

KLASIFIKASI PENYAKIT DAUN TEH MENGUNAKAN ARSITEKTUR MOBILENETV4

Muhammad Agus Dwifa Sophian, Nur Rachmat

Program Studi Informatika, Fakultas Ilmu Komputer dan Rekayasa, Universitas Multi Data Palembang
Jalan Rajawali No. 14, Palembang, Indonesia
muhammadagusdwifasophian_2226250085@mhs.mdp.ac.id

ABSTRAK

Produktivitas tanaman teh di Indonesia sering terganggu oleh serangan penyakit daun, namun identifikasi secara manual kurang efisien karena membutuhkan waktu lama dan sangat bergantung pada keahlian pengamat. Penelitian ini bertujuan mengembangkan model klasifikasi penyakit daun teh menggunakan arsitektur MobileNetV4 serta menganalisis pengaruh *optimizer* dan *batch size* terhadap performa model. Menggunakan dataset teaLeafBD sebanyak 5.278 citra dari tujuh kategori, pendekatan *transfer learning* dengan strategi *full fine-tuning* diterapkan pada varian MobileNetV4-Conv-Small, Medium, dan Large. Model dilatih menggunakan *optimizer* Adam dan SGD dengan *batch size* 16 dan 32, kemudian dievaluasi melalui metrik akurasi, *precision*, *recall*, dan *F1-score*. Hasil pengujian menunjukkan bahwa *optimizer* Adam secara konsisten mengungguli SGD. Akurasi tertinggi sebesar 90,53% dicapai oleh varian Small dan Large pada *batch size* 32. Varian Small direkomendasikan sebagai konfigurasi terbaik karena mampu menyeimbangkan akurasi tinggi dengan efisiensi komputasi.

Kata kunci : MobileNetV4; penyakit daun teh; klasifikasi citra; transfer learning; optimizer

1. PENDAHULUAN

Tanaman teh (*Camellia sinensis*) merupakan salah satu komoditas perkebunan penting di Indonesia yang memiliki nilai ekonomi tinggi serta berperan strategis dalam sektor ekspor dan penyerapan tenaga kerja. Komoditas ini tidak hanya menjadi bahan baku utama industri minuman, tetapi juga menjadi sumber pendapatan utama bagi banyak petani di sektor perkebunan. Berdasarkan data Badan Pusat Statistik pada tahun 2024, luas areal perkebunan teh di Indonesia mencapai sekitar 97.539 hektare dengan total produksi sebesar 118.895 ton yang tersebar di berbagai provinsi serta melibatkan lebih dari 170.000 petani [1].

Data tersebut menegaskan bahwa keberlanjutan produksi teh sangat krusial dalam mendukung perekonomian nasional serta kesejahteraan masyarakat di sektor pertanian.

Meskipun memiliki potensi ekonomi yang besar, proses budidaya tanaman teh sering menghadapi kendala penurunan produktivitas. Salah satu permasalahan utama di lapangan adalah serangan penyakit pada daun yang disebabkan oleh berbagai faktor biotik seperti jamur, bakteri, dan virus [2].

Serangan patogen ini merusak jaringan daun, mengganggu proses fotosintesis, dan secara signifikan menurunkan kualitas serta kuantitas hasil panen pucuk teh. Masalahnya, proses identifikasi penyakit saat ini umumnya masih dilakukan secara manual berdasarkan gejala visual seperti bercak, perubahan warna, dan kerusakan jaringan [3].

Beberapa jenis penyakit memiliki karakteristik visual yang hampir serupa sehingga memerlukan ketelitian tinggi. Selain itu, identifikasi manual membutuhkan waktu yang relatif lama, sangat

bergantung pada subjektivitas keahlian pengamat, dan kurang efisien untuk skala perkebunan yang luas.

Sebagai rencana penyelesaian masalah tersebut, pendekatan otomatis berbasis teknologi kecerdasan buatan, khususnya *deep learning* dengan metode *Convolutional Neural Network* (CNN), dapat diimplementasikan. CNN memiliki kemampuan unggul dalam mengekstraksi fitur citra secara otomatis serta mengenali pola visual yang kompleks sehingga sangat efektif untuk klasifikasi citra penyakit [4].

Beberapa penelitian terdahulu telah memanfaatkan CNN untuk kasus ini; Winnarto dkk. [5] menggunakan arsitektur VGG16 dengan teknik *transfer learning* dan memperoleh akurasi 94%, Azis dan Irawan [6] menggunakan EfficientNetB0 dengan akurasi 97%, sedangkan Dwijayana dan Wibawa [7] menerapkan MobileNetV2 dan mencapai akurasi 94,6%. Lebih lanjut, Li dkk. [8] mengembangkan model berbasis MobileNetV3 dengan mekanisme *coordinate attention* yang menghasilkan akurasi 95,88%, sementara Pratama dkk. [9] memanfaatkan DenseNet121 dengan capaian akurasi sebesar 92,57%.

Perkembangan mutakhir juga menunjukkan bahwa arsitektur generasi terbaru, MobileNetV4, memiliki performa yang sangat menjanjikan dalam klasifikasi penyakit tanaman. Anugrah dan Rachmat [10] menggunakan MobileNetV4 untuk mendeteksi penyakit daun jagung dan memperoleh akurasi sebesar 95,25%, sementara Chen dkk. [11] memanfaatkan varian MobileNetV4-*small* untuk identifikasi penyakit daun anggur dengan akurasi mencapai 99,62%. Meskipun memiliki potensi performa dan efisiensi komputasi yang tinggi, penerapan MobileNetV4 pada klasifikasi penyakit daun teh hingga saat ini masih belum banyak dikaji. Penelitian terdahulu umumnya

masih berfokus pada perbandingan antararsitektur lama, serta belum mengeksplorasi secara spesifik pengaruh konfigurasi *hyperparameter* internal pada arsitektur ini.

Dalam proses pelatihan model *deep learning*, pemilihan komponen *optimizer* serta ukuran *batch size* memiliki pengaruh yang sangat besar terhadap stabilitas konvergensi dan akurasi akhir model. Oleh karena itu, penelitian ini dirancang untuk menyelesaikan permasalahan identifikasi dengan menguji tiga varian arsitektur yaitu MobileNetV4-Conv-Small, Medium, dan Large, menggunakan dataset teaLeafBD sebanyak 5.278 citra yang terbagi dalam tujuh kategori. Penelitian ini bertujuan untuk mengembangkan model klasifikasi penyakit daun teh yang optimal serta menganalisis secara mendalam pengaruh penggunaan *optimizer* (Adam dan SGD) serta variasi *batch size* (16 dan 32) terhadap performa model. Melalui pendekatan ini, penelitian diharapkan dapat memberikan kontribusi teoretis pada arsitektur *deep learning* terbaru sekaligus kontribusi praktis dalam mempercepat proses identifikasi penyakit daun teh secara cepat, akurat, dan efisien.

2. TINJAUAN PUSTAKA

2.1. Penyakit Daun Teh

Penyakit pada tanaman teh merupakan salah satu faktor yang dapat menurunkan kualitas dan produktivitas tanaman. Berbagai jenis patogen seperti jamur, bakteri, dan virus dapat menyebabkan penyakit pada tanaman teh, namun sebagian besar penyakit disebabkan oleh jamur patogen yang menyerang daun, batang, maupun akar tanaman [2].

Beberapa penyakit daun yang sering menyerang tanaman teh antara lain *brown blight*, *gray blight*, dan serangan hama seperti *helopeltis*. Penyakit *gray blight* disebabkan oleh jamur dari genus *Pestalotiopsis* yang menimbulkan lesi berwarna abu-abu dengan tepi cokelat dan dapat menyebar dengan cepat pada kondisi kelembapan tinggi [12].

Serangan penyakit pada daun teh dapat menyebabkan penurunan hasil panen sehingga identifikasi secara cepat dan akurat sangat penting untuk mendukung proses pengendalian.

2.2. Deep Learning dan Convolutional Neural Network

Deep learning merupakan bagian dari *machine learning* yang memanfaatkan jaringan saraf tiruan berlapis untuk mengolah data kompleks dan tidak terstruktur menjadi informasi yang lebih bermakna melalui proses pembelajaran secara hierarkis [13].

Salah satu keunggulan utama *deep learning* adalah kemampuannya melakukan ekstraksi fitur secara otomatis tanpa memerlukan intervensi manusia secara signifikan.

Convolutional Neural Network (CNN) merupakan salah satu jenis jaringan saraf tiruan dalam *deep learning* yang banyak digunakan pada pengolahan citra [14].

CNN dirancang untuk memproses data berbentuk grid seperti citra dengan memanfaatkan struktur tiga dimensi, yaitu *width*, *height*, dan *depth*. Secara umum, arsitektur CNN terdiri atas *feature extraction layer* yang berfungsi mengekstraksi fitur dari masukan serta *fully connected layer* yang digunakan untuk klasifikasi akhir. CNN tersusun dari beberapa jenis lapisan, yaitu *convolutional layer*, *pooling layer*, *activation layer* seperti ReLU, *fully connected layer*, dan *loss layer* [15].

2.3. Transfer Learning

Transfer learning merupakan metode yang memanfaatkan model arsitektur CNN yang telah dilatih sebelumnya untuk digunakan dalam mengklasifikasikan dataset baru sehingga tidak perlu melakukan proses pelatihan dari awal [16].

Pendekatan ini memberikan keuntungan berupa waktu pelatihan yang lebih singkat dan efisien serta memungkinkan model mencapai tingkat akurasi yang baik meskipun hanya didukung oleh dataset dengan jumlah terbatas [17].

Pada penelitian ini, transfer learning diterapkan dengan menggunakan bobot pralatih dari dataset ImageNet, yang kemudian disesuaikan terhadap karakteristik dataset penyakit daun teh melalui proses *fine tuning*.

2.4. MobileNetV4

MobileNetV4 merupakan iterasi terbaru dari arsitektur CNN yang dikembangkan oleh Google pada tahun 2024. Arsitektur ini dirancang untuk mencapai efisiensi universal serta performa optimal pada berbagai perangkat, termasuk CPU, GPU, DSP, dan akselerator khusus, dengan tujuan utama mencapai keseimbangan antara akurasi dan latensi [18].

MobileNetV4 memperkenalkan dua komponen utama, yaitu *Universal Inverted Bottleneck* (UIB) dan *Mobile Multi-Query Attention* (MQA). UIB merupakan pengembangan dari *Inverted Bottleneck* yang mengintegrasikan berbagai pendekatan arsitektur modern seperti *Inverted Bottleneck*, *ConvNext*, *Feed Forward Network*, serta varian *Extra Depthwise* sehingga memungkinkan optimasi proses *spatial mixing* dan *channel mixing* sesuai kebutuhan komputasi perangkat target [18]. Mekanisme MQA dirancang untuk meningkatkan efisiensi komputasi dengan mengurangi kebutuhan memori dibandingkan *Multi-Head Attention*.

MobileNetV4 memiliki beberapa varian, salah satunya MobileNetV4-Conv yang dirancang untuk menghasilkan latensi rendah dan terdiri atas varian Small, Medium, dan Large. Varian ini didominasi oleh operasi konvolusi serta meminimalkan penggunaan mekanisme *attention* untuk meningkatkan efisiensi pada perangkat dengan keterbatasan sumber daya [18].

Pada penelitian ini, ketiga varian tersebut digunakan dan dibandingkan untuk memperoleh

konfigurasi terbaik dalam klasifikasi penyakit daun teh.

2.5. Optimizer

Optimizer merupakan metode yang digunakan untuk meminimalkan nilai *loss* hingga mencapai kondisi konvergen pada saat proses pelatihan [13].

Dua jenis optimizer yang sering digunakan dalam proses klasifikasi adalah *Stochastic Gradient Descent* (SGD) dan *Adaptive Moment Estimation* (Adam). SGD memperbarui bobot secara bertahap untuk mengecilkan nilai *loss* dan umumnya didukung oleh parameter momentum serta *weight decay* untuk menjaga stabilitas konvergensi. Sementara itu, Adam menggabungkan konsep momentum dan RMSProp serta memiliki mekanisme *adaptive learning rate* yang bertujuan mempercepat proses konvergensi dan mengurangi osilasi selama pembelajaran.

2.6. Penelitian Terkait

Beberapa penelitian terdahulu yang relevan menjadi acuan dalam penelitian ini. Perbandingan posisi penelitian ini terhadap penelitian terdahulu disajikan pada Tabel 1.

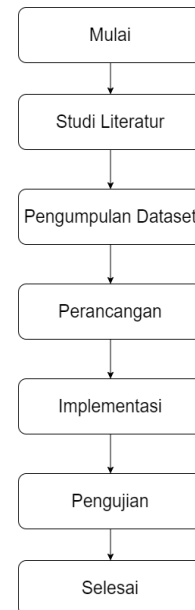
Tabel 1. Perbandingan penelitian terkait

Ref	Arsitektur	Mobile NetV4	Ada m	SGD	teaLeaf BD
[5]	VGG16		✓		
[6]	EfficientNetB0		✓		
[7]	MobileNetV2		✓		
[9]	DenseNet121		✓		
[10]	MobileNetV4	✓	✓	✓	
[11]	MobileNetV4-small	✓		✓	
Usulan	MobileNetV4	✓	✓	✓	✓

Berdasarkan Tabel 1 dapat dilihat bahwa penelitian klasifikasi penyakit daun teh sebelumnya umumnya belum menerapkan arsitektur MobileNetV4 dan belum membandingkan optimizer Adam dengan SGD pada dataset teaLeafBD. Posisi penelitian ini berada pada perpaduan penerapan tiga varian MobileNetV4, perbandingan dua optimizer, serta penggunaan dataset teaLeafBD yang belum banyak dikaji pada penelitian terdahulu.

3. METODE PENELITIAN

Penelitian ini dilaksanakan melalui beberapa tahapan, yaitu studi literatur, pengumpulan dataset, perancangan, implementasi, dan pengujian. Tahapan tersebut dirancang secara sistematis untuk meminimalkan kesalahan dalam proses penelitian. Alur tahapan penelitian ditunjukkan pada Gambar 1.



Gambar 1. Tahapan metodologi penelitian

Dari diagram pada Gambar 1 dapat dilihat bahwa penelitian dimulai dari studi literatur hingga tahap pengujian yang menghasilkan evaluasi performa model.

3.1. Studi Literatur

Tahapan ini dilakukan dengan mempelajari berbagai jurnal ilmiah dan buku yang relevan dengan topik klasifikasi penyakit daun teh menggunakan CNN, khususnya penerapan transfer learning pada arsitektur MobileNetV4. Hasil studi literatur digunakan sebagai acuan dalam menyusun metodologi penelitian, menentukan parameter pelatihan, serta sebagai dasar analisis terhadap hasil klasifikasi.

3.2. Pengumpulan Dataset

Dataset yang digunakan adalah teaLeafBD yang dikembangkan oleh Alam dkk. [3] dan dipublikasikan melalui jurnal *Data in Brief*. Dataset ini terdiri atas 5.278 citra daun teh dalam format JPG yang terbagi ke dalam enam kategori penyakit dan satu kategori daun sehat. Distribusi jumlah citra setiap kelas disajikan pada Tabel 2.

Tabel 2. Distribusi kelas pada dataset teaLeafBD

No	Kelas	Jumlah Citra	Citra
1	Tea algal leaf spot	418	
2	Brown blight	508	

No	Kelas	Jumlah Citra	Citra
3	Gray blight	1013	
4	Helopeltis	607	
5	Red spider	515	
6	Green mirid bug	1282	
7	Healthy leaf	935	
	Total	5278	

Berdasarkan Tabel 2 dapat dilihat bahwa distribusi citra antarkelas tidak sepenuhnya seimbang, dengan jumlah citra terbanyak pada kelas *green mirid bug* dan paling sedikit pada kelas *tea algal leaf spot*.

3.3. Prapemrosesan dan Augmentasi Data

Sebelum digunakan dalam pelatihan, seluruh citra melalui tahap prapemrosesan untuk menyesuaikan data dengan kebutuhan masukan model. Tahap ini meliputi penyeragaman ukuran citra menjadi 224×224 piksel, konversi citra ke bentuk *tensor*, serta normalisasi menggunakan nilai rata-rata dan simpangan baku standar ImageNet, yaitu *mean* (0,485; 0,456; 0,406) dan *std* (0,229; 0,224; 0,225).

Untuk meningkatkan keberagaman data pelatihan serta mengurangi risiko *overfitting*, diterapkan teknik augmentasi data secara acak pada data latih. Teknik augmentasi yang digunakan meliputi

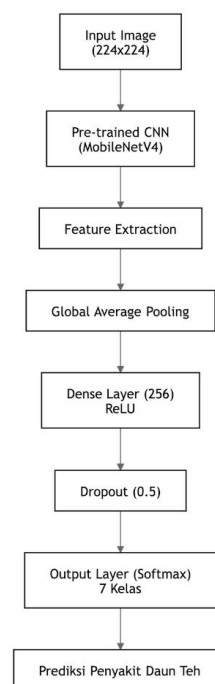
pembalikan citra secara horizontal (*random horizontal flip*) dengan probabilitas 0,5 transformasi berupa rotasi dalam rentang $\pm 20^\circ$ yang digabungkan dengan skala (*zooming*) pada rentang 0,8 hingga 1,2 serta penyesuaian *brightness*, *contrast* dan *saturation* sebesar 0,2 dan *hue* sebesar 0,1 melalui *color jitter*. Penggabungan rotasi dan penskalaan dalam satu transformasi afin dipilih karena lebih efisien secara komputasi dibandingkan penerapan transformasi geometri secara terpisah.

3.4. Pembagian Dataset

Dataset dibagi menjadi tiga bagian, yaitu 80% data latih, 10% data validasi, dan 10% data uji. Data latih dan data validasi digunakan dalam proses pelatihan model, sedangkan data uji digunakan untuk mengevaluasi kemampuan generalisasi model terhadap data yang tidak digunakan selama pelatihan.

3.5. Perancangan dan Pelatihan Model

Model dibangun menggunakan arsitektur MobileNetV4 dengan pendekatan *transfer learning*. Model pralatih MobileNetV4 yang telah dilatih pada dataset ImageNet digunakan sebagai dasar, kemudian dilakukan penyesuaian pada bagian klasifikasi sesuai dengan jumlah kelas pada dataset, yaitu tujuh kelas. Lapisan klasifikasi terdiri atas *global average pooling* serta lapisan keluaran dengan fungsi aktivasi *softmax*. Pada penelitian ini diterapkan strategi *full fine tuning*, yaitu seluruh lapisan model dibuka dan ikut diperbarui selama proses pelatihan sehingga model dapat menyesuaikan keseluruhan parameter terhadap karakteristik spesifik citra penyakit daun teh. Struktur arsitektur model ditunjukkan pada Gambar 2.



Gambar 2. Struktur arsitektur MobileNetV4 untuk klasifikasi penyakit daun teh

Penelitian ini menggunakan tiga varian arsitektur, yaitu MobileNetV4-Conv-Small, MobileNetV4-Conv-Medium, dan MobileNetV4-Conv-Large. Setiap varian dilatih menggunakan dua optimizer yang berbeda, yaitu Adam dan SGD, serta dua nilai *batch size*, yaitu 16 dan 32, untuk dibandingkan performanya. Proses perbandingan optimizer dilakukan dengan menggunakan konfigurasi optimal dari masing-masing optimizer, bukan dengan menyamakan nilai *learning rate*. Hal ini dikarenakan Adam memiliki mekanisme *adaptive learning rate* yang sensitif terhadap langkah besar sehingga membutuhkan *learning rate* kecil sebesar 0,0001. Sebaliknya, SGD memerlukan *learning rate* yang sedikit lebih besar sebesar 0,001 yang didukung oleh momentum sebesar 0,9 dan *weight decay* sebesar 0,0001 untuk menjaga stabilitas konvergensi tanpa terjebak dalam *local minimal* maupun mengalami *vanishing* atau *exploding gradient*. Konfigurasi *hyperparameter* yang digunakan disajikan pada Tabel 3.

Tabel 3. Konfigurasi hyperparameter pelatihan

Parameter	Adam	SGD
Epoch	50	50
Batch size	16 dan 32	16 dan 32
Learning rate	0,0001	0,001
Momentum	–	0,9
Weight decay	–	0,0001
Early stopping (patience)	12	12

Berdasarkan Tabel 3, kedua optimizer dilatih dengan jumlah *epoch* dan variasi *batch size* yang sama, sedangkan perbedaan utama terletak pada nilai *learning rate* serta parameter pendukung pada SGD. Mekanisme *early stopping* dengan nilai *patience* sebesar 12 diterapkan untuk menghentikan pelatihan apabila tidak terjadi peningkatan performa pada data validasi sehingga dapat mengurangi risiko *overfitting* dan menghemat waktu komputasi.

3.6. Pengujian dan Evaluasi

Model yang telah dilatih diuji menggunakan data uji yang tidak digunakan selama proses pelatihan. Hasil pengujian dievaluasi menggunakan confusion matrix untuk memperoleh nilai *accuracy*, *precision*, *recall*, dan *F1 score* yang dihitung menggunakan Persamaan (1), (2), (3), dan (4).

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \times 100\% \quad (1)$$

$$Precision = \frac{TP}{TP + FP} \times 100\% \quad (2)$$

$$Recall = \frac{TP}{TP + FN} \times 100\% \quad (3)$$

$$F1 - Score = 2X \frac{Precision \times Recall}{Precision + Recall} \times 100\% \quad (4)$$

Keterangan: TP (*True Positive*) merupakan jumlah data yang terklasifikasi dengan benar pada kelas tertentu, TN (*True Negative*) merupakan jumlah

data yang bukan kelas tertentu dan terklasifikasi dengan benar, FP (*False Positive*) merupakan jumlah data yang salah diklasifikasikan sebagai kelas tertentu, dan FN (*False Negative*) merupakan jumlah data yang salah diklasifikasikan sebagai bukan kelas tertentu. Nilai *accuracy* menunjukkan tingkat ketepatan model secara keseluruhan, *precision* menunjukkan ketepatan klasifikasi positif, *recall* menunjukkan kemampuan model dalam mendeteksi data yang sebenarnya positif, sedangkan *F1 score* merupakan rata-rata harmonik antara *precision* dan *recall*.

4. HASIL DAN PEMBAHASAN

4.1. Realisasi Pembagian Dataset

Dataset teaLeafBD yang berjumlah 5.278 citra dibagi dengan rasio 80% data latih, 10% data validasi, dan 10% data uji. Realisasi pembagian data tersebut menghasilkan 4.222 citra data latih, 528 citra data validasi, dan 528 citra data uji. Data uji sebanyak 528 citra inilah yang digunakan untuk mengukur kemampuan generalisasi seluruh konfigurasi model. Distribusi data uji mengikuti proporsi dataset awal, yaitu *tea algal leaf spot* 42 citra, *brown blight* 50 citra, *gray blight* 101 citra, *helopeltis* 61 citra, *red spider* 52 citra, *green mirid bug* 129 citra, dan *healthy leaf* 93 citra.

4.2. Skenario Pelatihan Model

Penelitian ini melatih tiga varian arsitektur MobileNetV4-Conv (Small, Medium, dan Large) dengan kombinasi dua optimizer (Adam dan SGD) dan dua nilai *batch size* (16 dan 32). Kombinasi tersebut menghasilkan 12 skenario pelatihan yang dirinci pada Tabel 4.

Tabel 4. Skenario pelatihan model

Skenario	Varian	Optimizer	Batch Size	Learning Rate	Epoch
1	Small	Adam	32	0,0001	50
2	Medium	Adam	32	0,0001	50
3	Large	Adam	32	0,0001	50
4	Small	SGD	32	0,001	50
5	Medium	SGD	32	0,001	50
6	Large	SGD	32	0,001	50
7	Small	Adam	16	0,0001	50
8	Medium	Adam	16	0,0001	50
9	Large	Adam	16	0,0001	50
10	Small	SGD	16	0,001	50
11	Medium	SGD	16	0,001	50
12	Large	SGD	16	0,001	50

Seluruh skenario menggunakan citra masukan yang telah diseragamkan menjadi 224×224 piksel, strategi *full fine tuning*, serta mekanisme *early stopping* dengan *patience* 12 untuk menghentikan pelatihan apabila performa validasi tidak lagi meningkat.

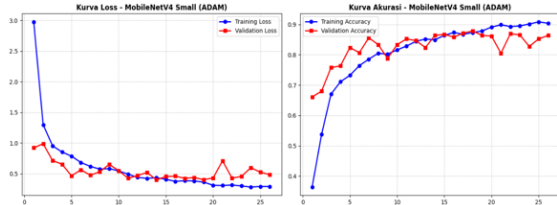
4.3. Hasil Pelatihan Model dengan Batch Size 16

Proses pelatihan pada ukuran *batch size* 16 memperlihatkan karakteristik kurva yang dinamis, di mana riak fluktuasi terlihat lebih kentara dibandingkan

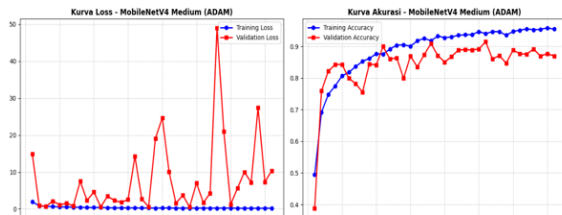
ukuran *batch size* yang lebih besar. Fenomena ini secara teoretis dipicu oleh *stochastic noise* yang tinggi, karena estimasi gradien dihitung berdasarkan sampel yang relatif sedikit pada setiap iterasi, sehingga pembaruan bobot model cenderung melompat-lompat.

4.4. Optimizer Adam (Batch Size 16)

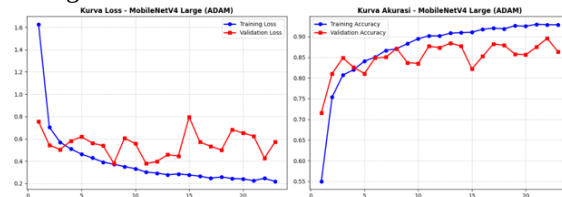
a. Small



b. Medium



c. Large



Gambar 3. Grafik kurva loss dan akurasi pelatihan pengoptimasi Adam batch size 16: (a) Small, (b) Medium, (c) Large]

Berdasarkan grafik pemantauan proses pelatihan (Gambar 3), ketiga varian arsitektur yang menggunakan pengoptimasi Adam berhasil memicu konvergensi dini sebelum menyentuh batas 50 *epoch* berkat sifat *adaptive learning rate* yang dimilikinya.

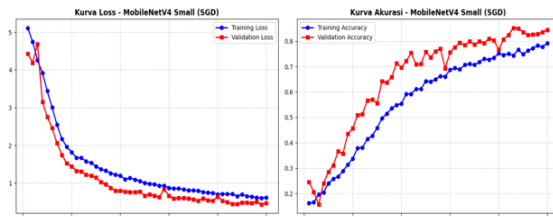
- Varian Small: Menunjukkan kurva *loss* yang menurun konsisten dan stabil, dengan penghentian otomatis (*early stopping*) terjadi pada *epoch* ke-26. Kurva akurasi data validasi bergerak beriringan rapat dengan akurasi data latih.
- Varian Medium: Kurva *training loss* langsung melandai mendekati angka nol pada *epoch* awal. Namun demikian, kurva *validation loss* mengalami fluktuasi ekstrem dengan lonjakan tajam (*spike*) yang menembus nilai ≈ 48 pada kisaran *epoch* ke-28 dan ke-34 sebelum akhirnya terhenti di *epoch* ke-37. Ketidakstabilan ini mengindikasikan adanya kendala *co-adaptation breakdown* sesaat, di mana perubahan bobot radikal pada seluruh lapisan model lewat metode *full fine-tuning* sempat merusak

representasi fitur yang telah dipelajari jaringan ketika diuji pada sampel data validasi.

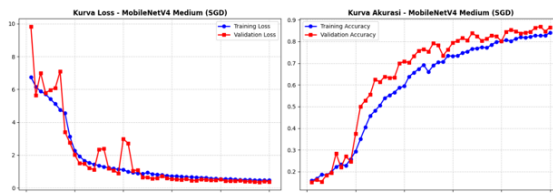
- Varian Large: Mengalami fluktuasi kurva *validation loss* yang melebar setelah *epoch* ke-12, dengan riak naik-turun berkisar antara 0,4 hingga 0,8 sebelum pelatihan diputus pada *epoch* ke-23. Fluktuasi ini menandakan ukuran *batch* 16 kurang mampu memberikan kestabilan arah gradien pada model dengan jumlah parameter yang sangat padat seperti varian Large.

4.5. Optimizer SGD (Batch Size 16)

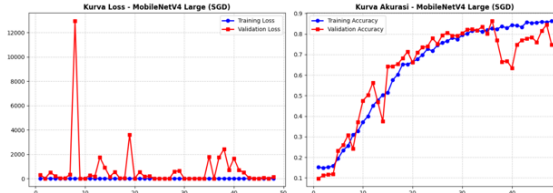
a. Small



b. Medium



c. Large



Gambar 4. Grafik loss dan akurasi pelatihan optimizer SGD batch size 16: (a) Small, (b) Medium, (c) Large]

Pada Gambar 4 menunjukkan kesesuaian dengan karakteristik dasar pengoptimasi *Stochastic Gradient Descent* (SGD), seluruh varian membutuhkan durasi belajar yang lebih panjang dan menghabiskan total 50 *epoch* penuh tanpa memicu *early stopping*.

- Varian Small: Memperlihatkan performa kurva ideal yang sangat halus (*smooth*) tanpa adanya lonjakan. Kurva *loss* menurun secara bertahap, sementara kurva akurasi merangkak naik secara linier dan konvergen mapan hingga akhir iterasi.
- Varian Medium: Menunjukkan kestabilan kurva yang baik, kendati terdapat riak riak fluktuasi minor pada *validation loss* di sekitar *epoch* ke-7 dan ke-21 yang sempat menyentuh nilai 3 hingga 4 sebelum akhirnya melandai stabil.
- Varian Large: Menampilkan fenomena berupa lonjakan kesalahan (*error spike*) yang luar biasa masif pada kurva *validation loss*, di mana nilainya

meledak melampaui angka 12.000 pada *epoch* ke-13. Kejadian ekstrem ini merupakan indikasi terjadinya *gradient explosion* sesaat, yang dipicu oleh akumulasi nilai gradien yang terlalu besar ketika melewati lapisan arsitektur *Large* yang sangat dalam saat seluruh layernya dibuka. Walau begitu, pengkondisian parameter *weight decay* (10^{-4}) berhasil meredam ledakan tersebut pada *epoch* berikutnya, sehingga kurva *loss*

langsung terjun bebas kembali melandai mendekati angka nol hingga *epoch* ke-50.

4.6. Hasil Evaluasi Data Uji Batch Size 16

Setelah proses pelatihan selesai, seluruh model diuji menggunakan 528 data uji independen yang belum pernah dilihat sebelumnya. Hasil ekstraksi nilai metrik performa klasifikasi laporan evaluasi disajikan secara komprehensif pada Tabel 5.

Tabel 5. Hasil evaluasi data uji batch size 16

Optimizer	Varian	Kategori	Precision	Recall	F1-Score	Accuracy
Adam	Small	Tea algal leaf spot	85%	79%	81%	87,31%
		Brown blight	70%	74%	72%	
		Gray blight	91%	81%	86%	
		Helopeltis	83%	89%	86%	
		Red spider	88%	81%	84%	
		Green mirid bug	87%	93%	90%	
		Healthy leaf	98%	100%	99%	
Adam	Medium	Tea algal leaf spot	89%	81%	85%	89,58%
		Brown blight	68%	82%	75%	
		Gray blight	93%	88%	90%	
		Helopeltis	89%	90%	89%	
		Red spider	91%	79%	85%	
		Green mirid bug	92%	94%	93%	
		Healthy leaf	96%	99%	97%	
Adam	Large	Tea algal leaf spot	84%	76%	80%	85,80%
		Brown blight	60%	82%	69%	
		Gray blight	91%	77%	83%	
		Helopeltis	88%	87%	88%	
		Red spider	80%	83%	81%	
		Green mirid bug	90%	88%	89%	
		Healthy leaf	96%	99%	97%	
SGD	Small	Tea algal leaf spot	64%	71%	67%	82,01%
		Brown blight	77%	54%	64%	
		Gray blight	88%	80%	84%	
		Helopeltis	77%	77%	77%	
		Red spider	71%	92%	80%	
		Green mirid bug	88%	83%	86%	
		Healthy leaf	89%	100%	94%	
SGD	Medium	Tea algal leaf spot	73%	79%	76%	86,74%
		Brown blight	81%	58%	67%	
		Gray blight	85%	94%	89%	
		Helopeltis	79%	90%	84%	
		Red spider	84%	88%	86%	
		Green mirid bug	94%	84%	89%	
		Healthy leaf	97%	99%	98%	
SGD	Large	Tea algal leaf spot	77%	86%	81%	87,50%
		Brown blight	84%	62%	71%	
		Gray blight	89%	90%	90%	
		Helopeltis	85%	82%	83%	
		Red spider	90%	83%	86%	
		Green mirid bug	89%	91%	90%	
		Healthy leaf	91%	100%	95%	

Berdasarkan Tabel 5, pada *batch size* 16 akurasi tertinggi diperoleh optimizer Adam pada varian Medium sebesar 89,58%, sedangkan pada SGD akurasi tertinggi diperoleh varian Large sebesar 87,50%. Sama seperti pada *batch size* 32, kelas *healthy leaf* secara konsisten memperoleh *F1-score* tertinggi, sementara *brown blight* tetap menjadi

kelas yang paling sulit dikenali dengan *recall* di kisaran 54–82%.

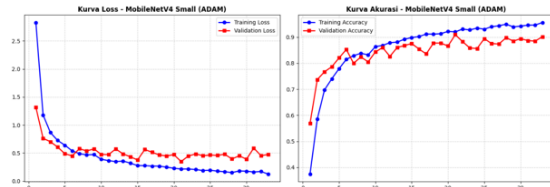
4.7. Hasil Pelatihan Model dengan Batch Size 32

Secara umum, transisi penggunaan ukuran *batch size* menuju angka 32 berhasil mereduksi tingkat kebisingan gradien (*gradient noise*) selama iterasi berlangsung. Langkah optimasi menjadi jauh lebih

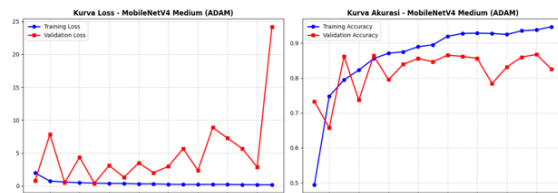
terarah sehingga kurva pergerakan model tampak lebih mulus, stabil, dan minim riak fluktuasi radikal bila dibandingkan dengan pengujian pada *batch size* 16.

4.8. Optimizer Adam (Batch Size 32)

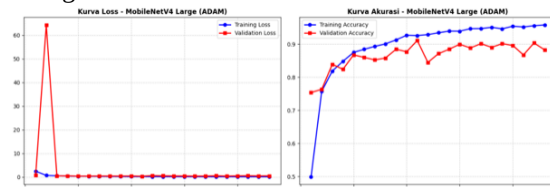
a. Small



b. Medium



c. Large



Gambar 5. Grafik loss dan akurasi pelatihan optimizer SGD batch size 32: (a) Small, (b) Medium, (c) Large]

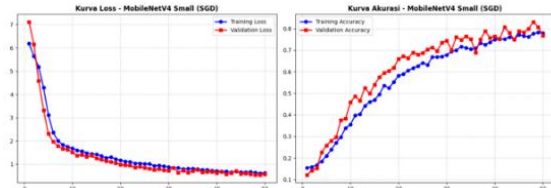
Sifat adaptif Adam pada *batch size* 32 kembali memicu pemberhentian dini (*early stopping*) pada seluruh tingkatan varian arsitektur.

- Varian Small: Menunjukkan profil kurva pelatihan paling ideal dalam penelitian ini. Penurunan kurva *loss* meluncur mulus tanpa fluktuasi berarti dan saling berhimpitan rapat sejak *epoch* awal. Kurva akurasi validasi pun menanjak kokoh hingga menyentuh kisaran angka 90% sebelum mekanisme *early stopping* mengakhiri pelatihan secara bersih pada *epoch* ke-33. Hal ini membuktikan model berhasil mengalami konvergensi optimal tanpa terjebak gejala *overfitting* maupun *underfitting*.
- Varian Medium: Pelatihan terhenti sangat dini, yaitu tepat pada *epoch* ke-17. Meskipun kurva akurasi latih melesat naik, kurva *validation loss* menunjukkan perilaku fluktuasi bising dengan lonjakan pertama ke nilai ≈ 8 di *epoch* ke-2 dan melonjak drastis kembali ke nilai ≈ 24 pada *epoch* ke-17. Gejala ketidakstabilan ini mengindikasikan model mengalami penyesuaian bobot yang terlalu agresif (*overshooting*), membuat varian Medium kesulitan menemukan titik keseimbangan parameter pada konfigurasi ini.

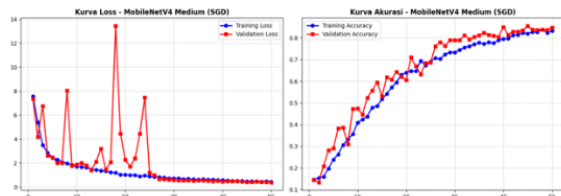
- Varian Large: Mengalami lonjakan *validation loss* sesaat yang cukup tinggi hingga menyentuh nilai ≈ 64 pada *epoch* ke-2. Namun, berkat kestabilan suplai data gradien pada *batch size* 32, pengoptimasi Adam mampu meredam lonjakan tersebut secara cepat pada *epoch* ke-3, membawa kurva *loss* melandai stabil mendekati nol, dan menuntaskan pelatihan di *epoch* ke-23 dengan akurasi validasi melampaui 88%.

4.9. Optimizer SGD (Batch Size 32)

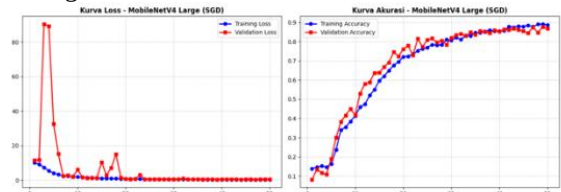
a. Small



b. Medium



c. Large



Gambar 6. Grafik loss dan akurasi pelatihan optimizer SGD batch size 32: (a) Small, (b) Medium, (c) Large]

Pada varian Small, kurva *training loss* dan *validation loss* menurun secara konsisten dan saling berdekatan sejak *epoch* awal, sementara akurasi pelatihan dan validasi meningkat stabil hingga akurasi validasi mendekati 90%. Pelatihan berhenti pada sekitar *epoch* 33 akibat *early stopping*, yang menandakan model telah konvergen dengan baik tanpa indikasi *overfitting* yang berarti. Pada varian Medium, akurasi validasi naik cepat pada *epoch* awal namun kurva *validation loss* menunjukkan fluktuasi yang cukup besar dan cenderung meningkat di *epoch* akhir, yang mengindikasikan bahwa varian Medium kurang stabil pada konfigurasi ini meskipun akurasi pelatihannya tinggi. Pada varian Large, terdapat lonjakan *validation loss* yang tajam pada *epoch* awal yang kemudian turun dan stabil mendekati nol, sedangkan akurasi validasi meningkat stabil hingga melampaui 88%, sehingga varian Large menunjukkan kestabilan yang lebih baik dibandingkan Medium.

4.10. Hasil Evaluasi Data Uji Batch Size 32

Tabel 6. Hasil evaluasi data uji batch size 32

Optimizer	Varian	Kategori	Precision	Recall	F1-Score	Accuracy
Adam	Small	Tea algal leaf spot	75%	95%	84%	90,53%
		Brown blight	82%	74%	78%	
		Gray blight	95%	86%	90%	
		Helopeltis	95%	87%	91%	
		Red spider	87%	88%	88%	
		Green mirid bug	92%	95%	93%	
		Healthy leaf	96%	100%	98%	
Adam	Medium	Tea algal leaf spot	79%	81%	80%	87,50%
		Brown blight	70%	70%	70%	
		Gray blight	83%	97%	89%	
		Helopeltis	81%	85%	83%	
		Red spider	100%	75%	86%	
		Green mirid bug	92%	86%	89%	
		Healthy leaf	100%	100%	100%	
Adam	Large	Tea algal leaf spot	75%	95%	84%	90,53%
		Brown blight	90%	70%	79%	
		Gray blight	95%	90%	92%	
		Helopeltis	87%	85%	86%	
		Red spider	87%	92%	90%	
		Green mirid bug	90%	93%	92%	
		Healthy leaf	100%	99%	99%	
SGD	Small	Tea algal leaf spot	65%	81%	72%	81,82%
		Brown blight	69%	74%	71%	
		Gray blight	92%	75%	83%	
		Helopeltis	72%	67%	69%	
		Red spider	76%	96%	85%	
		Green mirid bug	86%	78%	82%	
		Healthy leaf	94%	100%	97%	
SGD	Medium	Tea algal leaf spot	66%	95%	78%	86,17%
		Brown blight	77%	48%	59%	
		Gray blight	91%	88%	89%	
		Helopeltis	82%	89%	85%	
		Red spider	88%	85%	86%	
		Green mirid bug	91%	86%	88%	
		Healthy leaf	93%	100%	96%	
SGD	Large	Tea algal leaf spot	80%	88%	84%	87,88%
		Brown blight	86%	72%	78%	
		Gray blight	92%	89%	90%	
		Helopeltis	73%	87%	79%	
		Red spider	94%	90%	92%	
		Green mirid bug	91%	84%	88%	
		Healthy leaf	93%	99%	96%	

Berdasarkan Tabel 6, pada *batch size* 32 optimizer Adam memberikan performa lebih tinggi dibandingkan SGD pada varian Small dan Large, dengan akurasi tertinggi sebesar 90,53% yang dicapai oleh MobileNetV4-Conv-Small dan MobileNetV4-Conv-Large. Pada kedua varian terbaik tersebut, kelas *healthy leaf* dapat dikenali hampir sempurna dengan *F1-score* 98–99%, sedangkan kesalahan paling banyak terjadi pada kelas *brown blight* yang memiliki *recall* relatif rendah (70–74%). Pada optimizer SGD, akurasi tertinggi diperoleh varian Large sebesar 87,88%.

4.11. Perbandingan Seluruh Skenario

Perbandingan akurasi pengujian seluruh skenario disajikan secara ringkas pada Tabel 7.

Tabel 7. Perbandingan akurasi pengujian seluruh skenario

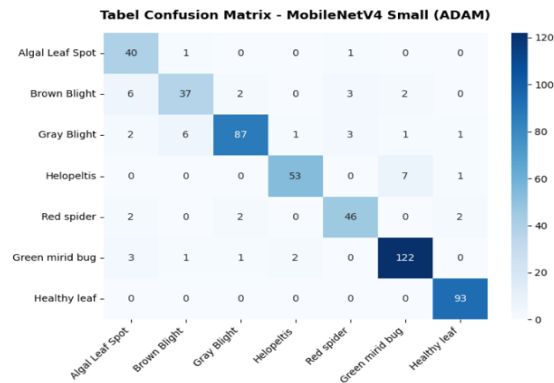
Varian	Adam (BS 16)	Adam (BS 32)	SGD (BS 16)	SGD (BS 32)
Small	87,31%	90,53%	82,01%	81,82%
Medium	89,58%	87,50%	86,74%	86,17%
Large	85,80%	90,53%	87,50%	87,88%

Berdasarkan Tabel 7, optimizer Adam secara konsisten menghasilkan akurasi lebih tinggi dibandingkan SGD pada hampir seluruh kombinasi varian dan *batch size*. Akurasi tertinggi secara keseluruhan sebesar 90,53% diperoleh konfigurasi Adam *batch size* 32 pada varian Small dan Large. Selain itu, pemilihan optimizer terlihat lebih berpengaruh terhadap performa model dibandingkan variasi *batch size*, karena selisih akurasi

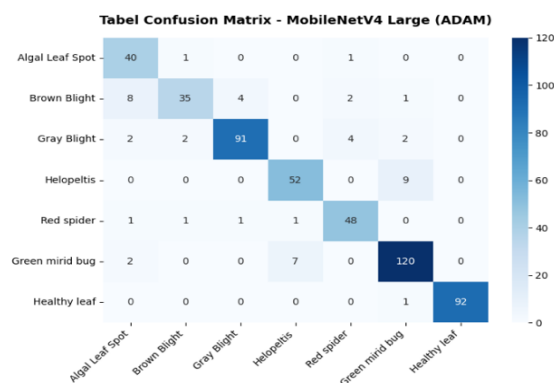
antarooptimizer (mencapai sekitar 8% pada varian Small) jauh lebih besar daripada selisih akibat perubahan *batch size* pada optimizer yang sama.

4.12. Analisis Confusion Matrix

Confusion matrix dari dua konfigurasi dengan akurasi tertinggi, yaitu MobileNetV4-Conv-Small dan MobileNetV4-Conv-Large dengan optimizer Adam *batch size* 32, ditunjukkan pada Gambar 7 dan Gambar 8.



Gambar 7. Confusion matrix MobileNetV4-Conv-Small (Adam, batch size 32)



Gambar 8. Confusion matrix MobileNetV4-Conv-Large (Adam, batch size 32)

MobileNetV4-Conv-Small mampu mengklasifikasikan sebagian besar citra dengan benar, ditunjukkan oleh dominasi nilai pada diagonal utama. Kelas healthy leaf terklasifikasi sempurna dengan seluruh 93 citra benar, diikuti green mirid bug dengan 122 dari 129 citra benar dan gray blight dengan 87 dari 101 citra benar. Kesalahan klasifikasi paling menonjol terjadi pada kelas brown blight, di mana 6 citra terklasifikasi sebagai algal leaf spot dan beberapa lainnya tersebar ke gray blight dan green mirid bug. Selain itu, 7 citra helopeltis terklasifikasi sebagai green mirid bug. Pola serupa terlihat pada Gambar 8 untuk varian Large, di mana kesalahan utama juga terjadi pada brown blight (8 citra terklasifikasi sebagai algal leaf spot) dan helopeltis (9 citra terklasifikasi sebagai green mirid bug).

Kesalahan tersebut disebabkan oleh kemiripan karakteristik visual antarkelas, yaitu pola bercak dan kerusakan jaringan pada brown blight yang menyerupai algal leaf spot dan gray blight, serta pola bintik akibat serangan hama pada helopeltis yang menyerupai green mirid bug. Secara keseluruhan, kedua model tetap menunjukkan kemampuan generalisasi yang baik dalam mengklasifikasikan tujuh kategori penyakit daun teh.

4.13. Pembahasan

Hasil penelitian menunjukkan bahwa arsitektur MobileNetV4 dengan pendekatan *transfer learning* dan strategi *full fine tuning* efektif digunakan untuk klasifikasi penyakit daun teh pada dataset teaLeafBD, dengan akurasi tertinggi sebesar 90,53%. Temuan utama penelitian ini adalah bahwa pemilihan optimizer memberikan pengaruh yang lebih besar terhadap performa model dibandingkan pemilihan *batch size*. Optimizer Adam secara konsisten mengungguli SGD pada hampir seluruh konfigurasi, yang menegaskan bahwa karakteristik *adaptive learning rate* lebih sesuai untuk dataset dengan variasi visual antarkelas yang tinggi seperti pada penyakit daun teh. Temuan ini sejalan dengan kecenderungan pada penelitian klasifikasi penyakit daun lainnya yang menunjukkan keunggulan optimizer Adam dibandingkan SGD dalam mencapai akurasi yang lebih tinggi.

Capaian akurasi sebesar 90,53% pada penelitian ini relatif sebanding dengan penelitian terdahulu pada domain penyakit daun teh, seperti MobileNetV2 yang memperoleh akurasi 94,6% [7] dan MobileNetV3 dengan *coordinate attention* sebesar 95,88% [8]. Perlu dicatat bahwa penelitian ini menggunakan dataset teaLeafBD yang memiliki tujuh kelas dengan distribusi yang tidak seimbang, sehingga perbandingan langsung antarpemelitian perlu dilakukan secara hati-hati karena perbedaan jumlah kelas, ukuran dataset, dan tingkat keseimbangan data.

Dari sisi efisiensi, varian MobileNetV4-Conv-Small mampu mencapai akurasi yang setara dengan varian Large (sama-sama 90,53%) meskipun memiliki jumlah parameter yang jauh lebih sedikit. Oleh karena itu, varian Small dengan optimizer Adam dan *batch size* 32 dipilih sebagai konfigurasi terbaik karena menawarkan keseimbangan paling optimal antara akurasi dan efisiensi komputasi, sehingga lebih sesuai untuk implementasi pada perangkat dengan keterbatasan sumber daya.

Meskipun demikian, penelitian ini memiliki keterbatasan. Dataset yang digunakan masih berada pada kondisi terkontrol dengan latar belakang putih dan pencahayaan yang seragam, sehingga belum merepresentasikan kondisi pengambilan citra di lapangan. Selain itu, ketidakseimbangan distribusi kelas berkontribusi pada rendahnya *recall* pada kelas minoritas seperti brown blight. Hal ini menjadi peluang pengembangan pada penelitian selanjutnya.

5. KESIMPULAN DAN SARAN

Berdasarkan hasil pengujian terhadap 12 skenario yang memvariasikan tiga varian arsitektur MobileNetV4-Conv, dua optimizer, dan dua nilai *batch size* pada dataset teaLeafBD, dapat disimpulkan bahwa arsitektur MobileNetV4 dengan pendekatan *transfer learning* dan strategi *full fine tuning* mampu mengklasifikasikan tujuh kategori penyakit daun teh dengan baik, di mana pemilihan optimizer memegang peranan paling menentukan karena optimizer Adam secara konsisten mengungguli SGD dengan akurasi tertinggi sebesar 90,53% yang dicapai konfigurasi MobileNetV4-Conv-Small dan MobileNetV4-Conv-Large pada *batch size* 32, sedangkan konfigurasi SGD tertinggi hanya mencapai 87,88%, sehingga MobileNetV4-Conv-Small dengan optimizer Adam dan *batch size* 32 ditetapkan sebagai model terbaik karena menyeimbangkan akurasi maksimal dengan efisiensi komputasi, dan untuk penelitian selanjutnya disarankan dilakukan penanganan ketidakseimbangan kelas melalui *class weighting* atau augmentasi terarah pada kelas minoritas, penambahan mekanisme *attention* yang memanfaatkan komponen MQA pada MobileNetV4, serta pengujian model pada citra daun teh yang diambil langsung di lapangan dengan kondisi pencahayaan dan latar belakang yang lebih bervariasi.

DAFTAR PUSTAKA

- [1] Badan Pusat Statistik, *Statistik Perkebunan Indonesia 2024: Teh*. Jakarta: Badan Pusat Statistik, 2025.
- [2] K. Pandey, G. D. Sinniah, A. Babu, and A. Tanti, "How the Global Tea Industry Copes With Fungal Diseases: Challenges and Opportunities," *Plant Disease*, vol. 105, no. 7, pp. 1868–1879, 2021.
- [3] M. S. Alam *et al.*, "teaLeafBD: A comprehensive image dataset to classify the diseased tea leaf to automate the leaf selection process in Bangladesh," *Data in Brief*, vol. 61, p. 111769, 2025.
- [4] U. Hairah, A. Septiarini, N. Puspitasari, A. Tejawati, and H. Hamdani, "Classification of tea leaf disease using convolutional neural network approach," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 14, no. 3, pp. 3287–3294, 2024.
- [5] M. N. Winnarto, M. Mailasari, and A. Purnamawati, "Klasifikasi Jenis Penyakit Daun Teh Menggunakan Arsitektur VGG16," *Jurnal Sistem Informasi dan Teknologi (SINTEK)*, vol. 6, no. 1, pp. 121–126, 2022.
- [6] S. Azis and B. Irawan, "Evaluasi Klasifikasi Penyakit Daun Teh Menggunakan Transfer Learning EfficientNetB0," *Jurnal Informatika dan Teknik Elektro Terapan (JITET)*, vol. 14, no. 1, 2026.
- [7] I. G. D. Dwijayana and I. G. A. Wibawa, "Implementasi Transfer Learning Dalam Klasifikasi Penyakit Pada Daun Teh Menggunakan MobileNetV2," *Jurnal Nasional Teknologi Informasi dan Aplikasinya (JNATIA)*, vol. 1, no. 1, pp. 379–387, 2022.
- [8] Y. Li *et al.*, "Tea leaf disease and insect identification based on improved MobileNetV3," *Frontiers in Plant Science*, vol. 15, pp. 1–12, 2024.
- [9] F. A. Pratama, M. Solihin, and S. Mujilawati, "Deep Learning-Based CNN for Tea Leaf Disease Classification," *Journal of Artificial Intelligence and Engineering Applications*, vol. 5, no. 1, 2025.
- [10] M. N. Anugrah and N. Rachmat, "Klasifikasi Penyakit Daun Tanaman Jagung Menggunakan Pendekatan Transfer Learning Arsitektur MobileNetV4," *Jurnal Teknologi dan Manajemen Industri Terapan (JTMIT)*, vol. 4, no. 4, pp. 2174–2182, 2025.
- [11] G. Chen, Z. Xia, X. Ma, Y. Jiang, and Z. He, "MobileNet-GDR: A lightweight algorithm for grape leaf disease identification based on improved MobileNetV4-small," *Frontiers in Plant Science*, vol. 16, pp. 1–17, 2025.
- [12] Y. Wan, L. Zeng, Q. Meng, L. Yuan, and H. Tong, "Characterization of *Pestalotiopsis chamaeropsis* causing gray blight disease on tea leaves (*Camellia sinensis*) in Chongqing, China," *Canadian Journal of Plant Pathology*, vol. 43, no. 3, pp. 413–420, 2021.
- [13] N. T. S. Saptadi *et al.*, *Deep Learning: Teori, Algoritma, dan Aplikasi*. Tasikmalaya: PT Sada Kurnia Pustaka, 2025.
- [14] B. Purnama, *Pengantar Machine Learning: Konsep dan Praktikum dengan Contoh Latihan Berbasis R dan Python*. Bandung: Penerbit Informatika, 2019.
- [15] V. H. Phung and E. J. Rhee, "A High-Accuracy Model Average Ensemble of Convolutional Neural Networks for Classification of Cloud Image Patches on Small Datasets," *Applied Sciences*, vol. 9, no. 21, p. 4500, 2019.
- [16] A. N. Inaya, A. U. Rahma, M. Jannah, L. R. K. Arafah, L. Ishak, and M. R. Edy, "Klasifikasi Citra Dengan Pendekatan Transfer Learning Pada Gambar Fauna Terbang," *Jurnal Media Pendidikan Teknik Informatika dan Komputer*, vol. 7, no. 1, pp. 85–89, 2024.
- [17] I. G. N. A. B. Adhipramana, A. Y. Husodo, and A. Aranta, "Implementasi Deep Learning untuk Image Classification menggunakan Arsitektur Convolutional Neural Network (CNN) pada Citra Sampah Hotel," Universitas Mataram, 2023.
- [18] D. Qin *et al.*, "MobileNetV4: Universal Models for the Mobile Ecosystem," *arXiv preprint arXiv:2404.10518*, pp. 1–18, 2024.