



SPECTA Journal of Technology

E-ISSN : 2622-9099

P-ISSN : 2549-2713

Homepage jurnal: <https://journal.itk.ac.id/index.php/sjt>



Klasifikasi Penyakit Daun Tanaman Menggunakan MobileNetV4 Berbasis *Transfer Learning*

Syifa Maulida¹, Tito Ariffianto Miftahul Huda², Wafiq Ajriyah³, Rizky Amelia⁴, Gusti Ahmad Fanshuri Alfariy^{5*}

^{12345*}Program Studi Informatika, Jurusan Matematika dan Teknologi Informasi, Fakultas Sains dan Teknologi Informasi, Institut Teknologi Kalimantan, Balikpapan, Indonesia

*Corresponding email: gusti.alfariy@lecturer.itk.ac.id

Received: 3/July/2025
Accepted: 2/April/2026

Revised: 9/September/2025
Published: 31/August/2026

To cite this article:

Maulida, S., Huda, T. A. M., Ajriyah, W., Amelia, R & Alfariy, G. A. F (2026). Klasifikasi Penyakit Daun Tanaman menggunakan MobileNetV4 berbasis Transfer Learning. *SPECTA Journal of Technology*, 10 (2), 116 - 131. [10.35718/specta.v10i2.8481436](https://doi.org/10.35718/specta.v10i2.8481436)

Abstract

Digital image-based classification of plant leaf diseases is crucial for reducing agricultural losses and improving crop yields. This study applies the MobileNetV4 architecture with transfer learning to the PlantVillage dataset, which contains 20,600 images from 15 disease categories. MobileNetV4 was chosen because it is efficient in recognizing visual objects with a relatively small number of parameters, making it suitable for devices with limited resources. Trials were conducted on three variants of the MobileNetV4 model, namely Small, Medium, and Large, to compare them based on accuracy, F1-score, efficiency, and inference time. The results show that all variants achieve accuracy above 96% with an average F1-score of 0.99, indicating that the large-capacity model is more accurate but requires high computation, while the light-capacity model remains competitive with fast inference. For practical implementation, the model is integrated into a web interface using Streamlit, which enables real-time classification. These findings confirm the potential of MobileNetV4 as an accurate and efficient solution for plant disease detection on limited devices.

Keywords : Deep Learning, Convolutional Neural Network (CNN), MobileNetV4, Transfer Learning, Klasifikasi Penyakit Tanaman, Citra Digital

Abstrak

Klasifikasi penyakit daun tanaman berbasis citra digital merupakan salah satu pendekatan penting dalam upaya mengurangi kerugian di sektor pertanian dan meningkatkan hasil produksi tanaman. Penelitian ini menerapkan arsitektur MobileNetV4 dengan metode *transfer learning* pada dataset PlantVillage, yang terdiri atas 20.600 citra dari 15 kategori penyakit tanaman. MobileNetV4 dipilih karena memiliki efisiensi yang baik dalam mengenali objek visual dengan jumlah parameter yang relatif kecil, sehingga sesuai untuk diterapkan pada perangkat dengan sumber daya terbatas. Pengujian dilakukan pada tiga varian model MobileNetV4, yaitu Small, Medium, dan Large, untuk membandingkan kinerjanya berdasarkan akurasi, *F1-score*, efisiensi, dan waktu inferensi. Hasil penelitian menunjukkan bahwa seluruh varian mencapai tingkat akurasi di atas 96% dengan rata-rata *F1-score* sebesar 0,99. Hasil tersebut menunjukkan bahwa model dengan kapasitas besar memiliki tingkat akurasi yang lebih tinggi, tetapi membutuhkan komputasi yang lebih besar, sedangkan model dengan kapasitas ringan tetap menunjukkan kinerja

yang kompetitif dengan waktu inferensi yang lebih cepat. Untuk mendukung implementasi praktis, model diintegrasikan ke dalam antarmuka web menggunakan Streamlit, sehingga memungkinkan proses klasifikasi dilakukan secara *real-time*. Temuan ini mengonfirmasi potensi MobileNetV4 sebagai solusi yang akurat dan efisien untuk deteksi penyakit tanaman pada perangkat dengan sumber daya terbatas.

Kata kunci: *Deep Learning*, *Convolutional Neural Network (CNN)*, *MobileNetV4*, *Transfer Learning*, Klasifikasi Penyakit Tanaman, Citra Digital.

1. Pendahuluan

Penyakit tanaman merupakan ancaman besar terhadap hasil pertanian dan menjadi salah satu isu global, dikarenakan dapat menurunkan kualitas dan kuantitas panen secara signifikan. Isu ini tidak hanya berhubungan langsung dengan peningkatan produksi pertanian, melainkan berhubungan dengan kualitas hasil panen yang terjamin bebas dari penyakit. Dalam hal itu, Mohanty et al. menyebutkan bahwa lebih dari 80% produksi pertanian dihasilkan oleh petani kecil di negara berkembang, padahal petani kecil ini sangat rentan terhadap kerugian akibat penyakit tanaman. Lebih dari 50% kerugian hasil pertanian yang disebabkan oleh serangan hama dan penyakit tanaman, tentunya berdampak pada ekonomi petani serta ketahanan pangan secara keseluruhan (Mohanty et al., 2016).

Saat ini, teknologi seperti pengolahan citra digital dan *Artificial Intelligence (AI)* telah mengalami kemajuan yang sangat pesat dan membuka peluang untuk klasifikasi penyakit tanaman secara otomatis. CNN unggul dalam mengenali pola visual dan telah digunakan untuk berbagai studi klasifikasi penyakit daun dengan akurasi tinggi. Namun berbagai tantangan masih perlu dihadapi, seperti pada tahap implementasi di lapangan, di mana perangkat yang digunakan seringkali memiliki keterbatasan, seperti hanya menggunakan *smartphone* atau perangkat portabel lainnya dengan kapasitas komputasi yang rendah (Nayak et al., 2024). Oleh karena itu, dibutuhkan arsitektur CNN yang ringan, efisien, dan akurat.

Pada penelitian sebelumnya telah memanfaatkan arsitektur ringan seperti MobileNetV2 dan MobileNetV3, yang terbukti efisien untuk klasifikasi citra pertanian (Dwijayana & Wibawa, 2022; Zhu et al., 2023). MobileNetV4, sebagai arsitektur CNN terbaru, dirancang khusus untuk beroperasi pada perangkat dengan spesifikasi rendah, seperti *smartphone* atau perangkat kecil lainnya. Model ini dibuat dengan gabungan dua pendekatan, yaitu desain jaringan yang dirancang langsung oleh manusia dan sistem otomatis yang disebut *Neural Architecture Search (NAS)*. Berdasarkan penelitian terbaru dari Qin et al. telah menunjukkan keunggulan MobileNetV4 pada klasifikasi citra umum, namun belum ada penerapan secara sistematis untuk klasifikasi penyakit daun tanaman, khususnya dengan membandingkan tiga varian (*Small*, *Medium*, dan *Large*) dalam konteks *transfer learning*, masih jarang dilakukan. Hal ini menjadi celah untuk penelitian yang ingin dijawab dalam studi ini.

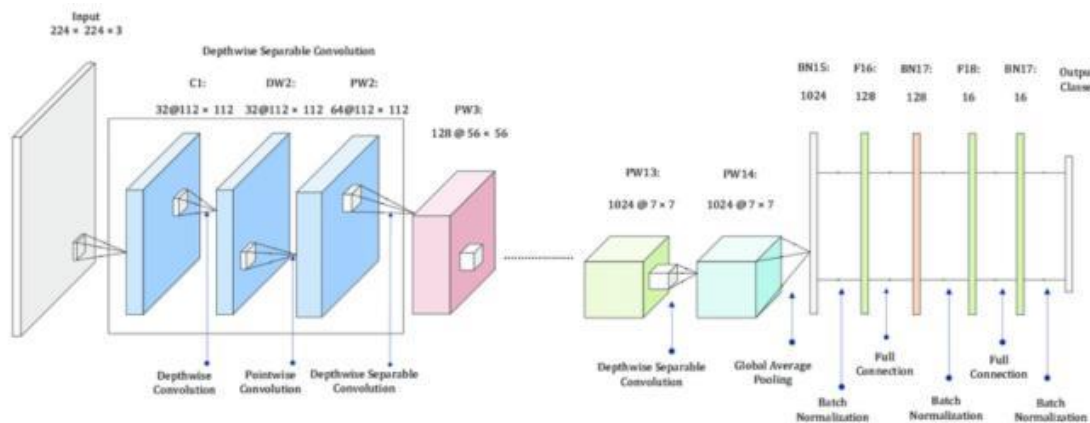
Dalam penelitian ini, MobileNetV4 diimplementasikan dengan pendekatan *transfer learning*, di mana model akan dilatih dari nol dengan memanfaatkan bobot awal yang telah diperoleh dari pelatihan pada dataset ImageNet. Proses pelatihan dilakukan dengan menyesuaikan bagian akhir arsitektur model supaya dapat mengenali dan mengklasifikasi jenis-jenis penyakit daun sesuai dengan karakteristik dataset yang digunakan dalam studi ini. Pendekatan ini dinilai lebih efisien dari segi waktu dan mampu meningkatkan akurasi, terutama ketika jumlah data latihnya terbatas. Strategi ini telah terbukti efektif dalam penelitian sebelumnya yang dilakukan oleh Qin et al. (2024).

Tujuan dari penelitian ini adalah menerapkan MobileNetV4 dalam mengklasifikasi penyakit daun tanaman secara otomatis berdasarkan citra digital, dan memanfaatkan pendekatan *transfer learning* untuk mempercepat proses pelatihan, serta meningkatkan kinerja model. Diharapkan hasil dari penelitian ini dapat memberikan kontribusi nyata dalam pengembangan teknologi pertanian digital yang efisien dan juga akurat untuk klasifikasi penyakit daun tanaman.

2. Landasan Teori

2.1 Convolutional Neural Network (CNN)

Convolutional Neural Network (CNN) merupakan algoritma *deep learning* yang unggul dalam pengenalan pola visual, terutama pada pengolahan citra, karena terbukti mampu mengenali dan mengklasifikasikan gambar dengan akurasi yang tinggi, serta dapat menangkap pola-pola penting dalam gambar tanpa harus membuat fitur secara manual (Lu et al., 2021). Arsitektur CNN memiliki struktur yang relatif sederhana, yang terdiri dari satu lapisan masukan (*input layer*), beberapa lapisan tersembunyi (*hidden layer*), dan satu lapisan keluaran (*output layer*), serta menggunakan representasi tiga dimensi yang mencakup lebar (*width*), tinggi (*height*), dan kedalaman (*depth*). Adapun gambar arsitektur CNN sebagai berikut.



Gambar 1. Arsitektur CNN

Sumber: Sinaga & Sembiring, 2025

2.2 MobileNetV4

MobileNetV4 dapat dikategorikan sebagai versi terbaru dari seri MobileNet yang dirancang untuk perangkat dengan kapasitas komputasi terbatas, seperti *smartphone* atau alat kecil lainnya. Walaupun memiliki arsitektur yang ringan, MobileNetV4 tetap menawarkan performa yang bagus. Model ini dikembangkan dengan gabungan dua pendekatan, yaitu desain jaringan yang dirancang langsung oleh manusia dan sistem otomatis yang dikenal dengan istilah *Neural Architecture Search* (NAS). Beberapa fitur dari MobileNetV2 dan MobileNetV3 juga masih digunakan karena terbukti efektif.

2.2.1 Universal Inverted Bottleneck (UIB)

Salah satu komponen penting di MobileNetV4 adalah blok yang dinamakan *Universal Inverted Bottleneck* (UIB). Blok ini menggabungkan beberapa teknik dari model-model sebelumnya, seperti *Inverted Residual* dari MobileNetV2, gaya ConvNeXt, sampai sistem jaringan yang biasa digunakan pada *Vision Transformer*. Fleksibilitas ini memungkinkan MobileNetV4 untuk disesuaikan dengan beragam kebutuhan perangkat keras dan batasan komputasi. Selain itu, dua *depthwise convolution* tambahan memperluas area reseptif model dengan tetap menjaga efisiensi komputasi.

Struktur dasar operasi dibalik UIB adalah *depthwise separable convolution* yang terdiri dari *depthwise convolution* dan *pointwise convolution* (1x1 Conv). Jika X adalah *tensor input* dan Y adalah *tensor output* dari blok UIB, prosesnya secara konseptual dapat dirumuskan pada Persamaan 1.

$$Y = P \circ (D_2)^\circ \circ E \circ (D_1)^\circ(X) \quad (1)$$

Dimana:

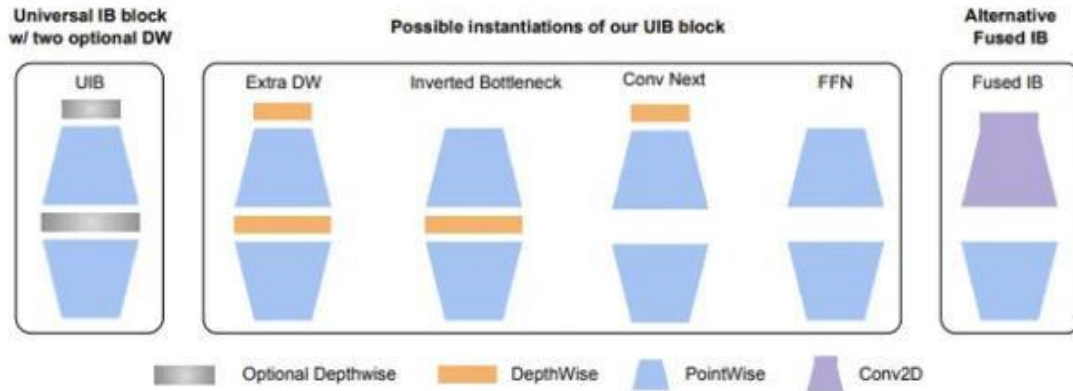
P : *Pointwise-projection* (1x1) yang dapat dipilih oleh NAS

E : *Pointwise-projection* (1x1)

D_1, D_2 : Depthwise conv optional

◦ : Fungsi komposisi

$()^\circ$: Layer opsional



Gambar 2. Ilustrasi Arsitektur Universal *Inverted Bottleneck* (IB)

Sumber: Qin et al. ,2024

2.2.2. Mobile Multi-Query Attention (MQA)

MobileNetV4 memperkenalkan mekanisme atensi baru, yaitu blok *Mobile Multi-Query Attention* (MQA). Blok ini membuat model agar bisa fokus ke bagian-bagian penting pada gambar dengan cara yang lebih efisien dibandingkan metode *attention* biasa.

MQA menyederhanakan *Multi-Head-Self Attention* (MHSA) dengan berbagai *keys* dan *values* di seluruh *heads*. Ketika jumlah *batched tokens* kecil dibandingkan dengan dimensi fitur, berbagi satu *head* di seluruh *keys* dan *values* secara signifikan mengurangi kebutuhan memori *bandwidth*, yang akan meningkatkan *Operational Intensity*. Penelitian yang dilakukan oleh Qin et al. (2024) ini mengkonfirmasi keunggulan MQA pada model *hybrid*, dengan percepatan mencapai lebih dari 39% pada EdgeTPU dan Samsung S23 GPU, disertai penurunan kualitas yang minimal, yaitu sebesar -0,03% dibandingkan MHSA. Selain itu, MQA juga mengurangi *Multiply-Accumulate Operations* (MAC) dan parameter model >25%.

Mekanisme model MQA dapat dijelaskan melalui fungsi $Mobile_{MQA}$. Fungsi tersebut didefinisikan dengan $Mobile_{MQA}(X) = Concat(attention_1, \dots, attention_N)W^0$ dimana N adalah jumlah *attention heads*, dan untuk setiap *attention head* j . Persamaan *attention* sendiri dapat dilihat pada Persamaan 2.

$$attention_j = softmax\left(\frac{(XW^{Qj})(SR(X)W^K)^T}{\sqrt{d_k}}\right)(SR(X)W^V) \quad (2)$$

Dimana:

X : Tensor masukan

W^{Qj} : Matriks bobot untuk *query head* ke- j

$SR(X)$: Fungsi *Spatial Reduction* yang diterapkan pada X . Ini dapat berupa *stride-2 3x3 DW* untuk reduksi spasial atau fungsi identitas jika reduksi spasial tidak digunakan.

W^K : Matriks bobot untuk *keys* (dibagi antar *heads*)

W^V : Matriks bobot untuk *values* (dibagi antar *heads*).

d_k : Dimensi dari *keys*.

W^0 : Matriks bobot keluaran

2.2.3. Neural Architecture Search (NAS)

MobileNetV4 menggunakan pendekatan *Neural Architecture Search* (NAS) dua tahap yang dioptimalkan, yaitu pencarian kasar (*coarse search*) untuk menentukan ukuran filter DWConv paling optimal, dan pencarian halus (*fine search*) untuk mengeksplorasi konfigurasi lapisan depthwise. Pendekatan ini merupakan evolusi dari arsitektur MobileNet sebelumnya, dimana MobileNetV1 memperkenalkan *depthwise separable convolutions* untuk efisiensi mobile (Howard et al., 2017), MobileNetV2 menambahkan *inverted residuals* dan *linear bottlenecks* (Sandler et al., 2018), sementara MobileNetV3 menggunakan AutoML untuk menemukan arsitektur yang optimal (Howard et al., 2019). Proses NAS ini dibantu oleh penggunaan *offline distillation* dataset dan teknik distilasi baru yang melibatkan pencampuran dataset dinamis (*dynamic dataset mixing*). Teknik ini sejalan dengan pendekatan EfficientNet (Tan & Le, 2019) yang menggunakan *compound scaling* dan *Once-for-All* (Cai et al., 2020) yang mengoptimalkan *deployment* untuk berbagai perangkat. Kombinasi ini menghasilkan serangkaian model MobileNetV4 yang sebagian besar Pareto-Optimal, yang artinya model tersebut memiliki keseimbangan terbaik antara akurasi dan efisiensi di berbagai perangkat seluler.

2.2.4. Spesifikasi Arsitektur MobileNetV4

MobileNetV4 hadir dalam berbagai varian ukuran untuk mengakomodasi kebutuhan komputasi dan akurasi yang berbeda. MobileNetV4 *Small* (Conv-S), *Medium* (Conv-M), *Large* (Conv-L) dirancang untuk memberikan kinerja optimal pada skala yang berbeda. Berikut disajikan spesifikasi arsitektur dari ketiga varian model MobileNetV4 pada Tabel 1.

Tabel 1. Spesifikasi Arsitektur Varian Model MobileNetV4

Fitur	MobileNetV4 Conv-S	MobileNetV4 Conv-M	MobileNetV4 Conv-S
Ukuran Input Awal	$224^2 \times 3$	$256^2 \times 3$	$384^2 \times 3$
Output Dimensi Awal(Setelah Conv2D)	32	32	32
Blok Pertama (Setelah CConv2D)	FusedIB	FusedIB	FusedIB
Kernel DW K1/K2 Umum	$3 \times 3, 5 \times 5$	$3 \times 3, 5 \times 5$	$3 \times 3, 5 \times 5$
Variasi Blok	FusedIB, ExtraDW, IB, ConvNext	FusedIB, ExtraDW, FFN, ConvNext	FusedIB, ExtraDW, ConvNext
Expanded Dim Max	576 (pada ExtraDW)	1024 (pada ExtraDW, ConvNext, FFN)	2048 (pada ExtraDW, ConvNext)
Output Dim Max	128 (pada banyak blok)	256 (pada banyak blok)	512 (pada banyak blok)
Dimensi Sebelum Avgpool	$7^2 \times 960$	$8^2 \times 960$	$12^2 \times 960$
Ukuran Kernel Avgpool	7×7	8×8	12×12
Output dimensi akhir (setelah Avgpool)	960	960	960

2.3 Transfer Learning

Transfer learning merupakan salah satu teknik dalam *machine learning* yang melibatkan pemanfaatan pengetahuan yang dipelajari dari suatu tugas tertentu untuk meningkatkan kinerja pada tugas lain yang terkait. Pendekatan ini sangat sesuai digunakan dalam situasi dimana label dataset yang digunakan terbatas. Dalam kasus ini, model dapat memanfaatkan pengetahuan yang dipelajari dari tugas terkait yang memiliki jumlah data berlabel lebih besar. Hal ini memungkinkan model untuk mengatasi masalah *overfitting* dan meningkatkan kinerjanya pada tugas yang akan diberikan (Ali et. al., 2023). Pada penelitian ini lapisan awal (*feature extractor*) dibekukan (*freeze*) untuk mempertahankan kemampuan ekstraksi fitur generik. Selanjutnya lapisan akhir (*fully connected layer*) diganti dan disesuaikan dengan jumlah kelas yang ada pada dataset PlantVillage.

2.4 Penelitian Terkait

Penelitian terkait pada klasifikasi penyakit daun tanaman dengan berbasis citra digital sudah menunjukkan perkembangan yang signifikan, khususnya dalam penerapan metode *deep learning* seperti CNN. Metode ini dikenal dengan kemampuan tinggi dalam mengenali pola visual dan sangat efektif untuk klasifikasi citra pada daun tanaman yang terinfeksi penyakit, serta otomatis dapat mengekstraksi fitur pada gambar tanpa memerlukan proses perancangan fitur secara manual, sehingga sesuai untuk digunakan dalam klasifikasi penyakit daun tanaman berbasis citra. Seiring dengan meningkatnya kebutuhan terhadap aplikasi yang bersifat *real-time* dan berbasis perangkat dengan sumber daya terbatas, penelitian terkini mulai mengarah pada pengembangan arsitektur CNN yang ringan namun tetap mempertahankan akurasi tinggi.

Salah satu penelitian yang relevan dilakukan oleh I Gede Diva Dwijayana dan I Gede Arta Wibawa (2022) dengan judul “Implementasi *Transfer Learning* Dalam Klasifikasi Penyakit Pada Daun Teh Menggunakan MobileNetV2”. Dalam penelitian ini penulis mengimplementasikan arsitektur MobileNetV2 untuk klasifikasi penyakit daun tanaman pada Daun Teh. Data dibagi menjadi 70% untuk pelatihan dan 30% untuk validasi, disertai dengan proses normalisasi dan *resize* citra ke ukuran 224x224 pixel. Hasil pelatihan ini menunjukkan bahwa arsitektur ringan seperti MobileNetV2 dapat memberikan performa klasifikasi yang kompetitif dengan beban komputasi yang relatif rendah.

Kemudian, Studi lain yang memperkuat efektivitas CNN dilakukan oleh Deva Strahl Ephifano Soreninu et al. (2023) dengan judul “Penggunaan *Deep Learning* Dalam Deteksi Dini Penyakit Tanaman Menggunakan Citra Digital”. Dalam penelitian ini penulis mengembangkan model CNN guna mendeteksi dini penyakit tanaman berbasis citra digital. Dalam penelitian ini, didapat akurasi mencapai 92%, ini menunjukkan waktu pemrosesan gambar kurang dari satu detik, sehingga potensial untuk implementasi pada aplikasi *mobile*. Adapun tantangan utama yang dihadapi yaitu variasi kualitas dalam gambar, yang diatasi melalui *preprocessing* dan augmentasi data. Oleh karena itu, penelitian ini menegaskan pentingnya integrasi pada data tambahan seperti cuaca dan kondisi pada tanah untuk meningkatkan performa klasifikasi dalam konteks dunia nyata.

Selanjutnya, penelitian oleh Putra et al. (2023) dengan judul “*Transfer Learning* Untuk klasifikasi Penyakit Dan Hama Padi Menggunakan MobileNetV2”. Dalam Penelitian ini penulis mengimplementasikan metode *transfer learning* dengan arsitektur MobileNetV2 untuk klasifikasi penyakit dan hama pada tanaman padi. Model ini dipilih karena kemampuannya yang efisien dalam mengenali objek pada citra dengan jumlah parameter yang relatif kecil. Hasil penelitian yang didapatkan menghasilkan akurasi klasifikasi yang cukup tinggi pada data citra daun padi. Dataset yang digunakan dalam penelitian ini berisikan 5 penyakit, 3 hama dan 1 tanaman padi sehat. Kemudian, setiap data akan melewati tahap *preprocessing* dan augmentasi. Jumlah parameter dan performa yang dimiliki MobileNetV2 cukup baik dalam mengklasifikasi penyakit pada tanaman padi dengan jumlah 2,27 juta

parameter, akurasi 96%, *precision* sebesar 96%, *recall* sebesar 96%, *specificity* sebesar 99%, dan *f1-score* sebesar 96%.

Penelitian serupa dilakukan oleh Jin Zhu, Chuanhui Zhang, dan Changjiang Zhang (2023) dengan judul “*An Improved MobileNetV3 Mushroom Quality Classification Model Using Images With Complex Backgrounds. Agronomy*,”. Pada penelitian ini penulis mengusulkan sebuah model bernama P-MobileNet klasifikasi PSPR. Dalam hal akurasi klasifikasi yang tinggi, model P-MobileNet memperlihatkan keunggulannya seperti jumlah parameter yang lebih sedikit, dan lebih cepat dibandingkan dengan model aslinya. Model ini mampu mempertahankan akurasi klasifikasi sebesar 98,9%, hal ini membuat model tersebut menjadi efisien dan andal untuk diterapkan dalam perangkat dengan keterbatasan sumber daya.

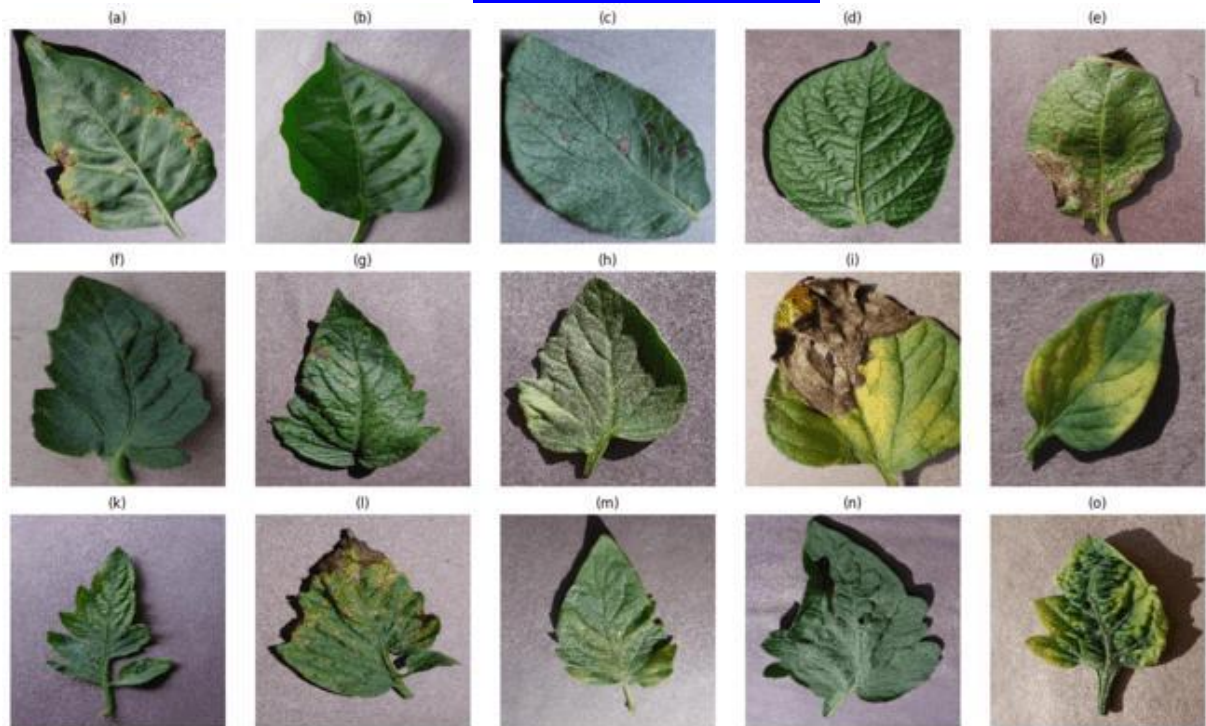
Implementasi CNN juga ditunjukkan dalam penelitian Dwianto et al. (2024) dengan judul “Penerapan *Convolutional Neural Network* (CNN) Dalam Deteksi Penyakit Pada Daun Tanaman Terong”. Pada penelitian ini penulis mengimplementasikan CNN untuk mendeteksi penyakit pada tanaman terong. Dari penelitian ini didapat bahwa CNN bisa membedakan mana daun tanaman terong yang sehat dan mana daun tanaman terong yang terinfeksi penyakit. Dengan *optimizer* Adam didapat hasil akurasi dan *loss* terbaik yaitu akurasi pelatihan dan validasi mendekati 1 setelah dilakukan 10 *epoch*. Kemudian, RMSProp menunjukkan akurasi baik mendekati 0,95. Sementara itu, pada *Stochastic Gradient Descent* (SGD) dapat meningkatkan akurasi lebih lambat. Oleh karena itu, CNN terbukti efektif dalam membedakan daun sakit dan sehat, serta dapat merekomendasikan eksplorasi lebih lanjut terhadap algoritma optimasi lainnya guna memperluas kumpulan data.

Penelitian-penelitian tersebut menjadi dasar penting dalam pengembangan sistem klasifikasi penyakit tanaman menggunakan arsitektur CNN dengan bobot ringan, untuk mencapai akurasi tinggi serta efisiensi komputasi yang lebih baik menggunakan MobileNetV4. Dengan menggunakan metode seperti segmentasi, ekstraksi fitur, dan klasifikasi berbasis *machine learning* membuat proses klasifikasi penyakit menjadi lebih cepat dan dapat diandalkan. CNN juga menjadi metode yang dominan digunakan dikarenakan strukturnya yang terdiri dari lapisan *convolution*, *pooling*, dan *fully connected*. Serta CNN mampu mendeteksi pola visual dengan baik dan menghasilkan *output* klasifikasi yang akurat (Jinan & Hayadi, 2022).

3. Metodologi Penelitian

3.1 Dataset

Dataset yang digunakan pada penelitian ini adalah PlantVillage, yang diperoleh dari situs kaggle. Dataset ini berisi sekitar 20.600 gambar daun tanaman yang terbagi ke dalam beberapa kelas berdasarkan jenis tanaman dan kondisi penyakit. Setiap gambar telah dikelompokkan ke dalam folder sesuai dengan nama kelas masing-masing. Adapun daftar lengkap dari nama kelas dapat ditemukan di repository Github berikut : <https://github.com/titoariffianto/Deteksi-Penyakit-Daun-MobileNetV4>.



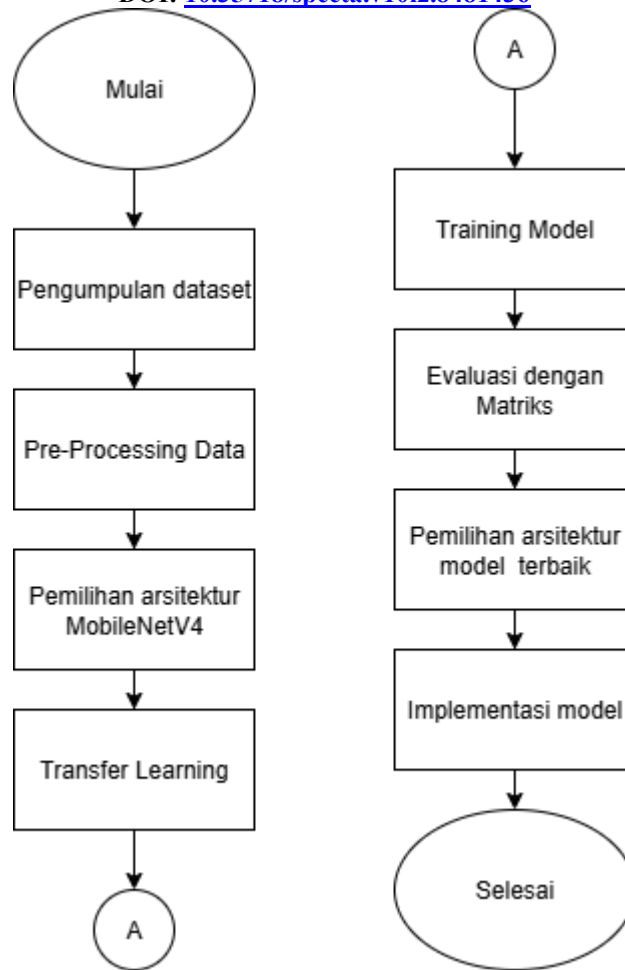
Gambar 3. Sampel Foto dari Dataset PlantVillage

Terdapat total 15 kelas berbeda pada dataset ini yang dapat dilihat setiap contoh nya pada gambar 2. Sampel foto mencakup: (a) *Pepper_bell_Bacterial_spot*, (b) *Pepper_bell_Healthy*, (c) *Potato_Early_Blight*, (d) *Potato_healthy*, (e) *Potato_Late_blight*, (f) *Tomato_Target_Spot*, (g) *Tomato_Mosaic_Virus*, (h) *Tomato_YellowLeaf_Curl_Virus*, (i) *Tomato_Bacterial_Spot*, (j) *Tomato_Early_blight*, (k) *Tomato_Healthy*, (l) *Tomato_Late_Blight*, (m) *Tomato_Leaf_Mold*, (n) *Tomato_Septoria_Leaf_spot*, (o) *Tomato_spider_mites_Two_spotted_spider_mite*

Dalam proses pelatihan dan validasi model, dataset dibagi menjadi dua bagian menggunakan metode *random_split* dari PyTorch dengan proporsi 80% untuk pelatihan (*training*) dan 20% untuk validasi (*validation*). Pembagian ini dilakukan secara acak guna menjaga distribusi kelas yang merata di setiap *subset*, setelah dataset dimuat menggunakan ImageFolder yang secara otomatis memberikan label berdasarkan nama folder.

3.2 Prosedur Penelitian

Prosedur penelitian yang dilakukan dalam penelitian ini disajikan pada gambar 3 sebagai berikut.



Gambar 4. Diagram Alir Penelitian

3.3 Desain Konfigurasi Eksperimentasi

Dalam penelitian yang dilakukan, eksperimen bertujuan untuk mengklasifikasikan penyakit daun tanaman secara otomatis berdasarkan citra daun tanaman menggunakan arsitektur model MobileNetV4 dengan pendekatan *transfer learning* untuk mempercepat proses pelatihan, serta meningkatkan kinerja model. Seperti yang telah dipaparkan pada bab 2, arsitektur MobileNetV4 mencakup tiga komponen utama, yaitu *UIB*, *MQA*, dan *NAS*. Ketiga elemen ini berfungsi untuk meningkatkan efisiensi komputasi, membuat model lebih fokus pada bagian penting di citra, serta membantu menghasilkan rancangan arsitektur yang optimal. Dalam studi ini, komponen tersebut digunakan sebagaimana bawaan asli MobileNetV4 tanpa dilakukan perubahan manual. Penelitian ini diharapkan dapat memberikan kontribusi nyata dalam pengembangan teknologi pertanian digital yang efisien dan juga akurat untuk klasifikasi penyakit daun tanaman. Kode implementasi dan eksperimen tersedia dalam repositori github berikut: <https://github.com/titoariffianto/Deteksi-Penyakit-Daun-MobileNetV4>.

Penelitian dilakukan di lingkungan *runtime* google colab. *Runtime* ini memiliki sistem operasi dasar yaitu Linux dengan GPU Nvidia T4 dengan kapasitas memori 16 GB VRAM dan CPU Intel Xeon E5-2673 v4 @ 2.30GHz serta memiliki RAM sebesar 13 GB. Framework utama yang digunakan adalah PyTorch (versi 2.x) dengan dukungan pustaka tambahan seperti TorchVision, scikit-learn, dan matplotlib. Seluruh eksperimen dijalankan pada *runtime* Google Colab berbasis *cloud* yang tersedia untuk digunakan secara gratis.

Model yang digunakan dalam pelatihan adalah varian MobileNetV4 *small*, MobileNetV4 *Medium*, dan MobileNetV4 *Large* yang akan disesuaikan dengan 15 kelas dari dataset PlantVillage. Untuk mencapai keseimbangan antara performa model dan efisiensi waktu pelatihan, beberapa parameter ditetapkan

yang meliputi jumlah *epoch* 50, ukuran *batch* 16, laju pembelajaran 0.0001, fungsi *loss* menggunakan *cross Entropy Loss*, dan *optimizer* Adam, yang dikenal mampu mempercepat konvergensi dan menghasilkan akurasi tinggi dalam pelatihan CNN berdasarkan studi empiris (Zohrevand & Imani, 2022). Selama proses pelatihan, model menjalani pelatihan terhadap data *training* dan evaluasi, dengan nilai *loss* dan *accuracy* dicatat untuk memantau performa dan stabilitas model. Proses pelatihan dilakukan dalam mode *train()*, sedangkan proses evaluasi dalam mode *eval()* tanpa pembaruan bobot.

Pra-pemrosesan data dilakukan pada dataset citra tanaman untuk menyesuaikan gambar dengan kebutuhan model MobileNetV4. Langkah-langkah yang dilakukan meliputi *random horizontal flip*, *random rotation* hingga 10 derajat, serta *color jitter* (Kecerahan dan kontras), normalisasi seluruh citra menggunakan nilai *mean* [0.485, 0.456, 0.406] dan *standard deviation* [0.229, 0.224, 0.225] yang berasal dari dataset ImageNet.

Sebelum mengimplementasikan model MobileNetV4, akan dibandingkan terlalu dahulu hasil kinerja dari tiga varian model MobileNetV4 yakni *Small*, *Medium* dan *Large* dalam melakukan klasifikasi daun tanaman. Analisis yang akan dilakukan mencakup evaluasi terhadap performa masing-masing model berdasarkan matriks evaluasi dari proses pelatihan serta evaluasi kinerja model akhir untuk mengidentifikasi arsitektur yang paling optimal dalam hal akurasi dan efisiensi komputasi, khususnya untuk perangkat yang memiliki sumber daya terbatas.

3.4 Matriks Evaluasi

Evaluasi dilakukan untuk mengukur performa model klasifikasi terhadap data validasi. Dalam penelitian ini, matriks evaluasi yang digunakan mencakup akurasi, presisi (*precision*), *recall*, *F1-score*, serta visualisasi menggunakan *confusion matrix*. Evaluasi ini penting untuk memastikan bahwa model tidak hanya menghafal data pelatihan (*overfitting*), tetapi juga mampu melakukan generalisasi terhadap data yang belum pernah dilihat sebelumnya.

- Akurasi (*accuracy*)

Akurasi adalah rasio jumlah prediksi yang benar terhadap seluruh data uji. Dirumuskan sebagai:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (3)$$

Dimana :

TP : *True Positive*

TN : *True Negative*

FP : *False Positive*

FN : *False Negative*

- *Precision*

Precision menunjukkan ketepatan model dalam memprediksi kelas tertentu dan dirumuskan sebagai:

$$Precision = \frac{TP}{TP+FP} \quad (4)$$

- *Recall*

Recall menunjukkan seberapa baik model dapat mengenali semua contoh kelas tertentu dan dirumuskan sebagai:

$$Recall = \frac{TP}{TP+FN} \quad (5)$$

- *F1-Score*

F1-score adalah rata-rata harmonik dari *precision* dan *recall*, yang berguna saat data tidak seimbang, dirumuskan sebagai:

$$F1 - Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (6)$$

- *Confusion Matrix*

Confusion matrix adalah tabel yang menggambarkan kinerja model klasifikasi dengan membandingkan hasil prediksi dengan label sebenarnya. Untuk kasus multi-class *confusion matrix* berbentuk matriks $K \times K$, dengan K adalah jumlah kelas.

$$\begin{bmatrix} TP & FP \\ FN & TN \end{bmatrix}$$

- Waktu Inferensi

Waktu inferensi adalah rata rata waktu yang dibutuhkan model untuk melakukan prediksi pada satu citra uji. Dirumuskan sebagai.

$$T_{inf} = \frac{\sum_{i=1}^N t_i}{N} \quad (8)$$

Dimana :

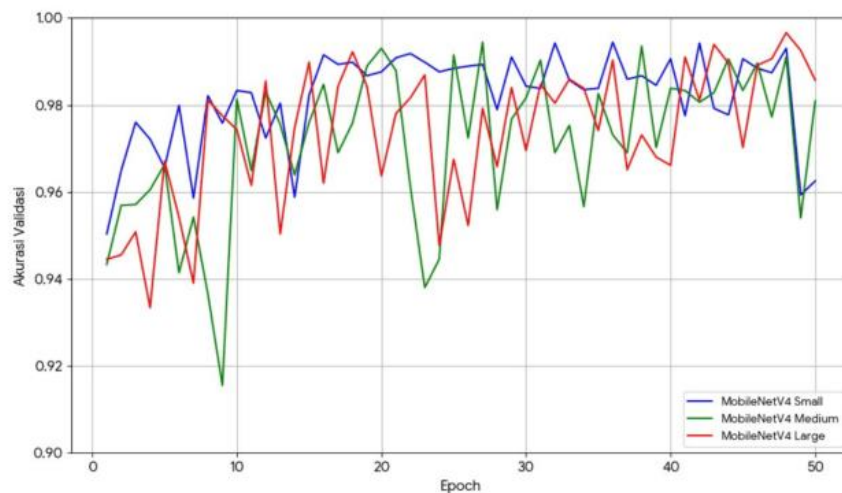
T_{inf} : Rata-rata waktu inferensi per citra (ms)

N : Jumlah citra uji

t_i : Waktu yang dibutuhkan untuk melakukan inferensi pada citra ke- i

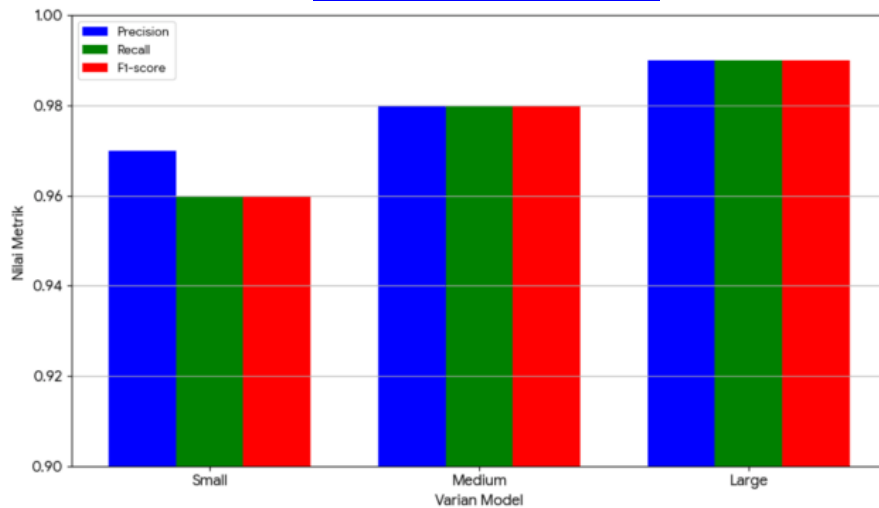
4. Hasil dan Pembahasan

Hasil dari penelitian ini menunjukkan bahwa arsitektur MobileNetV4 dengan pendekatan *transfer learning* berhasil mencapai performa yang sangat baik pada dataset PlantVillage dalam klasifikasi penyakit daun tanaman. Grafik perbandingan dalam akurasi validasi, matriks kinerja rata-rata, efisiensi komputasi, serta F1-score per kelas dari tiga variasi model (*Small*, *Medium*, dan *Large*) menunjukkan hasil yang konsisten, dimana semua model mampu memberikan akurasi tinggi, walaupun masih terdapat perbedaan signifikan dalam jumlah parameter dan waktu inferensi.



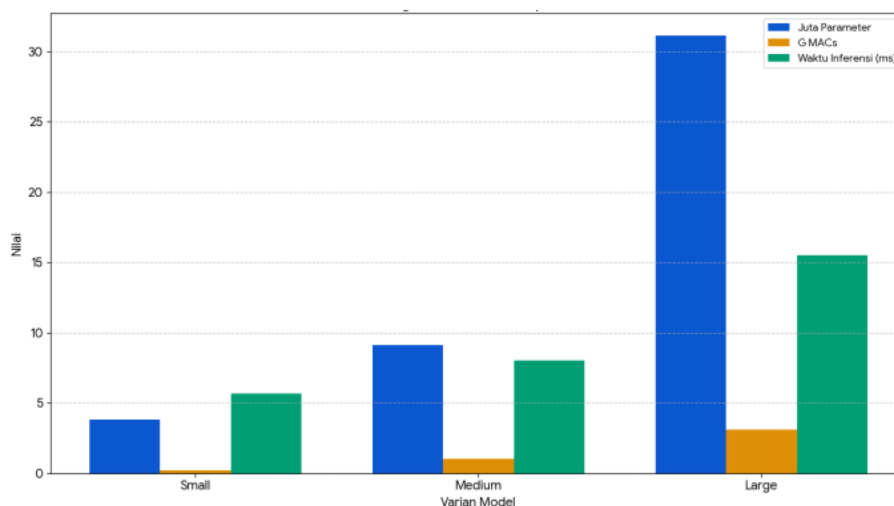
Gambar 5. Perbandingan Akurasi Validasi Model

Dari grafik yang ditunjukkan pada gambar 5, memperlihatkan bahwa MobileNetV4 *Small* mampu mencapai akurasi mendekati 99% dengan stabilitas pelatihan yang cukup baik. Akurasi tinggi juga didapatkan pada varian *Medium* dan *Large*, namun peningkatan performa relatif tidak signifikan dibandingkan dengan penambahan jumlah parameter yang besar. Hal ini menandakan adanya *trade-off* antara akurasi dan efisiensi model. Maka dari itu, meskipun parameter dari MobileNetV4 *Small* lebih sedikit, namun terbukti lebih seimbang karena performa akurasinya tidak kalah dibandingkan dengan varian yang lebih kompleks. Varian *Small* juga sangat sesuai untuk diimplementasikan pada perangkat dengan sumber daya terbatas seperti *smartphone*, karena dapat memperkecil penggunaan memori dan mempercepat waktu inferensi tanpa harus mengorbankan kualitas prediksi.



Gambar 6. Perbandingan Matriks Kinerja Rata-rata Model

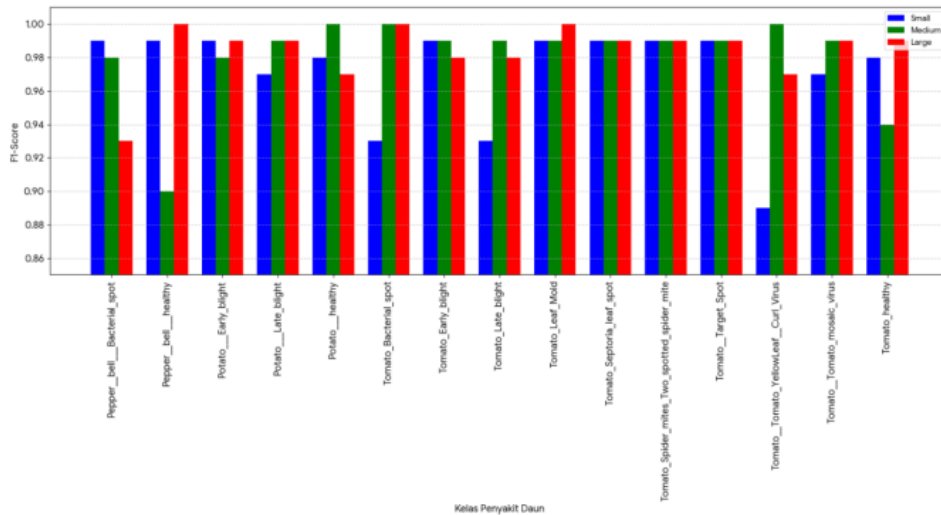
Grafik pada Gambar 6 menunjukkan bahwa ketiga varian model MobileNetV4 memiliki performa yang sangat baik dengan nilai rata-rata matriks di atas 0.96%. Secara keseluruhan, model yang lebih besar dan lebih kompleks, seperti MobileNetV4 *Large*, menunjukkan nilai matriks tertinggi yakni akurasi, presisi, recall, dan F1-score semuanya dinilai 0.99, diikuti oleh varian *Medium* (0.98), dan *Small* (0.96 - 0.97). Perbedaan ini meskipun memiliki nilai yang kecil, menegaskan bahwa peningkatan kapasitas model memang berkorelasi dengan peningkatan kinerja klasifikasi secara umum. Namun dalam hal ini juga memperkuat temuan kunci bahwa varian *Small* mampu mencapai kinerja hampir setara dengan model yang jauh lebih besar dan kompleks, menunjukkan bahwa model ini sangat efisien dalam memanfaatkan sumber daya minimal untuk menghasilkan hasil yang sangat andal.



Gambar 7. Perbandingan Efisiensi Komputasi Model

Dari grafik pada Gambar 7 menegaskan adanya *trade-off* yang jelas antara akurasi model dan kebutuhan sumber daya. MobileNetV4 *Large*, meskipun memiliki performa klasifikasi yang sangat kuat dengan akurasi puncak tertinggi (99.6%), menuntut biaya komputasi yang jauh lebih besar dibandingkan dengan varian lainnya. Hal ini terlihat dari jumlah parameternya yang mencapai 31 juta dan kompleksitas komputasi 3.1 G MACs, 15 kali lebih besar dari varian *Small*. Dampaknya, waktu inferensinya menjadi yang paling terlama, yaitu dengan rata-rata 15.49 ms per citra, dengan *throughput* terendah (64.56

citra/detik). Sebaliknya MobileNetV4 menunjukkan keseimbangan yang optimal antara performa dan efisiensi. Dengan jumlah parameter hanya 3.8 juta dan kompleksitas 0.2 G MACs, model ini mampu mencapai akurasi puncak yang hampir sama dengan *Medium* dan *Large*, namun dengan waktu inferensi yang jauh lebih cepat (5.67 ms/citra) dan *throughput* tertinggi (276.32 citra/detik). Efisiensi ini menjadikannya pilihan yang sangat praktis dan ideal untuk implementasi pada perangkat dengan sumber daya terbatas yang sangat menekankan kelancaran dan kecepatan.



Gambar 8. Perbandingan F1-Score per Kelas

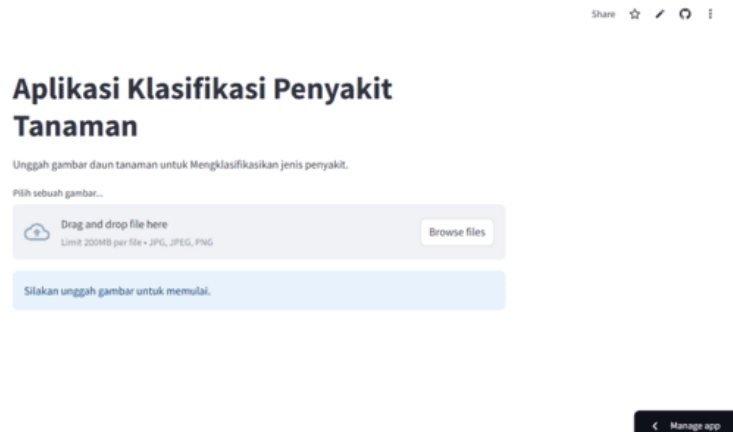
Analisis lebih lanjut ditunjukkan pada gambar 8, mengenai perbandingan F1-Score per Kelas yang menunjukkan bahwa sebagian besar mendekati nilai sempurna yang mencapai nilai F1. Varian model *Medium* dan *Large* secara konsisten memiliki performa yang sangat baik, dengan banyak kelas yang mencapai F1-score sempurna (1.00). Ini menunjukkan bahwa kapasitas dan kompleksitas model yang lebih besar membantu model ini untuk mengenali pola penyakit dengan presisi dan recall yang tinggi. MobileNetV4 *Medium* secara khusus unggul pada kelas Tomato_YellowLeaf_Curl_Virus dengan F1-Score 1.00, mengatasi tantangan yang dihadapi oleh varian *Small*. Meskipun secara keseluruhan menunjukkan hasil yang kuat, varian *Small* menunjukkan penurunan kinerja yang signifikan pada beberapa kelas. F1-score untuk Tomato_YellowLeaf_Curl_Virus hanya mencapai 0.89, dan F1-Score untuk Tomato_Bacterial_Spot dan Tomato_Late_Blight adalah 0.93. Data ini menunjukkan bahwa varian *Small* dengan arsitektur yang lebih ringan, mengalami kesulitan dalam membedakan pola-pola visual yang kompleks atau halus yang ditemukan pada kelas-kelas ini misalnya bintik atau bercak pada daun yang membuat model lebih kecil kesulitan untuk mengidentifikasi dengan akurat.

Secara keseluruhan, semua hasil eksperimen menunjukkan bahwa ketiga varian MobileNetV4 mampu memberikan akurasi tinggi di atas 96%. Namun, MobileNetV4 *Small* menjadi pilihan paling seimbang karena kinerjanya hampir setara dengan *Medium* dan *Large*, tetapi dengan jumlah parameter dan waktu inferensinya jauh lebih efisien. Dengan demikian, varian *Small* sangat sesuai untuk implementasi pada perangkat dengan sumber daya terbatas, sedangkan varian *Medium* dan *Large* lebih tepat digunakan pada sistem dengan kebutuhan akurasi maksimal.

4.1 Implementasi Model

Model klasifikasi penyakit daun tanaman yang telah dilatih sebelumnya diimplementasikan dalam bentuk antarmuka pengguna (UI) menggunakan framework Streamlit dari bahasa pemrograman Python. Untuk

implementasi ini, dipilih model yang telah dievaluasi dan mendapatkan hasil terbaik yaitu varian MobileNetV4 *Small*. Situs web ini memungkinkan pengguna untuk mengunggah citra daun tanaman dan menerima hasil klasifikasi penyakit. Selain itu, sistem juga menampilkan tingkat kepercayaan model terhadap prediksi tersebut dengan metode *softmax*, serta menyajikan deskripsi singkat mengenai penyakit tanaman yang terklasifikasi. Aplikasi ini dapat diakses melalui tautan <https://deteksidaun.streamlit.app/>. Berikut adalah tampilan dari antarmuka aplikasi klasifikasi penyakit daun tanaman yang ditampilkan pada Gambar 11.



Gambar 11. Tampilan Antarmuka Website Implementasi Model

5. Kesimpulan

Penelitian ini menunjukkan bahwa penerapan arsitektur MobileNetV4 dengan menggunakan pendekatan *transfer learning* mampu mengklasifikasikan penyakit daun tanaman secara otomatis pada dataset PlantVillage dengan akurasi diatas 96% dan rata-rata F1-score mencapai 0,99. Hasil perbandingan dari ketiga varian model memperlihatkan bahwa varian dengan kapasitas lebih besar memberikan peningkatan akurasi pada sejumlah kelas penyakit, namun membutuhkan jumlah parameter dan waktu inferensi yang lebih tinggi. Sementara itu, varian yang lebih ringan tetap menghasilkan kinerja klasifikasi yang kompetitif dengan keunggulan efisiensi komputasi serta kecepatan inferensi.

Kombinasi antara akurasi tinggi dan efisiensi ini menegaskan bahwa MobileNetV4 berpotensi diterapkan untuk sistem klasifikasi penyakit daun secara *real-time*, termasuk pada perangkat dengan sumber daya terbatas. Implementasi antarmuka berbasis web turut memperlihatkan bahwa hasil klasifikasi dapat diakses secara praktis oleh pengguna. Untuk pengembangan selanjutnya, sistem yang mampu berjalan secara offline serta integrasi dengan data kontekstual seperti cuaca dan kelembaban tanah, dan disertai evaluasi dari sisi pengguna, menjadi langkah penting agar teknologi ini lebih adaptif, relevan, dan berkelanjutan di lapangan.

Referensi

- Jinan, A., & Hayadi, B. H. (2022). Klasifikasi Penyakit Tanaman Padi Menggunakan Metode Convolutional Neural Network Melalui Citra Daun (Multilayer Perceptron). *Journal of Computer and Engineering Science*, 37-44.
- Mohanty, S. P., Hughes, D. P., & Salathé, M. (2016). *Using deep learning for image-based plant disease detection*. arXiv preprint arXiv:1604.03169. <https://arxiv.org/abs/1604.03169>
- Putra, O. V., Mustaqim, M. Z., & Muriatmoko, D. (2023). Transfer learning untuk klasifikasi penyakit dan hama padi menggunakan MobileNetV2. *Techno.COM: Jurnal Teknologi Informasi dan Komunikasi*, 22(3), 562–575.

- Deva Strahl Epifano Soreninu, Muhammad Hafidz Rizky Febrian, Pawit Mar'atun. (2023). *Penggunaan Deep Learning dalam Deteksi Dini Penyakit Tanaman Menggunakan Citra Digital*. Universitas Udayana.
- I Gede Diva Dwijayana, I Gede Arta Wibawa. (2022). *Implementasi Transfer Learning Dalam Klasifikasi Penyakit Pada Daun Teh Menggunakan MobileNetV2*. JNATIA, Vol. 1, No. 1, November 2022.
- Zhu, F., Sun, Y., Zhang, Y., Zhang, W., & Qi, J. (2023). An Improved MobileNetV3 Mushroom Quality Classification Model Using Images with Complex Backgrounds. *Agronomy*, 13(12).
- Gunawan, T. S., Ashraf, A., Riza, B. S., Haryanto, E. V., Rosnelly, R., Kartiwi, M., & Janin, Z. (2020). Development of video-based emotion recognition using deep learning with Google Colab. *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, 18(5), 2463-2471.
- Qin, D., Leichner, C., Delakis, M., Fornoni, M., Luo, S., Yang, F., Wang, W., Banbury, C., Ye, C., Akin, B., Aggarwal, V., Zhu, T., Moro, D., & Howard, A. (2024). MobileNetV4 — Universal Models for the Mobile Ecosystem. *arXiv*.
- Hendri, H., Hoki, L., Augusman, V., & Aryanto, D. (2021). *Penerapan machine learning untuk mengategorikan sampah plastik rumah tangga*. Jurnal Teknik Informatika STMIK TIME, 10(1), Juni 2021.
- Qin, X., Hu, J., Zhang, C., Wang, Y., Wang, W., & Li, H. (2024). *MobileNetV4: A Scalable and Efficient Network for On-device Image Classification*. In Proceedings of the European Conference on Computer Vision (ECCV 2024).
- Dwianto, S., Mubarak, F. N., Satriatama, D., & Agustin, T. (2024, November 23). *Penerapan Convolutional Neural Network (CNN) dalam Deteksi Penyakit pada Tanaman Terong*. Seminar Nasional AMIKOM Surakarta (SEMNAS) 2024. e-ISSN: 3031-5581. Sukoharjo, Indonesia.
- Kahana, Z. A. C. (2025). Implementasi Deep Learning untuk Deteksi Penyakit Tanaman Padi Menggunakan Citra Digital. *Prosiding Sains dan Teknologi*, 4(1), 459-465.
- Falaschetti, L., Manoni, L., Di Leo, D., Pau, D., Tomaselli, V., & Turchetti, C. (2022). A CNN-based image detector for plant leaf diseases classification. *HardwareX*, 12, e00363.
- Lu, J., Tan, L., & Jiang, H. (2021). Review on Convolutional Neural Network (CNN) applied to plant leaf disease classification. *Agriculture*, 11(8), 707.
- Nayak, D., Nair, R. R., Babu, T., & Kishore, S. (2024). CNN-based plant leaf disease detection: A key solution for enhancing agricultural productivity. *2024 3rd International Conference for Advancement in Technology (ICONAT)*, 1–6.
- Zohrevand, A., & Imani, Z. (2022). *An empirical study of the performance of different optimizers in the deep neural networks*. 2022 International Conference on Machine Vision and Image Processing (MVIP), 1–5.
- Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., & Adam, H. (2017). MobileNets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*.
- Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., & Chen, L. C. (2018). MobileNetV2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 4510-4520).
- Howard, A., Sandler, M., Chu, G., Chen, L. C., Chen, B., Tan, M., Wang, W., Zhu, Y., Pang, R., Vasudevan, V., Le, Q. V., & Adam, H. (2019). Searching for MobileNetV3. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (pp. 1314-1324).

- Tan, M., & Le, Q. (2019). EfficientNet: Rethinking model scaling for convolutional neural networks. In *International conference on machine learning* (pp. 6105-6114). PMLR.
- Cai, H., Gan, C., Wang, T., Zhang, Z., & Han, S. (2020). Once for all: Train one network and specialize it for efficient deployment. *arXiv preprint arXiv:1908.09791*.
- Sinaga, N. N., & Sembiring, A. (2025). Klasifikasi tumbuhan obat berdasarkan citra daun menggunakan algoritma CNN. *INCODING: Journal of Informatic and Computer Science Engineering*, 5(1), 64–74. <https://doi.org/10.34007/incoding.v5i1.833>