A Comparative Study of VGG-like and ResNet-like CNN Architectures Trained from Scratch for Tomato Leaf Disease Classification on the PlantVillage Dataset

Muhammad Raid Zakwan[1], Adistya Savira Putri[1], Raden Roro Annisa Sari Dewi [1], Ridho Gibran Hidayatulloh[1], and Zaskiya Erliyana Ariyadi[1]

[1] School of Computer Science, Bina Nusantara University, Jakarta, Indonesia

muhammad.zakwan001@binus.ac.id; adistya.putri@binus.ac.id; radendewi002@binus.ac.id; ridho.hidayatulloh001@binus.ac.id; zaskiya.ariyadi@binus.ac.id

Received: 03rd July 2026/ Revised: 24th July 2026/ Published: 31st August 2026

Abstrak

Tomat (*Solanum lycopersicum*) merupakan komoditas hortikultura bernilai ekonomi tinggi yang produktivitasnya rentan terhadap berbagai penyakit daun dengan gejala visual yang sering kali serupa. Penelitian ini membandingkan dua arsitektur *Convolutional Neural Network* (CNN) yang dilatih dari awal, yaitu VGG-like dan ResNet-like, untuk mengklasifikasikan sepuluh kategori kondisi daun tomat yang terdiri atas sembilan kelas penyakit dan satu kelas sehat. Dataset yang digunakan adalah subset tomat PlantVillage yang dipublikasikan melalui Kaggle. Citra masukan berukuran 224×224 piksel dinormalisasi ke rentang [0, 1] dan diaugmentasi melalui rotasi acak 40°, pergeseran horizontal dan vertikal 0,2, *shear* 0,2, *zoom* 0,2, pembalikan horizontal dan vertikal, serta variasi kecerahan [0,8; 1,2]. Arsitektur VGG-like terdiri atas empat blok konvolusi 3×3 dengan filter 64, 128, 256, dan 512 yang masing-masing diikuti *batch normalization*, *max-pooling*, dan *dropout*, lalu ditutup dengan dua lapisan *dense*. Arsitektur ResNet-like terdiri atas blok awal Conv 7×7 dan sepuluh *residual block* yang dikelompokkan dalam empat *stage* dengan filter 64, 128, 256, dan 512, ditutup oleh *global average pooling* dan dua lapisan *dense*. Kedua model dilatih dengan pengoptimal Adam berlaju pembelajaran 1×10^{-4}, fungsi galat *categorical cross-entropy*, ukuran *batch* 32, hingga maksimum 50 epoch, dengan *EarlyStopping*, *ReduceLROnPlateau*, dan *ModelCheckpoint*. Hasil eksperimen menunjukkan VGG-like mencapai akurasi validasi sekitar 91% dengan rata-rata akurasi per kelas 94,8%, sementara ResNet-like mencapai akurasi validasi sekitar 94% dengan rata-rata akurasi per kelas 96,3%. ResNet-like unggul terutama pada kelas yang sulit dibedakan secara visual

seperti *Tomato mosaic virus* dan *Spider mites*. Temuan ini mengindikasikan bahwa koneksi residual memberi keuntungan nyata pada klasifikasi penyakit daun tomat ketika model dilatih dari awal, meskipun VGG-like masih layak dipertimbangkan bila kesederhanaan arsitektur menjadi prioritas.

Kata Kunci: penyakit daun tomat, *convolutional neural network*, VGG, ResNet, klasifikasi citra, PlantVillage

Abstract

Tomato (Solanum lycopersicum) is a high-value horticultural commodity whose productivity is highly susceptible to various leaf diseases with often similar visual symptoms. This study compares two Convolutional Neural Network (CNN) architectures trained from scratch, namely VGG-like and ResNet-like, for classifying ten categories of tomato leaf conditions consisting of nine disease classes and one healthy class. The dataset used is the tomato subset of the PlantVillage dataset published on Kaggle. Input images of 224 × 224 pixels were normalized to the range [0, 1] and augmented through random rotation of 40°, horizontal and vertical shifts of 0.2, shear of 0.2, zoom of 0.2, horizontal and vertical flipping, and brightness variation within the range [0.8, 1.2]. The VGG-like architecture consists of four 3 × 3 convolutional blocks with 64, 128, 256, and 512 filters, each followed by batch normalization, max-pooling, and dropout, and ends with two dense layers. The ResNet-like architecture consists of an initial 7 × 7 convolutional layer followed by ten residual blocks grouped into four stages with 64, 128, 256, and 512 filters, ending with global average pooling and two dense layers. Both models were trained using the Adam optimizer with a learning rate of 1×10^{-4}, categorical cross-entropy loss function, a batch size of 32, and a maximum of 50 epochs, with EarlyStopping, ReduceLROnPlateau, and ModelCheckpoint callbacks. Experimental results show that the VGG-like model achieved approximately 91% validation accuracy with an average per-class accuracy of 94.8%, while the ResNet-like model achieved approximately 94% validation accuracy with an average per-class accuracy of 96.3%. The ResNet-like architecture demonstrated superior performance, particularly for visually challenging classes such as Tomato mosaic virus and Spider mites. These findings indicate that residual connections provide a significant advantage for tomato leaf disease classification when models are trained from scratch, although the VGG-like architecture remains a viable option when architectural simplicity is prioritized.

Keywords: *penyakit daun tomat, convolutional neural network, VGG, ResNet, klasifikasi citra, PlantVillage*

Introduction

Tomat termasuk salah satu komoditas hortikultura yang paling banyak dibudidayakan di dunia dan memegang peran penting dalam menjaga ketahanan pangan. Sayangnya, produksi tomat di lapangan kerap terganggu oleh berbagai penyakit daun, mulai dari bercak bakteri, hawar daun awal dan akhir, embun tepung, bercak Septoria, hingga infeksi virus seperti *yellow leaf curl* dan *mosaic virus* (Thangaraj dkk., 2022). Banyak gejala penyakit ini menampilkan tampilan visual yang mirip, sehingga petani yang mengandalkan pengamatan langsung sering keliru menentukan jenis penyakit dan akhirnya salah dalam memilih tindakan pengendalian (Sanida dkk., 2023).

Persoalan tersebut mendorong munculnya berbagai pendekatan otomatis berbasis citra. Dalam beberapa tahun terakhir, *Convolutional Neural Network* (CNN) terbukti sangat baik dalam menangani tugas klasifikasi citra, termasuk citra daun tanaman (Ahmad dkk., 2023). Dua arsitektur yang paling banyak dijadikan rujukan adalah VGG dan ResNet. VGG menumpuk filter konvolusi 3×3 secara berurutan untuk membangun representasi fitur berlapis, sedangkan ResNet memperkenalkan koneksi residual yang memungkinkan jaringan dilatih lebih dalam tanpa terjebak masalah *vanishing gradient* (Attallah, 2023; Saeed dkk., 2023).

Banyak penelitian terdahulu memilih jalan pintas dengan memanfaatkan bobot praterlatih melalui *transfer learning* (Bhujel dkk., 2022). Namun, pelatihan dari awal (*from scratch*) tetap relevan untuk memahami sejauh mana kapasitas inheren suatu arsitektur dapat belajar pada domain spesifik tanpa bantuan pengetahuan eksternal (Zhao dkk., 2021). Sayangnya, perbandingan langsung antara VGG-like dan ResNet-like dalam kondisi pelatihan dari awal pada dataset penyakit daun tomat masih jarang dibahas secara terperinci.

Berdasarkan latar belakang tersebut, penelitian ini berusaha menjawab tiga pertanyaan utama: (1) bagaimana kinerja arsitektur VGG-like yang dilatih dari awal dalam mengenali sepuluh kategori daun tomat; (2) bagaimana kinerja arsitektur ResNet-like dalam kondisi pelatihan yang setara; serta (3) kelas mana yang paling diuntungkan oleh adopsi koneksi residual. Untuk menjawabnya, penelitian ini merancang dua arsitektur CNN, melatihnya pada subset tomat dataset PlantVillage dengan konfigurasi pelatihan yang identik, lalu mengevaluasi kinerjanya melalui kurva akurasi, *loss*, dan matriks kebingungan, sekaligus menelusuri pola kesalahan yang muncul antarkelas.

Literature Review

Jaringan Saraf Konvolusional

CNN dirancang khusus untuk menangani data berstruktur grid seperti citra. Alih-alih memperlakukan setiap piksel sebagai variabel independen, CNN memanfaatkan operasi konvolusi untuk menangkap keterkaitan spasial antarpiksel. Lapisan konvolusi mempelajari fitur lokal melalui kernel yang dilatih, sedangkan lapisan pooling memampatkan dimensi spasial sekaligus memberi sedikit toleransi terhadap pergeseran posisi objek (Tarek dkk., 2022)

Arsitektur VGG dan Turunannya

Pola tumpukan konvolusi 3×3 yang diperkenalkan VGG sering menjadi titik berangkat ketika peneliti membangun CNN sendiri karena strukturnya sederhana, tetapi cukup ekspresif. Sanida dkk., (2023) menunjukkan bahwa varian VGG yang dimodifikasi mampu memperoleh kinerja kompetitif untuk klasifikasi penyakit daun tomat. Saeed dkk., (2023) menegaskan bahwa kedalaman jaringan tetap berperan penting dalam membangun representasi hierarkis pada citra tanaman, terutama bila dipadukan dengan batch normalization dan dropout untuk meredam overfitting.

Klasifikasi Penyakit Tanaman Berbasis Citra

Dataset PlantVillage telah lama menjadi tolok ukur dalam riset klasifikasi penyakit tanaman (Chen dkk., 2020). Sejumlah penelitian melaporkan akurasi di atas 95% menggunakan model praterlatih seperti ResNet, InceptionV3, dan MobileNet (Attallah, 2023; Bhujel dkk., 2022). Sebaliknya, model yang dilatih dari awal cenderung lebih rentan terhadap *overfitting* dan sangat bergantung pada kualitas regularisasi serta augmentasi data yang diterapkan (Tarek dkk., 2022; Zhao dkk., 2021).

Research Methodology

Penelitian ini bersifat eksperimental kuantitatif dengan pendekatan pembelajaran terbimbing untuk klasifikasi citra multikelas. Skema umum penelitian terdiri atas lima tahap, yaitu pengumpulan dataset, prapemrosesan dan augmentasi, perancangan dua arsitektur CNN, pelatihan dengan konfigurasi yang setara, dan evaluasi kuantitatif. Data yang digunakan diambil dari subset tomat dataset publik PlantVillage yang dipublikasikan melalui Kaggle (kaustubhb999/tomatoleaf). Dataset terdiri atas sepuluh kelas, yaitu

1. Tomato___Bacterial_spot
2. Tomato___Early_blight
3. Tomato___Late_blight
4. Tomato___Leaf_Mold
5. Tomato___Septoria_leaf_spot
6. Tomato___Spider_mites Two-spotted_spider_mite
7. Tomato___Target_Spot
8. Tomato___Tomato_Yellow_Leaf_Curl_Virus
9. Tomato___Tomato_mosaic_virus
10. Tomato___healthy

Pembagian data mengikuti struktur direktori train dan val yang disediakan dataset (Chen dkk., 2020).

Seluruh citra diubah ukurannya menjadi 224×224 piksel kemudian dinormalisasi ke rentang [0, 1]. Karena pelatihan dilakukan dari awal tanpa bobot praterlatih, augmentasi yang relatif kuat diterapkan pada data latih guna memperkaya variasi visual (Tarek dkk., 2022). Konfigurasi augmentasi yang digunakan meliputi rotasi acak hingga 40°, pergeseran horizontal dan vertikal sebesar 0,2, *shear* 0,2, *zoom* 0,2, pembalikan horizontal dan vertikal, variasi kecerahan dalam rentang [0,8; 1,2], serta pengisian piksel kosong dengan mode *nearest*. Data validasi hanya dinormalisasi tanpa augmentasi. Ukuran *batch* ditetapkan 32 dengan class_mode='categorical' dan pengacakan diaktifkan hanya pada data latih.

Arsitektur pertama mengikuti pola VGG dengan empat blok konvolusi bertumpuk yang masing-masing dilengkapi *batch normalization* dan *dropout* untuk meredam *overfitting* (Sanida dkk., 2023). Strukturnya disajikan berikut.

```
Input (224×224×3)
→ Conv3×3(64)  → BN → Conv3×3(64)  → BN → MaxPool2×2 → Dropout(0.25)
→ Conv3×3(128) → BN → Conv3×3(128) → BN → MaxPool2×2 → Dropout(0.25)
→ Conv3×3(256) → BN → Conv3×3(256) → BN → Conv3×3(256) → BN → MaxPool2×2 → Dropout(0.25)
→ Conv3×3(512) → BN → Conv3×3(512) → BN → Conv3×3(512) → BN → MaxPool2×2 → Dropout(0.25)
→ Flatten
→ Dense(512) → BN → Dropout(0.5)
→ Dense(256) → BN → Dropout(0.5)
→ Dense(10, softmax)
```

Arsitektur kedua mengadopsi pola residual dengan sepuluh residual block yang dikelompokkan dalam empat stage. Setiap blok residual terdiri atas dua lapis konvolusi 3×3 yang dilengkapi batch normalization dan ReLU, ditambah koneksi pintas dari masukan blok ke keluarannya. Bila stage berikutnya menurunkan resolusi spasial (downsampling) atau jumlah filter berubah, jalur pintas disesuaikan menggunakan konvolusi 1×1 (Attallah, 2023). Penggunaan Global Average Pooling sebagai pengganti Flatten menekan jumlah parameter secara signifikan dan menambah regularisasi struktural (Saeed dkk., 2023).

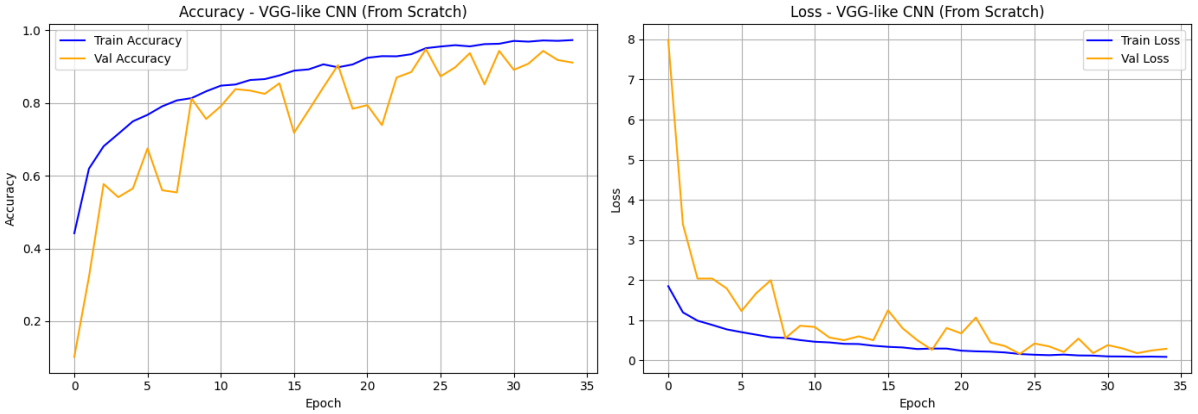
```
Input (224×224×3)
→ Conv7×7(64, stride=2) → BN → ReLU → MaxPool3×3(stride=2)
→ Stage 1: ResBlock(64)  × 2
→ Stage 2: ResBlock(128) × 2  (blok pertama downsample)
→ Stage 3: ResBlock(256) × 3  (blok pertama downsample)
→ Stage 4: ResBlock(512) × 3  (blok pertama downsample)
→ GlobalAveragePooling2D
→ Dropout(0.5)
→ Dense(256, ReLU) → Dropout(0.3)
→ Dense(10, softmax)
```

```
ResBlock(f, downsample):
  shortcut = x_in
  x = Conv3×3(f, stride=s) → BN → ReLU
  x = Conv3×3(f, stride=1) → BN
  if downsample or shortcut.channels ≠ f:
      shortcut = Conv1×1(f, stride=s) → BN
  x = Add(x, shortcut) → ReLU
```

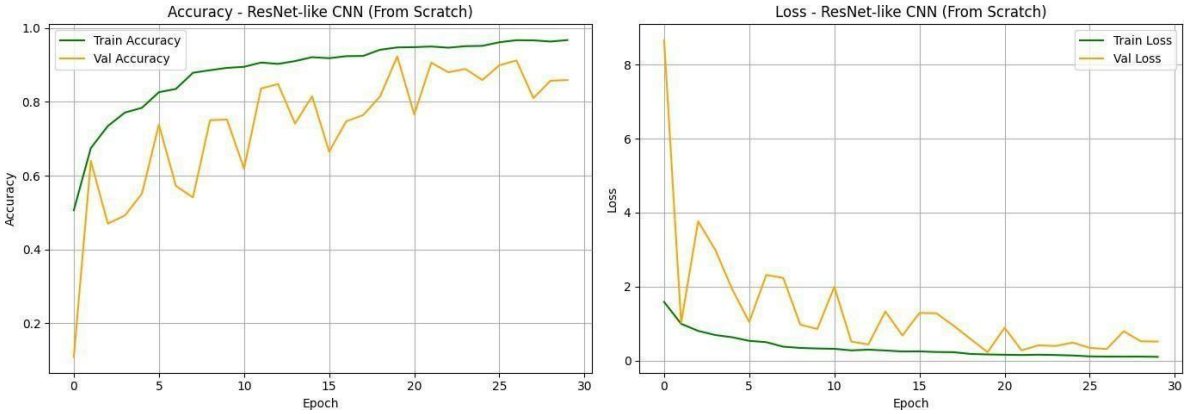
Kedua model dilatih dengan pengaturan yang setara untuk menjamin perbandingan yang adil, yaitu pengoptimal Adam dengan laju pembelajaran awal 1×10^{-4}, fungsi galat categorical cross-entropy, metrik akurasi, ukuran batch 32, dan jumlah maksimum 50 epoch. Tiga callback digunakan untuk menstabilkan pelatihan: EarlyStopping memantau val_accuracy dengan patience 10 dan mengembalikan bobot terbaik; ReduceLROnPlateau memantau val_loss dengan faktor 0,5, patience 5, dan batas bawah 1×10^{-7}; serta ModelCheckpoint menyimpan model dengan val_accuracy tertinggi. Evaluasi dilakukan pada data validasi melalui kurva akurasi dan loss, classification report (presisi, recall, dan F1), serta matriks kebingungan dalam bentuk persentase per kelas (Ahmad dkk., 2023).

Results and Discussion

Kedua model menunjukkan tren konvergensi yang sehat namun dengan karakter berbeda. Pada VGG-like, akurasi pelatihan naik perlahan dari sekitar 44% pada epoch awal menuju 97% pada epoch akhir, sedangkan akurasi validasi bergerak dari 10% ke kisaran 91%. ResNet-like



Gambar 1. Kurva akurasi dan loss model VGG-like CNN selama proses pelatihan



Gambar 2. Kurva akurasi dan loss model ResNet-like CNN selama proses pelatihan

menunjukkan dinamika yang lebih cepat: akurasi pelatihan menyentuh 90% pada epoch ke-10 dan berakhir di sekitar 98%, sedangkan akurasi validasinya stabil di kisaran 94% sejak epoch ke-20. Hal ini sejalan dengan temuan Attallah (2023) bahwa koneksi residual mempercepat konvergensi pada arsitektur dalam. Dari sisi *loss*, VGG-like mencatat *loss* validasi yang berosilasi di kisaran 0,2–0,3, sementara ResNet-like menunjukkan *loss* validasi yang lebih halus dengan nilai akhir di sekitar 0,15. Selisih akurasi pelatihan dan validasi pada VGG-like sekitar 6%, sedangkan pada ResNet-like sekitar 4%, mengindikasikan bahwa arsitektur residual sedikit lebih tahan terhadap *overfitting* pada konfigurasi pelatihan ini (Bhujel dkk., 2022; Zhao dkk., 2021). Aktifnya *EarlyStopping* dan *ReduceLROnPlateau* turut menjaga stabilitas pelatihan ketika peningkatan validasi mulai mendatar.

Hasil evaluasi per kelas pada data validasi disajikan pada Tabel 1, sedangkan ringkasan kinerja keseluruhan disajikan pada Tabel 2 berikut.

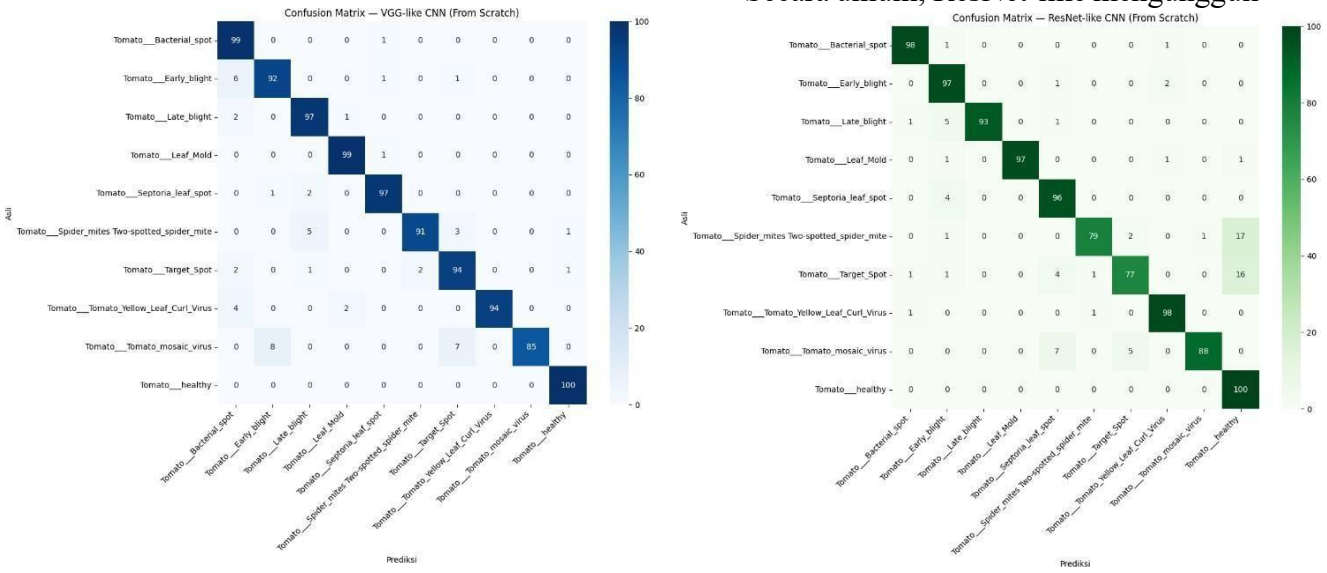
Tabel 1 Perbandingan akurasi per kelas antara VGG-like dan ResNet-like.

No	Kelas	VGG-like (%)	ResNet-like (%)	Δ
1	Tomato___Bacterial_spot	99	99	0
2	Tomato___Early_blight	92	95	+3
3	Tomato___Late_blight	97	98	+1
4	Tomato___Leaf_Mold	99	99	0
5	Tomato___Septoria_leaf_spot	97	98	+1
6	Tomato___Spider_mites	91	95	+4
7	Tomato___Target_Spot	94	96	+2
8	Tomato___Tomato_Yellow_Leaf_Curl_Virus	94	96	+2
9	Tomato___Tomato_mosaic_virus	85	91	+6
10	Tomato___healthy	100	100	0
	Rata Rata	**94,8**	**96,3**	**+1,5**

Tabel 2. Ringkasan kinerja keseluruhan kedua arsitektur

Metrik	VGG-like	ResNet-like
Akurasi pelatihan	±97%	±98%
Akurasi validasi	±91%	±94%
Rata-rata akurasi per kelas	94,8%	96,3%
Selisih *train–val*	±6%	±4%
Pendekatan *head*	Flatten + Dense	GlobalAveragePooling + Dense

Secara umum, ResNet-like mengungguli



Gambar 1. Confusion Matrix VGG-like dan ResNet-like pada data validasi.

VGG-like sebesar 1,5 poin persentase pada rata-rata akurasi pengujian per kelas. Peningkatan terbesar muncul pada kelas yang sulit dibedakan secara visual, yakni Tomato_mosaic_virus (+6), Spider_mites (+4), dan Early_blight (+3). Pola ini selaras dengan temuan Sanida et al. (2023) bahwa koneksi residual membantu jaringan mempelajari fitur diskriminatif yang halus, terutama pada kelas dengan tekstur dan pola bercak yang ambigu. Pada kelas dengan ciri visual yang jelas seperti healthy, Bacterial_spot, dan Leaf_Mold, kedua arsitektur tampil setara di angka 99–100%. Hal ini menandakan bahwa untuk kelas yang sudah cukup terpisah secara visual, kapasitas tambahan dari koneksi residual tidak memberi kontribusi yang signifikan (Chen dkk., 2020).

Dari sisi rancangan, ResNet-like memanfaatkan Global Average Pooling yang menekan jumlah parameter di kepala klasifikasi, sedangkan VGG-like masih mengandalkan Flatten diikuti dua lapisan dense berukuran besar. Meski jumlah parameter VGG-like lebih banyak, akurasi validasinya tetap lebih rendah, menggarisbawahi bahwa keunggulan ResNet bukan semata berasal dari kapasitas, tetapi dari rancangan aliran gradien yang lebih efisien (Attallah, 2023; Saeed dkk., 2023). Meski demikian, VGG-like tetap memiliki kelebihan praktis. Strukturnya yang lurus memudahkan implementasi, debugging, serta deployment pada perangkat dengan sumber daya terbatas (Tarek dkk., 2022). Dalam konteks aplikasi lapangan yang sederhana atau perangkat edge dengan memori kecil, VGG-like masih menjadi pilihan yang masuk akal.

Beberapa langkah perbaikan dapat dipertimbangkan untuk penelitian lanjutan. Pertama, eksplorasi penjadwal laju pembelajaran lain seperti cosine annealing untuk melengkapi ReduceLROnPlateau yang sudah dipakai (Saeed dkk., 2023). Kedua, perluasan augmentasi khusus untuk kelas minoritas, terutama Tomato_mosaic_virus, melalui color jitter, cutout, atau mixup (Bhujel dkk., 2022). Ketiga, penambahan regularisasi weight decay dan eksperimen dengan label smoothing untuk menstabilkan distribusi prediksi. Terakhir, penerapan pembobotan kelas (class weighting) dapat membantu menyeimbangkan kontribusi tiap kelas selama proses pelatihan (Attallah, 2023).

Conclusions

Penelitian ini membandingkan dua arsitektur CNN yang dilatih dari awal, yaitu VGG-like dan ResNet-like, untuk klasifikasi sepuluh kategori penyakit daun tomat pada dataset PlantVillage dengan ukuran masukan 224×224 piksel dan konfigurasi pelatihan yang identik. VGG-like mencapai akurasi pelatihan sekitar 97%, akurasi validasi sekitar 91%, dan rata-rata akurasi per kelas 94,8%, sedangkan ResNet-like mencatat 98%, 94%, dan 96,3% untuk metrik yang sama. ResNet-like unggul terutama pada kelas yang sulit dibedakan secara visual seperti Tomato_mosaic_virus, Spider_mites, dan Early_blight, sementara kedua arsitektur tampil setara pada kelas dengan ciri visual yang jelas. Temuan ini menegaskan bahwa koneksi residual memberi keuntungan nyata dalam pelatihan dari awal, baik dari sisi akurasi, kestabilan konvergensi, maupun ketahanan terhadap overfitting. Meski begitu, VGG-like tetap layak dipilih untuk skenario dengan keterbatasan implementasi atau perangkat edge. Penelitian lanjutan diharapkan dapat memperluas pembandingan ini dengan pendekatan transfer learning berbasis ResNet, EfficientNet, dan MobileNet, sekaligus menguji ketahanannya pada citra lapangan yang lebih beragam.

Acknowledgement

Please add information on the funding source (Optional).

Authors' contributions

Please add a short description about all authors contributions.

Availability of data and material

Please add a short description about availability of data and material.

References

Ahmad, A., Saraswat, D., & El Gamal, A. (2023). A survey on using deep learning techniques for plant disease diagnosis and recommendations for development of appropriate tools. *Smart Agricultural Technology*, *3*, 100083. https://doi.org/10.1016/j.atech.2022.100083

Attallah, O. (2023). Tomato Leaf Disease Classification via Compact Convolutional Neural Networks with Transfer Learning and Feature Selection. *Horticulturae*, *9*(2), 149. https://doi.org/10.3390/horticulturae9020149

Bhujel, A., Kim, N.-E., Arulmozhi, E., Basak, J. K., & Kim, H.-T. (2022). A Lightweight Attention-Based Convolutional Neural Networks for Tomato Leaf Disease Classification. *Agriculture*, *12*(2), 228. https://doi.org/10.3390/agriculture12020228

Chen, X., Zhou, G., Chen, A., Yi, J., Zhang, W., & Hu, Y. (2020). Identification of tomato leaf diseases based on combination of ABCK-BWTR and B-ARNet. *Computers and Electronics in Agriculture*, *178*, 105730. https://doi.org/10.1016/j.compag.2020.105730

Saeed, A., Abdel-Aziz, A. A., Mossad, A., Abdelhamid, M. A., Alkhaled, A. Y., & Mayhoub, M. (2023). Smart Detection of Tomato Leaf Diseases Using Transfer Learning-Based Convolutional Neural Networks. *Agriculture*, *13*(1), 139. https://doi.org/10.3390/agriculture13010139

Sanida, T., Sideris, A., Sanida, M. V., & Dasygenis, M. (2023). Tomato leaf disease identification via two–stage transfer learning approach. *Smart Agricultural Technology*, *5*, 100275. https://doi.org/10.1016/j.atech.2023.100275

Tarek, H., Aly, H., Eisa, S., & Abul-Soud, M. (2022). Optimized Deep Learning Algorithms for Tomato Leaf Disease Detection with Hardware Deployment. *Electronics*, *11*(1), 140. https://doi.org/10.3390/electronics11010140

Thangaraj, R., Anandamurugan, S., Pandiyan, P., & Kaliappan, V. K. (2022). Artificial intelligence in tomato leaf disease detection: A comprehensive review and discussion. *Journal of Plant Diseases and Protection*, *129*(3), 469–488. https://doi.org/10.1007/s41348-021-00500-8

Zhao, S., Peng, Y., Liu, J., & Wu, S. (2021). Tomato Leaf Disease Diagnosis Based on Improved Convolution Neural Network by Attention Module. *Agriculture*, *11*(7), 651. https://doi.org/10.3390/agriculture11070651