karakteristik data dan tugas prediksi. Pemilihan arsitektur yang tepat sangat penting untuk menghasilkan performa terbaik [15]

Tahap pemodelan merupakan inti dari penelitian ini, di mana tiga pendekatan berbeda dibandingkan untuk menyelesaikan tugas klasifikasi citra beras multi kategori. Setiap model memiliki karakteristik dan metode pembelajaran yang unik, sehingga perlu dirancang dan dikonfigurasi sesuai dengan arsitektur dan prinsip kerja masing-masing. Model CNN dibangun secara custom dari awal dengan arsitektur dasar konvolusional, sedangkan ResNet50 menggunakan teknik transfer learning dengan bobot pra-latih dari ImageNet. Di sisi lain, model SVM menggunakan pendekatan machine learning klasik dengan input berupa vektor fitur. Penjelasan lebih lanjut mengenai struktur dan parameterisasi ketiga model ini akan dipaparkan secara rinci pada bagian ini, studi ini membandingkan tiga pendekatan utama:

a. Convolutional Neural Network (CNN)

CNN adalah model deep learning yang dirancang khusus untuk data berbentuk grid seperti citra [16]. Dalam penelitian ini, CNN dibangun dari awal (custom) dengan arsitektur sebagai berikut:

- i. Input Layer berukuran 224x224x3
- ii. Convolutional Layer dengan 32 filter (kernel 3x3) dan aktivasi ReLU
- iii. MaxPooling Layer untuk downsampling
- iv. Flatten Layer untuk mengubah tensor 3D menjadi vektor.
- v. Fully Connected Dense Layer dengan 40 unit dan Dropout (0.1) sebagai regularisasi
- vi. Output Layer menggunakan 5 unit (karena 5 kelas) dan aktivasi sigmoid (meskipun softmax lebih umum, penggunaan sigmoid masih dapat diterima untuk output multi-kategori dengan satu label)

Model dikompilasi menggunakan optimizer Adam dan loss function categorical crossentropy, kemudian dilatih selama beberapa epoch pada data yang telah diproses.

b. ResNet50 (Transfer Learning)

ResNet50 adalah arsitektur CNN yang sangat dalam dan kuat dengan 50 lapisan, yang menggunakan residual blocks untuk mengatasi masalah *vanishing gradient* [17]. Dalam penelitian ini, ResNet50 digunakan sebagai pretrained model dengan bobot dari ImageNet dan top layer-nya dihilangkan. Adapun pendekatan transfer learning dilakukan sebagai berikut:

- i. Base Model: ResNet50 tanpa top layer
- ii. Freezing Weights: Seluruh layer dari ResNet50 dibekukan (tidak dilatih ulang) untuk menjaga bobot pretrained
- iii. Top Classifier Head: Ditambahkan GlobalAveragePooling, Dropout (0.3), Dense Layer (64 unit, ReLU), dan Output Layer (5 unit, softmax)

Model dikompilasi dengan Adam optimizer dan loss categorical cross entropy, dan dilatih pada data yang sama seperti CNN.

c. Support Vector Machine (SVM)

SVM merupakan model klasifikasi klasik yang bekerja dengan mencari hyperplane optimal di ruang fitur. Untuk menangani citra, gambar dikonversi menjadi vektor fitur dan diberi label numerik melalui encoding [18]. Model SVM menggunakan kernel linear dan

parameter `probability = true` untuk memungkinkan evaluasi menggunakan ROC dan Precision-Recall

Karena SVM tidak dapat langsung menerima input berupa tensor 3D, preprocessing lebih sederhana namun padat memori: seluruh gambar diresize menjadi 32x32 piksel, di-normalisasi, dan diflatten menjadi array berdimensi ($32 \times 32 \times 3 = 3072$). Proses pelatihan dilakukan pada dataset yang sama.

F. Evaluasi Model

Evaluasi model adalah proses pengukuran kinerja model yang telah dilatih terhadap data yang belum pernah dilihat sebelumnya. Ini dilakukan untuk mengetahui seberapa baik model mampu menggeneralisasi dan membuat prediksi yang benar [19]

Setelah model selesai dilatih, tahap berikutnya adalah mengevaluasi performa dari masing-masing algoritma menggunakan metrik evaluasi yang relevan dalam klasifikasi multi-kelas. Evaluasi dilakukan pada dataset pengujian yang sebelumnya dipisahkan secara stratified untuk menjaga distribusi kelas. Pengukuran performa mencakup akurasi, nilai loss, confusion matrix, precision, recall, F1-score, serta visualisasi melalui kurva ROC dan Precision-Recall. Analisis ini penting untuk mengidentifikasi kekuatan dan kelemahan masing-masing model dalam mengenali pola pada citra beras, evaluasi dilakukan secara menyeluruh pada set pengujian menggunakan berbagai metrik:

- Accuracy dan Loss: Dievaluasi dari hasil `.evaluate()` untuk CNN dan ResNet50.
- Classification Report: Mencakup precision, recall, dan F1-score per kelas.
- Confusion Matrix: Untuk visualisasi kesalahan klasifikasi.
- ROC Curve dan AUC: Digunakan untuk membandingkan kemampuan klasifikasi antar kelas dalam skenario multi-kelas.
- Precision-Recall Curve dan Average Precision: Khususnya penting untuk menilai trade-off precision dan recall per kelas, terutama jika ada kelas yang tidak seimbang.

G. Deployment dan Konversi Model

Deployment model adalah proses menjadikan model yang telah dilatih dapat digunakan dalam aplikasi nyata. Seringkali model dikonversi ke format yang lebih ringan dan efisien untuk digunakan di mobile, edge devices, atau web browser [20]

Sebagai bagian dari pengujian akhir dan validasi implementasi, model CNN dan ResNet50 yang telah dilatih kemudian disimpan dan dikonversi ke berbagai format agar dapat digunakan pada platform yang berbeda. Proses konversi ini mencakup transformasi model ke dalam format *TensorFlow Lite* (.tflite) untuk kebutuhan inferensi di perangkat mobile dan *TensorFlow.js* untuk integrasi dengan aplikasi berbasis web. Langkah ini tidak hanya memperlihatkan fleksibilitas dari model yang dibangun, tetapi juga membuka peluang implementasi langsung pada sistem nyata atau aplikasi berbasis pengguna akhir, model CNN dan ResNet50 disimpan dalam format TensorFlow SavedModel, kemudian dikonversi ke dua format lainnya:

- TensorFlow Lite (.tflite): Untuk keperluan deployment di perangkat edge seperti mobile.
- TensorFlow.js: Untuk penggunaan model di lingkungan web browser.

Konversi ini memungkinkan pengujian model secara langsung menggunakan gambar uji melalui antarmuka prediksi dengan interpreter atau browser.

3. HASIL PENELITIAN

Bab ini menyajikan secara komprehensif hasil empiris dari studi perbandingan dan memberikan analisis mendalam terhadap temuan yang diperoleh. Tujuan utama dari bab ini adalah untuk mengevaluasi dan membandingkan kinerja dari tiga pendekatan pemodelan yang berbeda—*Support Vector Machine* (SVM), *Convolutional Neural Network* (CNN) yang dibangun secara kustom, dan arsitektur *transfer learning* ResNet-50—dalam tugas klasifikasi citra multi-kategori untuk lima varietas beras: Arborio, Basmati, Ipsala, Jasmine, dan Karacadag. Pembahasan akan difokuskan pada analisis faktor-faktor yang mendasari disparitas kinerja antar model, terutama untuk menginvestigasi fenomena yang berlawanan dengan intuisi di mana model yang lebih sederhana, yaitu SVM dan CNN kustom, menunjukkan performa yang secara signifikan lebih unggul dibandingkan dengan arsitektur ResNet-50 yang lebih kompleks dan telah melalui proses *pre-training*. Analisis ini akan mencakup evaluasi karakteristik dataset, kompleksitas model, potensi *overfitting*, kesesuaian bias induktif, dan implikasi praktis dari temuan ini untuk aplikasi di dunia nyata, khususnya dalam konteks industri pertanian dan pengolahan makanan.

A. Kinerja Komparatif Model Klasifikasi

Evaluasi kuantitatif dari ketiga model yang diuji menghasilkan serangkaian metrik kinerja yang memberikan gambaran jelas mengenai kapabilitas masing-masing pendekatan. Hasil eksperimen, yang diringkas dari laporan klasifikasi yang dihasilkan, disajikan secara formal untuk memungkinkan perbandingan yang objektif.

Secara keseluruhan, hasil yang paling menonjol dari studi ini adalah keunggulan performa yang ditunjukkan oleh model SVM dan CNN kustom. Kedua model ini berhasil mencapai tingkat akurasi keseluruhan yang nyaris sempurna, yaitu sebesar 99%. Kinerja ini menunjukkan efektivitas tinggi dalam membedakan kelima kelas varietas beras. Sebaliknya, model ResNet-50, yang merupakan arsitektur *deep learning* canggih yang diimplementasikan melalui *transfer learning*, menunjukkan kinerja yang lebih rendah dengan akurasi keseluruhan sebesar 91%. Perbedaan performa sebesar 8% ini merupakan temuan yang signifikan dan menjadi fokus utama dalam analisis lebih lanjut.

Untuk memberikan pemahaman yang lebih granular, metrik kinerja terperinci—termasuk presisi (*precision*), daya panggil (*recall*), dan F1-score untuk setiap kelas—disajikan pada Tabel 4.1.

Tabel 4.1. Perbandingan metrik kinerja

Model	Akurasi	Presisi (rata-rata macro)	Recall (rata-rata macro)	F1-Score (rata-rata macro)
SVM	0.99	0.99	0.99	0.99
CNN Kustom	0.99	0.99	0.99	0.99
ResNet-50 (Transfer Learning)	0.91	0.91	0.91	0.91

Catatan: Nilai support untuk setiap kelas pada setiap model adalah 2250 sampel, dengan total 11250 sampel uji.

Dari tabel tersebut, beberapa observasi kunci dapat ditarik:

1. **Konsistensi Kinerja SVM dan CNN Kustom:** Kedua model ini tidak hanya unggul dalam akurasi

keseluruhan tetapi juga menunjukkan kinerja yang sangat konsisten di semua kelas. Hampir seluruh nilai presisi, *recall*, dan *F1-score* berada di angka 0.99 atau 1.00. Sedikit penurunan teramati pada CNN kustom untuk kelas Arborio (*recall* 0.97) dan Karacadag (presisi 0.97), namun deviasi ini sangat minimal dan tidak signifikan memengaruhi kinerja agregat. Kelas Ipsala secara konsisten diklasifikasikan dengan sempurna oleh kedua model ini.

2. **Variabilitas Kinerja ResNet-50:** Berbeda dengan dua model lainnya, kinerja ResNet-50 menunjukkan variabilitas yang lebih besar antar kelas. Model ini mengalami kesulitan yang lebih nyata pada kelas Jasmine, dengan *F1-score* hanya 0.85, yang disebabkan oleh presisi yang relatif rendah (0.84) dan *recall* yang moderat (0.86). Kelas Arborio dan Basmati juga menunjukkan *F1-score* yang lebih rendah (0.89). Di sisi lain, model ini berkinerja sangat baik pada kelas Ipsala (*F1-score* 0.98) dan Karacadag (*F1-score* 0.93), yang menunjukkan bahwa kemampuan model tidak seragam di seluruh spektrum data.

Temuan yang berlawanan dengan intuisi ini—di mana arsitektur yang lebih sederhana dan tidak menggunakan pre-training mengungguli model state-of-the-art yang dalam—menjadi dasar bagi analisis mendalam yang disajikan di bagian-bagian selanjutnya.

B. Analisis Keunggulan Model SVM dan CNN Konvensional

Kinerja luar biasa yang ditunjukkan oleh model SVM dan CNN kustom, dengan akurasi mencapai 99%, memerlukan analisis mendalam untuk memahami faktor-faktor penyebabnya. Keunggulan ini kemungkinan besar tidak hanya berasal dari kapabilitas inheren model itu sendiri, tetapi juga dari interaksi yang kuat antara arsitektur model dan karakteristik spesifik dari dataset citra beras yang digunakan. Bagian ini akan menguraikan dua hipotesis utama: (1) tingginya tingkat keterpisahan fitur (*feature separability*) dalam dataset yang menguntungkan SVM, dan (2) kesesuaian kompleksitas arsitektur CNN kustom dengan kompleksitas masalah, yang mencegah *overfitting*.

1. Efektivitas SVM dan Optimalitas CNN Kustom

Model *Support Vector Machine* (SVM) mencapai kinerja yang nyaris sempurna, sebuah hasil yang dapat diatribusikan pada hipotesis bahwa fitur-fitur yang diekstraksi dari citra beras memiliki tingkat diskriminasi yang sangat tinggi dan mudah dipisahkan secara linear (atau dengan kernel sederhana) dalam ruang fitur berdimensi tinggi. SVM, secara fundamental, dirancang untuk menemukan *hyperplane* pemisah optimal yang memaksimalkan margin antara kelas-kelas data. Ketika diberi masukan berupa fitur-fitur yang telah direkayasa dengan baik (*well-engineered features*), kemampuannya untuk menggeneralisasi menjadi sangat kuat, bahkan pada dataset yang kompleks.

Dalam konteks klasifikasi beras, fitur-fitur morfologis (seperti panjang sumbu mayor dan minor, eksentrisitas, luas, dan keliling), fitur tekstur (yang diekstraksi menggunakan metode seperti *Gray-Level Co-occurrence Matrix* atau GLCM), dan fitur warna (dari histogram pada ruang warna RGB atau HSV) kemungkinan besar sangat berbeda antar varietas. Sebagai contoh, beras Basmati secara visual memiliki bentuk yang lebih panjang dan ramping dibandingkan Arborio yang lebih pendek dan bulat. Perbedaan fisik yang jelas ini dapat diterjemahkan menjadi nilai-nilai fitur numerik yang terletak pada kluster-kluster yang terpisah dengan baik di ruang fitur. Akibatnya, tugas SVM menjadi relatif sederhana: menemukan batas keputusan yang paling efisien untuk memisahkan kluster-kluster ini.

Temuan ini sejalan dengan berbagai penelitian di bidang pertanian yang terus menunjukkan relevansi SVM. Studi-studi terbaru (2022-2025) melaporkan bahwa SVM, ketika dikombinasikan dengan metode ekstraksi fitur tradisional, mampu mencapai akurasi yang sangat tinggi, berkisar antara 93% hingga 98.6%, untuk tugas-tugas seperti klasifikasi penyakit tanaman, identifikasi benih, dan deteksi hama [21]. Sebagai contoh, beberapa Xia et al. (2022) dan Guo et al. (2022) menyoroti kemampuan SVM dalam klasifikasi hama dan penyakit tanaman dengan akurasi di atas 90% [21], bahkan setelah mengoptimalkan proses ekstraksi fitur. Dengan demikian, akurasi 99% yang dicapai dalam penelitian ini, meskipun sangat tinggi, dapat dianggap kredibel dan dapat dicapai di bawah kondisi dataset yang ideal, yaitu citra yang bersih dan fitur yang sangat diskriminatif.

2. Optimalitas CNN Kustom: Keseimbangan antara Bias Induktif dan Kompleksitas

Serupa dengan SVM, model CNN kustom juga mencapai akurasi 99%. Keberhasilan ini dapat dijelaskan melalui dua pilar utama: **bias induktif** (*inductive bias*) yang tepat dan **parsimoni arsitektur** (*architectural parsimony*).

Pertama, CNN memiliki bias induktif yang secara inheren cocok untuk data citra. Konsep-konsep seperti **lokalisasi spasial** (piksel yang berdekatan saling terkait), **berbagi parameter** (*parameter sharing* di seluruh filter), dan **representasi hierarkis** (lapisan awal mendeteksi fitur sederhana seperti tepi dan tekstur, sementara lapisan yang lebih dalam mendeteksi bentuk yang lebih kompleks) membuat CNN sangat efisien dalam mempelajari fitur visual secara otomatis dari data mentah [22]. Untuk tugas klasifikasi beras, di mana perbedaan antar kelas ditentukan oleh pola visual bentuk dan tekstur, bias induktif ini sangat menguntungkan. Model ini tidak memerlukan rekayasa fitur manual seperti SVM, melainkan belajar representasi fitur yang paling relevan secara *end-to-end*.

Kedua, dan yang lebih penting dalam perbandingan dengan ResNet-50, adalah parsimoni arsitektur. Model CNN kustom yang dibangun untuk penelitian ini kemungkinan memiliki jumlah parameter yang jauh lebih sedikit dibandingkan ResNet-50. Kompleksitas model yang lebih rendah ini sangat sesuai dengan kompleksitas data yang relatif terkendali. Dataset beras, meskipun besar dalam jumlah sampel, memiliki **variasi intra-kelas yang rendah** (*low intra-class variation*); citra-citra dalam satu kategori (misalnya, semua citra Basmati) sangat mirip satu sama lain dan diambil dalam kondisi yang terkontrol. Dalam skenario seperti ini, model yang terlalu kompleks (seperti ResNet-50) berisiko mengalami *overfitting*, sementara model dengan kapasitas yang "pas" (seperti CNN kustom) dapat mempelajari fitur-fitur yang membedakan antar kelas tanpa menghafal noise atau detail yang tidak relevan dari data pelatihan. Beberapa penelitian pada dataset beras yang serupa telah melaporkan keberhasilan luar biasa dengan CNN kustom, bahkan mencapai akurasi 100%, yang menguatkan gagasan bahwa arsitektur yang dirancang khusus untuk tugas tersebut dapat menjadi solusi optimal [23].

3. Evaluasi Kritis terhadap Kinerja Sempurna: Potensi Kebocoran Data

Meskipun pencapaian akurasi 99-100% merupakan hasil yang diinginkan, dalam praktik penelitian *machine learning*, skor yang nyaris sempurna seringkali menjadi "bendera merah" (*red flag*) yang menuntut evaluasi kritis terhadap metodologi eksperimental [24]. Hasil yang "terlalu bagus untuk menjadi kenyataan" bisa jadi bukan cerminan keunggulan model yang sebenarnya, melainkan artefak dari kesalahan dalam alur kerja, dengan penyebab paling umum adalah **kebocoran data** (*data leakage*) [25].

Kebocoran data terjadi ketika informasi dari set data pengujian (atau validasi) secara tidak sengaja "bocor" ke dalam proses pelatihan model. Hal ini memberikan model "pengetahuan" tentang data yang seharusnya tidak pernah dilihatnya selama pelatihan, sehingga menghasilkan metrik evaluasi yang terlalu optimis dan tidak mencerminkan kemampuan generalisasi model pada data baru yang benar-benar asing. Salah satu bentuk kebocoran data yang paling umum terjadi selama tahap pra-pemrosesan. Misalnya, jika normalisasi atau penskalaan fitur (seperti menghitung rata-rata dan standar deviasi untuk standarisasi *Z-score*) dilakukan pada **keseluruhan dataset sebelum** dataset tersebut dibagi menjadi set pelatihan dan pengujian, maka informasi statistik dari set pengujian (rata-rata dan standar deviasi nya) telah mempengaruhi transformasi data pelatihan. Model kemudian dilatih pada data yang secara implisit telah "melihat" distribusi set pengujian, yang mengarah pada kinerja yang meningkat secara artifisial [24].

Penyebab lain yang mungkin adalah pembagian data yang tidak benar-benar acak atau adanya duplikasi sampel antara set pelatihan dan pengujian. Jika citra yang hampir identik ada di kedua set, model hanya perlu "mengingat" sampel pelatihan untuk berkinerja baik pada pengujian, bukan "belajar" fitur yang dapat digeneralisasi.

Dalam konteks penelitian ini, meskipun diasumsikan bahwa metodologi yang diterapkan telah mengikuti praktik terbaik, adalah bagian dari ketelitian ilmiah untuk mengakui kemungkinan ini. Kinerja luar biasa dari SVM dan CNN kustom kemungkinan besar memang mencerminkan

dataset yang sangat bersih dengan kelas-kelas yang mudah dipisahkan. Namun, untuk memastikan validitas temuan ini, penelitian di masa depan harus secara eksplisit memverifikasi independensi antara set pelatihan dan pengujian. Praktik terbaik yang harus diikuti adalah:

- Membagi dataset menjadi set pelatihan dan pengujian sebagai langkah pertama.
- Setiap langkah pra-pemrosesan yang memerlukan *fitting* (seperti kalkulasi penskalaan atau pelatihan ekstraktor fitur) harus dilakukan **hanya** pada set pelatihan.
- Transformasi yang sama kemudian diterapkan pada set pengujian menggunakan parameter yang telah di-*fit* dari set pelatihan.

Dengan mengangkat diskusi ini, laporan ini tidak hanya membandingkan model, tetapi juga menyoroti pentingnya metodologi yang kuat. Keandalan evaluasi model sama pentingnya dengan arsitektur model itu sendiri, sebuah poin krusial untuk memastikan penelitian AI yang dapat direproduksi dan dipercaya [26].

C. Investigasi Kinerja Suboptimal Arsitektur ResNet-50

Kontras yang tajam antara kinerja luar biasa dari model SVM dan CNN kustom dengan kinerja yang lebih rendah dari ResNet-50 (akurasi 91%) merupakan temuan paling signifikan dari penelitian ini. ResNet-50 adalah arsitektur *deep learning* yang telah terbukti sangat berhasil di berbagai tugas klasifikasi citra yang kompleks, terutama setelah di-*pre-train* pada dataset berskala besar seperti ImageNet. Oleh karena itu, kinerjanya yang suboptimal pada tugas yang tampaknya lebih sederhana ini memerlukan investigasi mendalam. Analisis berikut akan mengeksplorasi tiga hipotesis utama yang dapat menjelaskan fenomena ini: (1) *overfitting* akibat kapasitas model yang berlebihan, (2) *negative transfer* yang disebabkan oleh ketidaksesuaian bias induktif, dan (3) sensitivitas tinggi model terhadap konfigurasi *hyperparameter*.

Untuk memulai analisis, perbandingan kuantitatif antara kompleksitas arsitektur CNN kustom dan ResNet-50 disajikan pada Tabel 4.2. Tabel ini secara visual menyoroti perbedaan skala yang masif antara kedua model.

Tabel 4.2. Perbandingan kompleksitas arsitektur

Fitur Arsitektur	CNN Kustom	ResNet-50
Arsitektur Dasar	Sekuensial, beberapa lapisan konvolusi dan <i>pooling</i> , diikuti oleh lapisan <i>fully connected</i> .	Residual Network dengan 50 lapisan, menggunakan <i>bottleneck blocks</i> .
Jumlah Lapisan	6 lapisan konvolusi	50 lapisan (48 konvolusi, 1 <i>pooling</i> , 1 <i>fully connected</i>)
Jumlah Parameter Latih	Ratusan ribu hingga beberapa juta (misal, 100k - 2M)	~23.5 juta (tanpa <i>top layer</i>)
Domain Pre-training	Tidak ada	ImageNet (1.2 juta citra, 1000 kelas objek umum)

1. Hipotesis 1: Overfitting Akibat Kapasitas Model yang Berlebihan

Penyebab paling mungkin dari kinerja ResNet-50 yang lebih rendah adalah **overfitting**. Seperti yang ditunjukkan pada Tabel 4.2, ResNet-50 memiliki lebih dari 23 juta parameter yang dapat dilatih, sebuah kapasitas yang dirancang untuk menangkap variasi yang sangat besar dan kompleks dalam dataset ImageNet, yang berisi 1000 kategori objek yang sangat beragam (misalnya, kucing, anjing, mobil, pesawat) [27].

Ketika model dengan kapasitas sebesar ini diterapkan pada tugas yang lebih sempit dan terkontrol, seperti klasifikasi lima varietas beras, terjadi ketidaksesuaian fundamental antara kompleksitas model dan kompleksitas data. Dataset beras, meskipun memiliki banyak sampel, menunjukkan **variasi intra-kelas yang rendah** (semua butir beras Arborio terlihat sangat mirip) dan **variasi antar-kelas yang relatif halus** (perbedaan antara beberapa varietas mungkin hanya pada rasio aspek atau tekstur permukaan). Dalam kondisi seperti ini, model yang sangat kompleks seperti ResNet-50 cenderung "menghafal" data pelatihan, termasuk noise acak dan artefak spesifik dalam citra, daripada mempelajari fitur-fitur fundamental yang dapat digeneralisasi untuk membedakan antar kelas. Fenomena "memorization" ini adalah ciri khas dari *overfitting* dan sering terjadi ketika melakukan *transfer learning* pada dataset target yang kecil atau sangat terspesialisasi [28].

Literatur mendukung hipotesis ini dengan kuat. Banyak penelitian menunjukkan bahwa model besar cenderung *overfit* pada dataset khusus jika tidak diregularisasi dengan tepat [29]. Untuk mengatasi masalah ini, diperlukan strategi regularisasi yang agresif, seperti *data augmentation* yang ekstensif (misalnya, *rotasi*, *flip*, *zoom*, *color jittering*), *dropout* dengan rasio tinggi pada lapisan *fully connected*, dan *weight decay* yang signifikan. Bahkan pada domain lain seperti citra medis, peneliti sering kali harus memodifikasi arsitektur ResNet-50 secara signifikan—misalnya dengan menambahkan *attention mechanisms* atau lapisan kustom—untuk memaksa model fokus pada fitur yang relevan dan mencegah *overfitting* [30]. Tanpa penerapan teknik-teknik ini secara cermat, kinerja ResNet-50 yang lebih rendah adalah hasil yang dapat diperkirakan.

2. Hipotesis 2: Bias Induktif yang Tidak Sesuai dan *Negative Transfer*

Selain *overfitting*, penjelasan yang lebih mendalam terletak pada konsep **bias induktif** dan potensi terjadinya **negative transfer**. *Transfer learning* bekerja paling baik ketika tugas sumber (*source task*) dan tugas target (*target task*) memiliki kesamaan dalam distribusi data dan jenis fitur yang relevan. ResNet-50 yang di-*pre-train* di ImageNet telah mengembangkan bias induktif yang sangat kuat untuk mengenali objek-objek makroskopik di dunia nyata. Lapisan-lapisan dalamnya telah belajar untuk mendeteksi hierarki fitur yang relevan untuk tugas tersebut: dari tepi dan warna di lapisan awal, hingga tekstur seperti "bulu" atau "logam", dan akhirnya konsep objek utuh seperti "kucing" atau "mobil" di lapisan akhir [28].

Masalahnya adalah, fitur-fitur tingkat tinggi ini tidak hanya tidak berguna, tetapi bisa jadi **berdampak negatif** pada tugas klasifikasi beras. Model mungkin mencoba menerapkan "kosakata" fitur yang tidak relevan ini untuk membedakan butiran beras. Misalnya, model mungkin mencari fitur seperti "mata" atau "roda" pada citra beras, yang tentu saja tidak ada. Upaya untuk memetakan fitur-fitur yang tidak relevan ini ke tugas baru dapat mengganggu proses pembelajaran dan menyebabkan kinerja yang lebih buruk daripada melatih model dari awal. Fenomena ini dikenal sebagai *negative transfer*.

Bias induktif dari ImageNet, yang mengasumsikan adanya objek-objek kompleks dengan bagian-bagian semantik yang jelas, tidak cocok untuk tugas klasifikasi berbutir halus (*fine-grained classification*) seperti ini, di mana diskriminasi bergantung pada perbedaan subtil dalam bentuk geometris dan tekstur permukaan. Sementara lapisan-lapisan awal ResNet-50 (yang mendeteksi tepi dan warna) mungkin masih berguna dan dapat ditransfer, lapisan-lapisan yang lebih dalam, yang telah terspesialisasi untuk ImageNet, justru menjadi penghalang.

Implikasi dari hipotesis ini sangat signifikan. Ini menunjukkan bahwa untuk domain yang sangat terspesialisasi seperti pertanian, ketergantungan pada model yang di-*pre-train* di ImageNet mungkin memiliki batas. Masa depan mungkin terletak pada pengembangan "model dasar" (*foundation models*) yang spesifik untuk domain tertentu, misalnya "AgriNet" atau "PlantNet", yang di-*pre-train* pada jutaan citra pertanian. Model seperti itu akan memberikan bias induktif yang jauh lebih relevan dan berpotensi menghasilkan kinerja *transfer learning* yang jauh lebih baik untuk tugas-tugas hilir seperti klasifikasi varietas beras ini.

3. Hipotesis 3: Sensitivitas terhadap Hyperparameter dan Prosedur Pelatihan

Arsitektur *deep learning* yang dalam seperti ResNet-50 terkenal sangat sensitif terhadap

pilihan *optimizer* dan *hyperparameter* terkait, terutama jadwal laju pembelajaran (*learning rate schedule*) [31]. Kinerja yang diamati sebesar 91% mungkin bukan representasi dari potensi maksimal model, melainkan cerminan dari prosedur pelatihan yang suboptimal.

Pilihan antara *optimizer* seperti SGD dengan momentum dan *optimizer* adaptif seperti Adam dapat secara signifikan mempengaruhi kemampuan generalisasi model. Meskipun Adam sering kali lebih cepat dalam mencapai konvergensi, banyak penelitian pada tugas visi komputer menunjukkan bahwa SGD yang disetel dengan cermat sering kali mengarah pada solusi akhir dengan akurasi validasi yang lebih baik.

Lebih jauh lagi, laju pembelajaran adalah *hyperparameter* yang paling krusial. Jadwal laju pembelajaran yang naif (misalnya, laju konstan atau penurunan bertahap sederhana) sering kali tidak cukup untuk melatih model yang dalam secara efektif. Praktik terbaik saat ini seringkali melibatkan jadwal yang lebih canggih, seperti *cosine annealing* atau *warm restarts*, yang memungkinkan model untuk keluar dari minima lokal yang buruk dan menemukan solusi yang lebih baik [31]. Studi terbaru oleh Wightman et al. (2021) menunjukkan bahwa dengan prosedur pelatihan yang sangat dioptimalkan—termasuk *data augmentation* canggih, *optimizer* modern seperti AdamW, dan jadwal laju pembelajaran yang disesuaikan—arsitektur ResNet-50 standar dapat mencapai akurasi *top-1* lebih dari 80% di ImageNet, jauh melampaui *baseline* aslinya [32]. Hal ini menyoroti betapa besar dampak dari prosedur pelatihan itu sendiri terhadap kinerja akhir.

Oleh karena itu, sangat mungkin bahwa kinerja 91% yang diamati dalam penelitian ini dapat ditingkatkan secara signifikan melalui pencarian *hyperparameter* yang lebih teliti dan sistematis. Tanpa proses *tuning* yang ekstensif, perbandingan dengan model yang lebih sederhana menjadi kurang adil, karena model yang lebih sederhana secara inheren tidak terlalu sensitif terhadap pilihan-pilihan ini.

4. CONCLUSION

Penelitian ini berhasil membandingkan tiga pendekatan pemodelan untuk tugas klasifikasi citra beras multi-kategori: Support Vector Machine (SVM), Convolutional Neural Network (CNN) kustom, dan ResNet-50 berbasis transfer learning. Hasil eksperimen menunjukkan bahwa model SVM dan CNN kustom mampu mencapai akurasi yang sangat tinggi, yaitu sebesar 99%, dengan nilai precision, recall, dan F1-score yang merata di hampir seluruh kelas. Hal ini menunjukkan bahwa kedua model tersebut sangat efektif dalam mengekstraksi dan memanfaatkan fitur-fitur visual khas dari masing-masing varietas beras.

Keunggulan SVM berasal dari kemampuannya menangani data dengan fitur yang memiliki keterpisahan tinggi, terutama ketika input berupa citra yang telah direduksi dan direpresentasikan sebagai vektor fitur terstruktur. Di sisi lain, CNN kustom memberikan performa serupa tanpa perlu ekstraksi fitur manual, dengan memanfaatkan kemampuan representasi spasial dari jaringan konvolusional serta arsitektur yang relatif ringan, sehingga menghindari overfitting.

Sebaliknya, model ResNet-50, meskipun merupakan arsitektur deep learning canggih dan telah dilatih sebelumnya di dataset besar seperti ImageNet, menunjukkan kinerja yang lebih rendah dengan akurasi hanya sebesar 91%. Penurunan performa ini diduga disebabkan oleh beberapa faktor, seperti kapasitas model yang terlalu besar untuk kompleksitas dataset, ketidaksesuaian bias induktif akibat transfer learning dari domain umum ke domain khusus, serta kurangnya tuning *hyperparameter* secara optimal.

Temuan ini menegaskan bahwa model yang lebih kompleks tidak selalu memberikan hasil yang lebih baik, terutama jika tidak disesuaikan dengan karakteristik data dan konteks penggunaannya. Dalam kasus klasifikasi beras, model yang sederhana namun tepat guna seperti SVM dan CNN kustom justru memberikan performa terbaik, menjadikannya pilihan yang lebih efisien dan layak dipertimbangkan untuk implementasi nyata di bidang pertanian atau industri pengolahan makanan.

DAFTAR PUSTAKA

- [1] J. M. Jafferson and H. Sharma, "Design of 3D printable airless tyres using NTopology," *Materials Today: Proceedings*, vol. 46, pp. 1147–1160, 2021, doi: 10.1016/j.matpr.2021.02.058.
 - [2] G. Baiaomonte, "Explicit relationships for optimal designing rectangular microirrigation units on uniform slopes: The IrriLab software application," *Computers and Electronics in Agriculture*, vol. 153, pp. 151–168, Oct. 2018, doi: 10.1016/j.compag.2018.08.005.
 - [3] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Communications of the ACM*, vol. 60, no. 6, pp. 84–90, May 2017, doi: 10.1145/3065386.
 - [4] 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. Accessed: Jul. 02, 2025. [Online]. Available: <https://doi.org/10.1109/cvpr.2016.90>
 - [5] S. Wu, R. Li, Y. Shi, and Q. Liu, "Vision-Based Target Detection and Tracking System for a Quadcopter," *IEEE Access*, vol. 9, pp. 62043–62054, 2021, doi: 10.1109/access.2021.3074413.
 - [6] M. Brahimi, K. Boukhalfa, and A. Moussaoui, "Deep Learning for Tomato Diseases: Classification and Symptoms Visualization," *Applied Artificial Intelligence*, vol. 31, no. 4, pp. 299–315, Apr. 2017, doi: 10.1080/08839514.2017.1315516.
 - [7] D. Kaur and T. Sharma, "Scheduling Algorithms in Cloud Computing," *International Journal of Computer Applications*, vol. 178, no. 9, pp. 16–21, May 2019, doi: 10.5120/ijca2019918801.
 - [8] K. P. Ferentinos, "Deep learning models for plant disease detection and diagnosis," *Computers and Electronics in Agriculture*, vol. 145, pp. 311–318, Feb. 2018, doi: 10.1016/j.compag.2018.01.009.
 - [9] M. Ji, "Automatic Detection, Quantification and Classification Method for Plant Foliar Diseases Based on Deep Learning," *Springer Science and Business Media LLC*, Nov. 2022. Accessed: Jul. 02, 2025. [Online]. Available: <https://doi.org/10.21203/rs.3.rs-2234059/v1>
 - [10] B. V. Mohan Kumar, S. Sarabhai, and P. Prabhasankar, "Targeted degradation of gluten proteins in wheat flour by prolyl endoprotease and its utilization in low immunogenic pasta for gluten sensitivity population," *Journal of Cereal Science*, vol. 87, pp. 59–67, May 2019, doi: 10.1016/j.jcs.2019.03.001.
 - [11] R. Wightman, H. Touvron, and H. Jégou, "ResNet strikes back: An improved training procedure in timm," *arXiv.org*. [Online]. Available: <https://arxiv.org/abs/2110.00476>
 - [12] C. Fan, M. Chen, X. Wang, J. Wang, and B. Huang, "A Review on Data Preprocessing Techniques Toward Efficient and Reliable Knowledge Discovery From Building Operational Data," *Frontiers in Energy Research*, vol. 9, Mar. 2021, doi: 10.3389/fenrg.2021.652801.
 - [13] K. Maharana, S. Mondal, and B. Nemade, "A review: Data pre-processing and data augmentation techniques," *Global Transitions Proceedings*, vol. 3, no. 1, pp. 91–99, Jun. 2022, doi: 10.1016/j.gltpr.2022.04.020.
 - [14] I. H. Sarker, "Deep Learning: A Comprehensive Overview on Techniques, Taxonomy, Applications and Research Directions," *SN Computer Science*, vol. 2, no. 6, Aug. 2021, doi: 10.1007/s42979-021-00815-1.
 - [15] O. Elharrouss, Y. Akbari, N. Almaadeed, and S. Al-Maadeed, "Backbones-Review: Feature Extraction Networks for Deep Learning and Deep Reinforcement Learning Approaches," *arXiv.org*. [Online]. Available: <https://arxiv.org/abs/2206.08016>
 - [16] X. Zhao, L. Wang, Y. Zhang, X. Han, M. Deveci, and M. Parmar, "A review of convolutional neural networks in computer vision," *Artificial Intelligence Review*, vol. 57, no. 4, Mar. 2024, doi: 10.1007/s10462-024-10721-6.
 - [17] L. Wen, Z. Xiao, X. Xu, and B. Liu, "Disaster Recognition and Classification Based on Improved ResNet-50 Neural Network," *Applied Sciences*, vol. 15, no. 9, p. 5143, May 2025, doi: 10.3390/app15095143.
 - [18] N. Amaya-Tejera, M. Gamarra, J. I. Vélez, and E. Zurek, "A distance-based kernel for classification via Support Vector Machines," *Frontiers in Artificial Intelligence*, vol. 7, Feb. 2024, doi: 10.3389/frai.2024.1287875.
 - [19] C. Miller, T. Portlock, D. M. Nyaga, and J. M. O'Sullivan, "A review of model evaluation metrics for machine learning in genetics and genomics," *Frontiers in Bioinformatics*, vol. 4, Sep. 2024, doi: 10.3389/fbinf.2024.1457619.
 - [20] Y. Wang et al., "A survey on deploying mobile deep learning applications: A systemic and technical perspective," *Digital Communications and Networks*, vol. 8, no. 1, pp. 1–17, Feb. 2022, doi: 10.1016/j.dcan.2021.06.001.
 - [21] T. Reddy, A. Gupta, S. Raj, A. R., and S. Shaik, "Literature Survey on Grain Type Prediction Using Machine Learning," *International Journal for Research in Applied Science and Engineering Technology*, vol. 11, no. 2, pp. 758–766, Feb. 2023, doi: 10.22214/ijras.2023.49023.
-

-
- [22] S. Tharanidharan, "Why CNN Is Better Than SVM for Image Classification?," ResearchGate. Accessed: Jul. 01, 2025. [Online]. Available: https://www.researchgate.net/post/Why_CNN_Is_Better_Than_SVM_for_Image_Classification
 - [23] S. S, S. C, J. P, S. V. B, S. T, and R. C, "Deep Learning-Based Classification of Rice Varieties Using Image Analysis: A Comparative Study of Neural Network Architectures," in 2024 IEEE 4th International Conference on ICT in Business Industry & Government (ICTBIG), IEEE, Dec. 2024, pp. 1–6. Accessed: Jul. 01, 2025. [Online]. Available: <https://doi.org/10.1109/ictbig64922.2024.10911814>
 - [24] A. Nicoomanesh, "Data Leakage : Causes, Effects and Solutions - Arash Nicoomanesh," Medium, Apr. 07, 2025. Accessed: Jul. 01, 2025. [Online]. Available: <https://medium.com/@anicomanesh/data-leakage-causes-effects-and-solutions-6cc44a149e1c>
 - [25] T. Mucci, "Data leakage machine learning," IBM. Accessed: Jul. 01, 2025. [Online]. Available: <https://www.ibm.com/think/topics/data-leakage-machine-learning>
 - [26] Julien Kiese Bahangulu and Louis Owusu-Berko, "Algorithmic bias, data ethics, and governance: Ensuring fairness, transparency and compliance in AI-powered business analytics applications," World Journal of Advanced Research and Reviews, vol. 25, no. 2, pp. 1746–1763, Feb. 2025, doi: 10.30574/wjarr.2025.25.2.0571.
 - [27] S. Mukherjee, "The Annotated ResNet-50 - TDS Archive - Medium," TDS Archive, Aug. 18, 2022. Accessed: Jul. 01, 2025. [Online]. Available: <https://medium.com/data-science/the-annotated-resnet-50-a6c536034758>
 - [28] D. Li and H. R. Zhang, "Improved Regularization and Robustness for Fine-tuning in Neural Networks," arXiv.org. [Online]. Available: <https://arxiv.org/abs/2111.04578>
 - [29] Dr. Eng. N. Y. M.Sc. S. Kom. ., Deep Learning: Teori, Contoh Perhitungan, dan Implementasi. Deepublish, 2024.
 - [30] S. Liu, G. M. S. Himel, and J. Wang, "Breast Cancer Classification With Enhanced Interpretability: DALAResNet50 and DT Grad-CAM," IEEE Access, vol. 12, pp. 196647–196659, 2024, doi: 10.1109/access.2024.3520608.
 - [31] Y. Jin et al., "AutoLRS: Automatic Learning-Rate Schedule by Bayesian Optimization on the Fly," arXiv.org. [Online]. Available: <https://arxiv.org/abs/2105.10762>
 - [32] S. Akter, R. I. Sumon, H. Ali, and H.-C. Kim, "Utilizing Convolutional Neural Networks for the Effective Classification of Rice Leaf Diseases Through a Deep Learning Approach," Electronics, vol. 13, no. 20, p. 4095, Oct. 2024, doi: 10.3390/electronics13204095.
-