

Implementasi CBAM pada Arsitektur ResNet50 dalam Klasifikasi Penyakit Daun Tanaman Kentang

DOI: <http://dx.doi.org/10.35889/jutisi.v15i1.3411>

Creative Commons License 4.0 (CC BY – NC)

Vicky Yanto^{1*}, Nur Rachmat²

Informatika, Universitas Multi Data Palembang, Palembang, Indonesia

*e-mail Corresponding Author: vicky.yanto2004@gmail.com

Abstract

Potato cultivation is inevitably susceptible to various challenges, particularly leaf diseases. Failure to address these issues effectively can lead to a significant decline in both crop yield and harvest quality. This study aims to implement the Convolution Block Attention Module (CBAM) within the ResNet50 architecture for the classification of potato leaf diseases. The dataset utilized in this research comprises 2,152 images categorized into three classes: 152 healthy leaves, 1,000 early blight leaves, and 1,000 late blight leaves. The data was partitioned into training, validation, and testing sets with a ratio of 80:10:10, respectively. Image augmentation techniques were employed to address the class imbalance by increasing the number of healthy leaf images and enhancing dataset variability. Experimental results demonstrate that the ResNet50+CBAM model achieved the highest accuracy of 92% in both Scenario 1 (Adam optimizer, batch size 16) and Scenario 3 (Adam optimizer, batch size 32). Conversely, Scenario 4 (SGD optimizer, batch size 32) yielded the lowest accuracy at 77%.

Keyword: CBAM; Classification; CNN; Potato; ResNet50

Abstrak

Dalam membudidayakan suatu tanaman kentang pastinya tidak terlepas dari permasalahan yang terjadi dalam tanaman kentang salah satunya yaitu penyakit pada daun kentang, bila tidak diperhatikan dengan baik maka dapat terjadinya penurunan produksi dan penurunan kualitas pada hasil panen. Penelitian ini bertujuan untuk mengimplementasikan *Convolution Block Attention Module* (CBAM) pada arsitektur ResNet50 dalam klasifikasi penyakit daun tanaman kentang. Dataset yang digunakan dalam penelitian ini berjumlah 2152 gambar yang terdiri dari 3 kategori yaitu 152 daun sehat, 1000 daun *early blight* dan 1000 daun *late blight* yang akan dibagi menjadi 80% data latih, 10% data validasi dan 10% data uji. Penelitian ini menggunakan teknik augmentasi gambar yang bertujuan untuk menambah jumlah gambar daun sehat dan meningkatkan variasi data. Hasil pengujian menunjukkan ResNet50+CBAM pada skenario 1 (optimasi Adam dan *batch size* 16) dan skenario 3 (optimasi Adam dan *batch size* 32) menghasilkan akurasi terendah yaitu 77%.

Kata kunci: CBAM; CNN; Kentang; Klasifikasi; ResNet50

1. Pendahuluan

Kentang merupakan salah satu makanan yang banyak mengandung karbohidrat dan kentang menjadi makanan yang dikonsumsi ketiga setelah beras dan gandum [1]. Berdasarkan data dari Badan Pusat Statistik, hasil dari produksi kentang di Indonesia pada tahun 2022 mencapai 1,42 juta ton, mengalami peningkatan sebesar 4,21% dibandingkan produksi pada tahun 2021 mencapai angka 1,36 juta ton [2]. Dalam membudidayakan suatu tanaman kentang pastinya tidak terlepas dari permasalahan yang terjadi dalam tanaman kentang salah satunya yaitu penyakit pada daun kentang, bila tidak diperhatikan dengan baik maka dapat terjadinya penurunan produksi dan penurunan kualitas pada hasil panen [3]. Penurunan produksi kentang diduga berasal dari umbi-umbi bibit dan pertumbuhannya dipengaruhi oleh suhu lingkungan.

Jamur merupakan penyebab utama terjadinya penyakit pada daun kentang seperti *early blight* dan *late blight* [4].

Serangan penyakit *early blight* dan *late blight* disebabkan oleh patogen *Phytophthora infestans* merupakan faktor utama terjadinya penurunan produksi. Patogen *Phytophthora infestans* berkembang sangat cepat pada suhu 18-20°C kelembapan udara (RH) di atas 80% dan umumnya akan muncul setelah umur tanaman mencapai usia 5-6 minggu. Penyakit *early blight* dan *late blight* dapat menyerang daun dan batang tanaman kentang serta menyerang umbi kentang di lapangan atau di penyimpanan [5]. Gejala awal Penyakit *early blight* ditandai dengan bercak coklat tua dengan tepi jelas dengan ukuran yang berbeda pada berbagai bagian daun dan *late blight* ditandai bercak hijau tua hingga hitam dengan tepi kabur yang memberikan kesan basah [6]. Penyakit daun kentang seperti *early blight* dan *late blight* merupakan ancaman bagi petani kentang, solusi yang dapat diberikan dengan adanya identifikasi penyakit daun kentang dapat membantu petani dalam mengurangi kerugian yang diakibatkan penyakit [7].

Dengan seiring berkembangnya teknologi dalam mendeteksi gambar adapun beberapa pendekatan yang mampu mengidentifikasi gambar salah satunya yaitu *Convolutional Neural Network* (CNN). CNN mampu mempelajari pola gambar yang kompleks berdasarkan elemen visual yang ada pada gambar [8]. Salah satu arsitektur CNN yang mampu mengenali pola gambar yaitu ResNet50. ResNet50 dirancang untuk mengatasi masalah hilangnya gradien dengan mengintegrasikan koneksi lompatan yang memungkinkan transfer informasi antar lapisan tanpa kehilangan informasi yang signifikan [9]. Selain itu, *Convolutional Block Attention Module* (CBAM) sering digunakan untuk meningkatkan kinerja CNN dalam tugas klasifikasi. CBAM adalah modul atensi yang terdiri dari dua komponen utama yaitu *Channel Attention* (Atensi Saluran) sebagai menentukan fitur apa yang penting pada gambar dan *Spatial Attention* (Atensi Spasial) sebagai menyoroti lokasi fitur yang penting yang telah diproses oleh atensi saluran. Modul ini bekerja dengan menyesuaikan bobot fitur berdasarkan kepentingannya di domain saluran maupun spasial, sehingga meningkatkan representasi fitur-fitur yang relevan [10][11].

Penelitian ini bertujuan untuk melakukan klasifikasi penyakit daun tanaman kentang menggunakan arsitektur ResNet50 dengan menambahkan *Attention Mechanism* CBAM untuk mendapatkan evaluasi kinerja model yang baik. Diharapkan model ini dapat memberikan kontribusi yang baru dan bermanfaat untuk petani serta menjadi solusi dalam mengatasi permasalahan penyakit pada daun tanaman kentang secara efisien dan efektif.

2. Tinjauan Pustaka

Beberapa penelitian terdahulu telah melakukan klasifikasi penyakit daun tanaman kentang menggunakan metode *Convolution Neural Network* (CNN). Dalam penelitian yang dilakukan [1] mengimplementasikan metode deep learning menggunakan *Convolutional Neural Network* (CNN) dan ekstraksi fitur VGG16 dalam klasifikasi penyakit daun tanaman kentang dengan menggunakan 900 gambar. Hasil pengujian model tersebut didapatkan nilai hasil akurasi sebesar 94% pada model CNN. Selanjutnya peneliti melakukan ekstraksi fitur dengan metode VGG16 untuk meningkatkan hasil akurasi 93,3%.

Pada penelitian yang dilakukan [12] membandingkan metode *Transfer Learning* yang digunakan pada CNN, VGG16, InceptionV3 dan ResNet50 dalam identifikasi penyakit daun kentang dengan menggunakan 450 gambar yang dibagi menjadi 3 kategori yaitu 150 daun sehat, 150 daun *early blight* dan 150 daun *late blight*. Hasil pengujian model tersebut didapatkan nilai hasil akurasi pada model CNN mencapai 80%, model VGG16 mencapai 95%, model InceptionV3 mencapai 78% dan model ResNet50 mencapai 78%.

Penelitian sebelumnya telah mengimplementasikan *Convolution Block Attention Module* (CBAM) pada metode CNN dalam klasifikasi penyakit daun kentang. Penelitian yang dilakukan [13] memanfaatkan *transfer learning* dan membandingkan VGG16, ResNet50, MobileNetV1, ViT serta menambahkan CBAM pada VGG16 pada klasifikasi penyakit daun kentang dengan menggunakan 3129 gambar penyakit daun kentang 3 kategori yaitu 975 daun sehat, 1082 daun *early blight* dan 1072 daun *virus disease*. Hasil pengujian model tersebut menunjukkan bahwa nilai akurasi VGG16 mencapai 97,23%, ResNet50 mencapai 81,36, MobileNetV1 mencapai 59,96, ViT mencapai 97,02. Namun saat menambahkan *Attention Mechanism* CBAM akurasi VGG16 meningkat menjadi 97,76%, dapat disimpulkan bahwa CBAM dapat membantu meningkatkan kinerja model.

Berdasarkan penelitian-penelitian yang dilakukan sebelumnya telah menunjukkan bahwa metode CNN pada arsitektur ResNet50 memiliki akurasi yang kurang baik dibandingkan dengan

arsitektur lainnya. Penelitian ini bertujuan untuk meningkatkan kinerja model pada arsitektur ResNet50 dengan menambahkan *Attention Mechanism* CBAM, penelitian ini juga menggunakan optimasi seperti Adam dan SGD serta menggunakan *pre-trained* (ImageNet) dan teknik *transfer learning*.

3. Metodologi

3.1. Pengumpulan Dataset

Pada tahap ini, dilakukan proses pengumpulan dataset berupa *dataset* penyakit daun tanaman kentang. *Dataset* yang diambil dari publik di website Kaggle dengan nama PlantVillage Dataset (<https://www.kaggle.com/datasets/emmarex/plantdisease>). *Dataset* berisi 2152 citra dengan terdiri dari 3 kategori yaitu, daun sehat, daun *early blight* dan daun *late blight*. Rincian dataset yang digunakan dapat dilihat pada tabel 1.

Tabel 1. Dataset PlantVillage Daun Kentang		
Kategori	Jumlah	Contoh Gambar
Daun Sehat	152	
Daun <i>Early Blight</i>	1000	
Daun <i>Late Blight</i>	1000	

3.2. Preprocessing Data

Pada tahapan ini dilakukan perancangan sistem yang dibutuhkan. Proses awal dengan pengumpulan dataset citra penyakit daun tanaman kentang, lalu dilakukan *preprocessing* yang terdiri dari *resize* 224x224 untuk semua kategori dan augmentasi data pada kategori daun sehat yang bertujuan untuk menyeimbangkan data gambar menjadi 1000 dengan cara di *flip*, rotasi dan *zoom* pada kelas daun sehat.

3.3. Pembagian Data

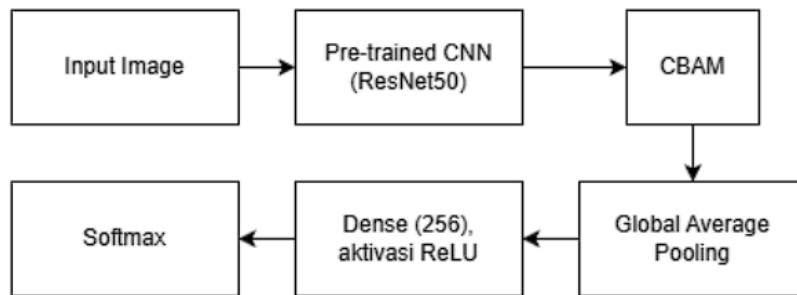
Proses selanjutnya berupa pembagian *dataset* yang sebelumnya telah dilakukan proses *preprocessing* dan augmentasi data menjadi tiga bagian yaitu data pelatihan, data validasi dan

data pengujian. Dataset nantinya akan dibagi menjadi 80% data pelatihan, 10% data validasi dan 10% data pengujian.

3.4. Pembuatan Model

Pada tahap selanjutnya merupakan tahapan pembuatan model menggunakan arsitektur ResNet50 yang telah *pre-trained* pada ImageNet dengan metode *transfer learning*. Pada *transfer learning* akan mengambil seluruh model yang telah di *pre-trained* dan melakukan tahapan *Feature Extraction* dengan cara membekukan semua lapisan konvolusi yang bertujuan untuk membiarkan model lebih belajar sebelum melakukan *Fine-Tuning* dan memanfaatkan fitur visual yang telah dipelajari ResNet50 saat melakukan *pre-trained* ImageNet. Setelah 48 membekukan lapisan konvolusi akan dilakukan *Fine-Tuning* dengan cara mencairkan 30 lapisan konvolusi terakhir yang telah dibekukan sebelumnya yang bertujuan untuk mempelajari lebih dalam terhadap bercak penyakit, tekstur dan warna serta menambahkan modul CBAM yang berfungsi untuk meningkatkan kemampuan ResNet50 dengan memperkuat area penting serta meningkatkan akurasi.

Pada proses *Fine-tuning* berlangsung akan digunakan 3 *callback* seperti *EarlyStopping*, *ReduceLROnPlateau* dan *ModelCheckpoint*. *EarlyStopping* yang berfungsi sebagai mencegah *Overfitting*, *ReduceLROnPlateau* yang berfungsi sebagai mengecilkan *learning rate* untuk membantu model mencapai titik terbaik dan *ModelCheckpoint* berfungsi sebagai menyimpan bobot model yang digunakan untuk evaluasi. Model akan dilatih menggunakan dua jenis optimasi yaitu optimasi Adam dan optimasi SGD. Rancangan ResNet50 dan CBAM dapat dilihat pada gambar 1.



Gambar 1. Rancangan Arsitektur ResNet50 dan CBAM

3.5 Pengujian Model dan Evaluasi Model

Pada tahapan ini, sistem yang telah dibuat sebelumnya akan melakukan uji coba terhadap data uji yang telah dibagi. Hasil yang diperoleh akan dievaluasi menggunakan confusion matrix yang bertujuan untuk mendapatkan hasil yang dihasilkan model yang telah dirancang berupa nilai *accuracy*, *precision*, *recall* dan *F1-Score* yang dapat dilihat pada Persamaan (1), (2), (3) dan (4).

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

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

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

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

Confusion Matrix digunakan sebagai untuk menghitung kinerja model klasifikasi, *matrix accuracy* berfungsi sebagai seberapa tepat atau akurat model dalam klasifikasi. *Matrix Precision* berfungsi sebagai mengukur proporsi data yang diprediksi *positif* yang sebenarnya *positif* dari semua data yang 52 diprediksi *positif*. *Matrix Recall* digunakan untuk mengukur proporsi data yang diprediksi *positif* yang sebenarnya *positif* dari semua data yang sebenarnya *positif*. *Matrix F1-Score* merupakan rata-rata dari presisi dan *recall* untuk mencari keseimbangan antara kedua *matrix* [14].

Selain menggunakan *confusion matrix*, performa model akan diukur dari proses *hyperparameter tuning*. *Hyperparameter tuning* merupakan proses menentukan kombinasi nilai

hyperparameter yang paling optimal untuk suatu model *machine learning* [15]. Untuk parameter yang digunakan dapat dilihat pada tabel 2.

Tabel 2. Parameter yang Digunakan

Parameter	Nilai/Jumlah yang Digunakan
Epoch	50
Batch Size	16 dan 32
Learning Rate Awal	0.0001
Optimasi	Adam dan SGD

4. Hasil dan Pembahasan

4.1. Realisasi Pembagian Dataset

Dataset yang digunakan berisi 2152 citra dengan terdiri dari 3 kategori yaitu, 152 daun sehat, 1000 daun *early blight* dan 1000 daun *late blight*, lalu daun sehat yang memiliki 152 gambar di augmentasi menjadi 1000 gambar. Dengan membagi menjadi 80:10:10:

- 1) Data latih: 2400 citra gambar (800 daun sehat, 800 daun *early blight*, 800 daun *late blight*).
- 2) Data validasi: 300 citra gambar (100 daun sehat, 100 daun *early blight*, 100 daun *late blight*).
- 3) Data uji: 300 citra gambar (100 daun sehat, 100 daun *early blight*, 100 daun *late blight*).

4.2. Skenario Pelatihan Model

Penelitian ini akan melakukan pelatihan model pada data latih dan data uji dengan 4 skenario pelatihan model untuk mencari model terbaik dari parameter dan parameter yang telah ditentukan. Skenario dapat dilihat pada tabel 3.

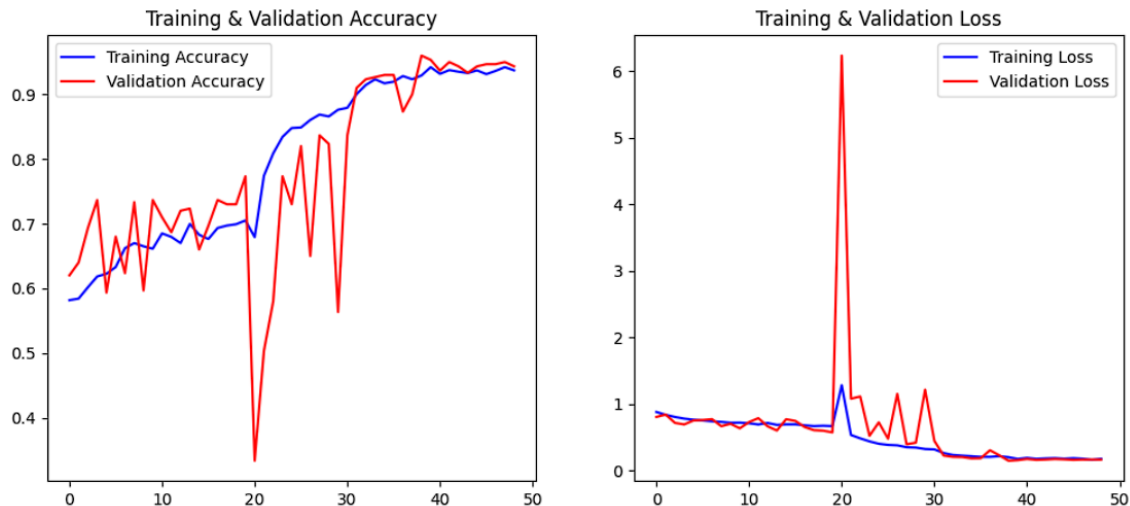
Tabel 3. Skenario Pelatihan Model

Skenario	Epoch	Batch Size	Learning Rate	Optimasi
1	50	16	0.0001	Adam
2	50	16	0.0001	SGD
3	50	32	0.0001	Adam
4	50	32	0.0001	SGD

Pada tabel 3. Skenario 1 akan menggunakan *Epoch* sebesar 50, *Batch Size* sebesar 16, *Learning rate* sebesar 0.0001 dan optimasi Adam. Skenario 2 akan menggunakan *Epoch* sebesar 50, *Batch Size* sebesar 16, *Learning rate* sebesar 0.0001 dan optimasi SGD. Skenario 3 akan menggunakan *Epoch* sebesar 50, *Batch Size* sebesar 32, *Learning rate* sebesar 0.0001 dan optimasi Adam. Skenario 4 akan menggunakan *Epoch* sebesar 50, *Batch Size* sebesar 32, *Learning rate* sebesar 0.0001 dan optimasi SGD. Semua skenario pada tabel menggunakan *input* gambar yang *resize* menjadi 224x224 dan dibagi menjadi 3 data yaitu 80% data latih, 10% data validasi dan 10% data uji dan khusus *learning rate* akan menggunakan teknik *ReduceLROnPlateau*, nilai *learning rate* akan berubah yang bertujuan untuk mengurangi *overfitting*.

4.3. Hasil Pelatihan Model Skenario 1

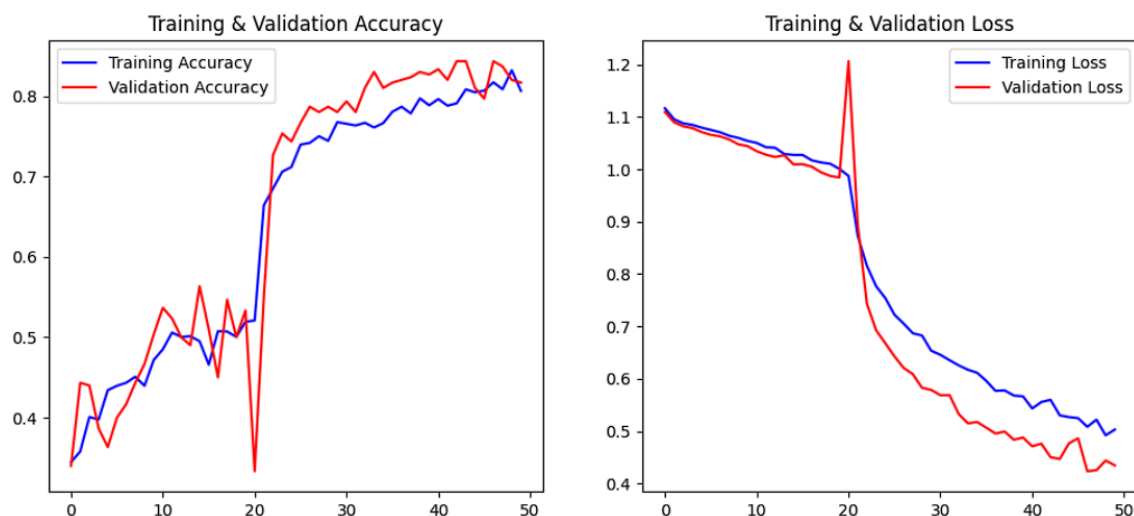
Pada grafik skenario 1 menunjukkan bahwa proses fase *Feature Extraction* (*freeze layer*) pada (*Epoch* 0-20) mencapai akurasi sekitar 70-75% dan selisih loss yang tidak terlalu jauh antara data latih dan data validasi. Namun saat transisi ke proses *Fine-Tuning* pada *Epoch* 20 terjadi lonjakan yang disebabkan dari 30 *layer* terakhir (*unfreeze*). Pada proses *Fine-Tuning* pada (*Epoch* 21-50) terjadi peningkatan akurasi hingga mencapai diatas 90%, nilai loss mencapai dibawah 0.2 dan proses berhenti saat *Epoch* 49 yang disebabkan oleh *callback EarlyStopping*.



Gambar 2. Grafik Accuracy dan Loss Skenario 1

4.4. Hasil Pelatihan Model Skenario 2

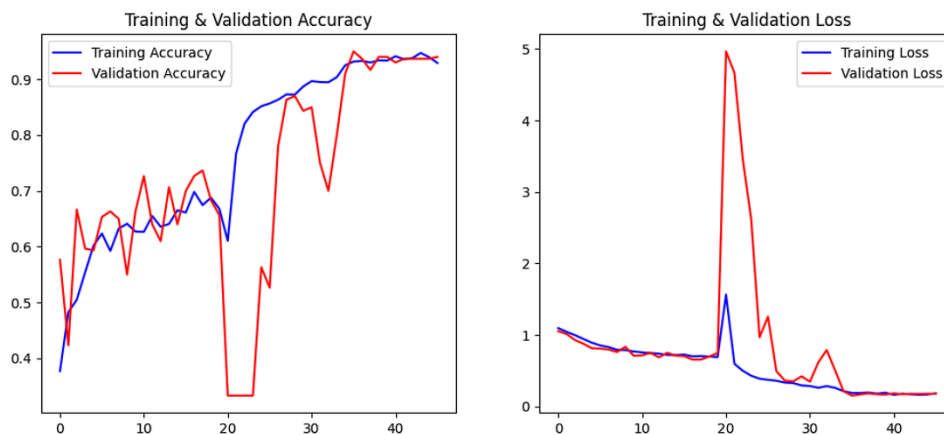
Pada grafik skenario 2 menunjukkan bahwa proses fase *Feature Extraction (freeze layer)* pada (*Epoch 0-20*) mencapai akurasi sekitar 60-65% dan selisih loss yang tidak terlalu jauh antara data latih dan data validasi. Namun saat transisi ke proses *Fine-Tuning* pada *Epoch 20* terjadi lonjakan yang disebabkan dari 30 *layer* terakhir (*unfreeze*). Pada proses *Fine-Tuning* pada (*Epoch 21-50*) terjadi peningkatan akurasi hampir mencapai 90%, nilai *training loss* mencapai sekitar 0.5 dan nilai validasi loss sekitar 0.4.



Gambar 3. Grafik Accuracy dan Loss Skenario 2

4.5. Hasil Pelatihan Model Skenario 3

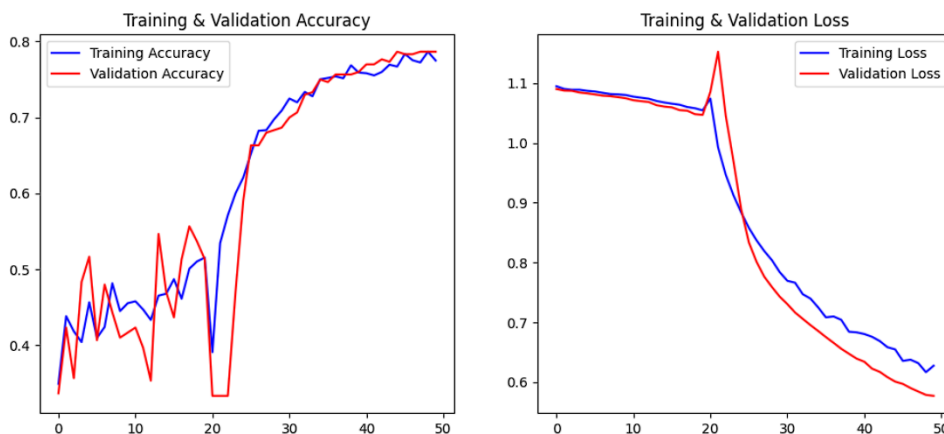
Pada grafik skenario 3 menunjukkan bahwa proses fase *Feature Extraction (freeze layer)* pada (*Epoch 0-20*) mencapai akurasi sekitar 70% dan selisih loss yang tidak terlalu jauh antara data latih dan data validasi. Namun saat transisi ke proses *Fine-Tuning* pada *Epoch 20* terjadi lonjakan yang disebabkan dari 30 *layer* terakhir (*unfreeze*). Pada proses *Fine-Tuning* pada (*Epoch 21-50*) terjadi peningkatan akurasi hingga mencapai diatas 90%, nilai loss mencapai dibawah 0.2 dan proses berhenti saat *Epoch 46* yang disebabkan oleh *callback EarlyStopping*.



Gambar 4. Grafik Accuracy dan Loss Skenario 3

4.6. Hasil Pelatihan Model Skenario 4

Pada grafik skenario 4 menunjukkan bahwa proses fase *Feature Extraction* (*freeze layer*) pada (*Epoch* 0-20) mencapai akurasi sekitar 40-50% dan selisih loss yang tidak terlalu jauh antara data latih dan data validasi. Namun saat transisi ke proses *Fine-Tuning* pada *Epoch* 20 terjadi lonjakan yang disebabkan dari 30 *layer* terakhir (*unfreeze*). Pada proses *Fine-Tuning* pada (*Epoch* 21-50) terjadi peningkatan akurasi hampir mencapai 80%, nilai *training loss* mencapai sekitar 0.65 dan nilai *validasi loss* sekitar 0.6.



Gambar 5. Grafik Accuracy dan Loss Skenario 4

4.7. Hasil Evaluasi Data Validasi Semua Skenario

Hasil Evaluasi dari data validasi yang terdiri dari 100 gambar daun sehat, 100 gambar daun *early blight* dan 100 gambar *late blight* menghasilkan nilai berupa *confusion matrix* yang terdiri dari nilai *akurasi*, nilai *presisi*, nilai *recall* dan nilai *F1-Score*. Hasil evaluasi semua skenario dapat dilihat pada tabel 4.

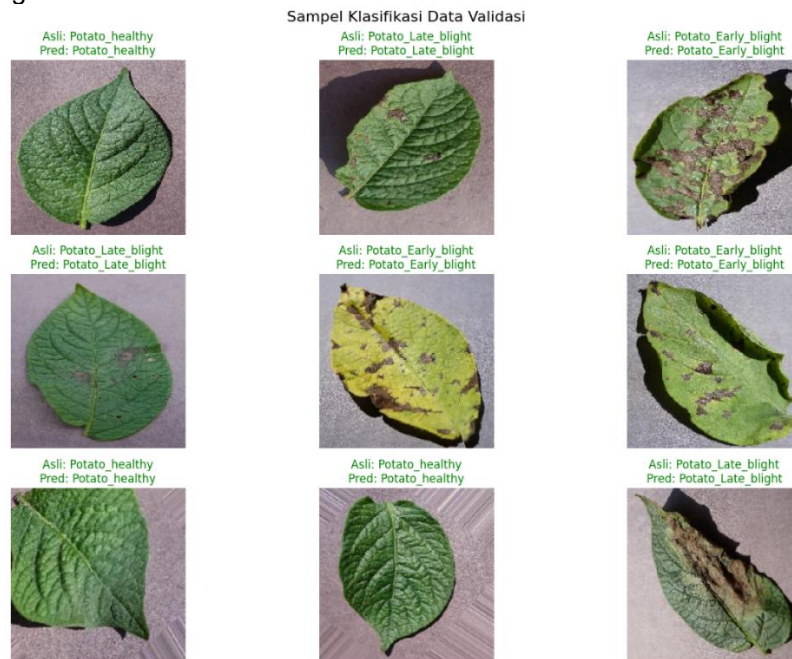
Tabel 4. Hasil Evaluasi Data Validasi

Skenario	Kategori	Presicion	Recall	F1-Score	Accuracy
Skenario 1	Daun Sehat	99%	88%	93%	93%
	Daun <i>Early Blight</i>	90%	96%	93%	
	Daun <i>Late Blight</i>	90%	94%	92%	
	Daun Sehat	88%	79%	83%	
Skenario 2	Daun <i>Early Blight</i>	82%	97%	89%	85%
	Daun <i>Late Blight</i>	87%	79%	83%	
	Daun Sehat	99%	90%	94%	
	Daun <i>Early Blight</i>	93%	98%	96%	
Skenario 3	Daun <i>Late Blight</i>	89%	93%	91%	94%

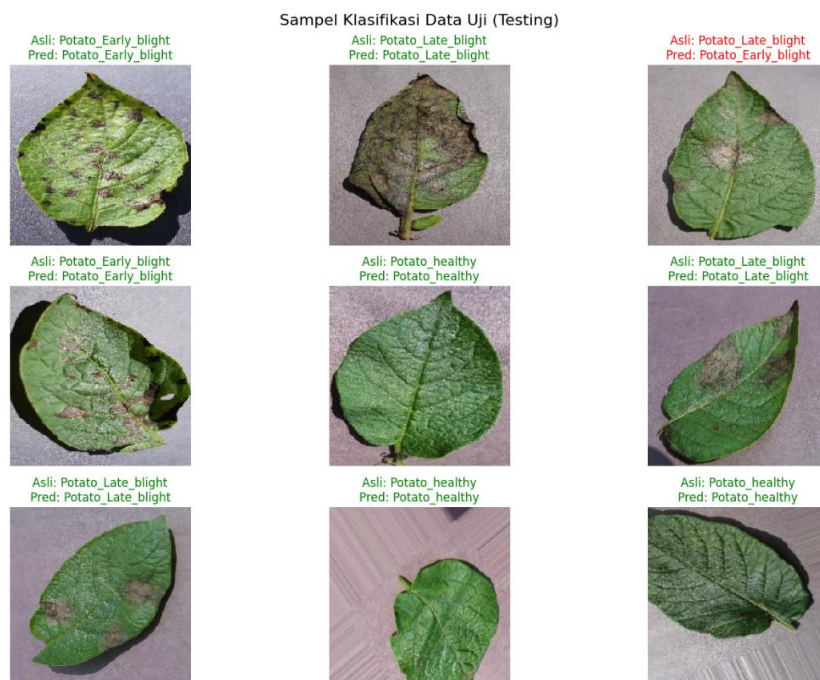
Skenario	Kategori	Precision	Recall	F1-Score	Accuracy
Skenario 4	Daun Sehat	89%	68%	77%	80%
	Daun <i>Early Blight</i>	73%	96%	83%	
	Daun <i>Late Blight</i>	81%	75%	78%	

4.8. Hasil Sample Data Validasi dan Data uji

Pada hasil data validasi dan data uji menunjukkan tulisan yang berwarna hijau merupakan hasil data yang diprediksi benar atau hasil prediksi sama dengan data dan tulisan yang berwarna merah merupakan hasil data yang diprediksi salah atau hasil prediksi berbeda dengan data asli. Berikut adalah hasil sampel data validasi dan data uji yang ditunjukkan pada gambar 6 dan gambar 7.



Gambar 6. Hasil Sampel Data Validasi



Gambar 7. Hasil Sampel Data Uji

4.9. Hasil Evaluasi Pengujian Semua Skenario

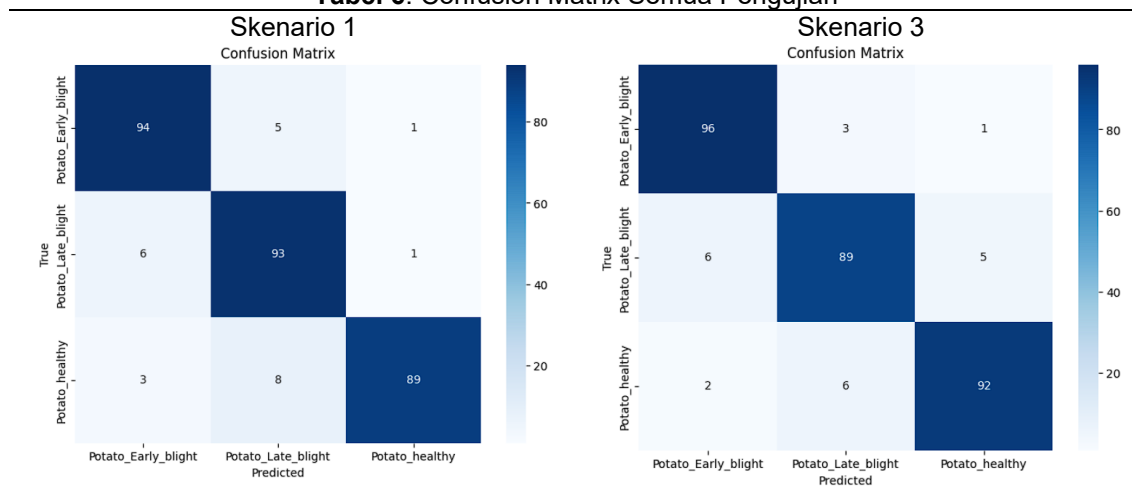
Model ResNet50 + CBAM yang diuji akan menghasilkan nilai evaluasi berupa *confusion matrix* yang terdiri dari nilai *akurasi*, nilai *presisi*, nilai *recall* dan nilai *F1-Score*. Pada skenario 1 menguji menggunakan optimasi Adam dan *batch size* 16 menghasilkan akurasi yang tinggi mencapai 92% dan hasil evaluasi menunjukkan bahwa model mampu mengenali semua kelas dengan baik dan presisi pada kelas daun sehat mencapai 98%. Pada skenario 2 menguji menggunakan optimasi SGD dan *batch size* 16 menghasilkan penurunan akurasi dari akurasi pada skenario 1 mencapai 84%. Pada skenario 3 menguji menggunakan optimasi Adam dan *batch size* 32 menghasilkan akurasi yang sama pada skenario 1 yaitu 92% namun penggunaan *batch size* yang lebih besar memberikan estimasi gradien yang lebih stabil. Pada skenario 4 menguji menggunakan optimasi SGD dan *batch size* 32 menghasilkan performa terburuk diantara semua skenario yang diuji dengan akurasi sebesar 77%. Kegagalan ini paling terlihat pada kelas daun sehat yang nilainya turun drastis pada *recall* hanya 65% dan *F1-Score* 75%, angka *recall* rendah menunjukkan bahwa model sering salah mengklasifikasikan daun sehat. Hasil evaluasi semua skenario dapat dilihat pada tabel 5.

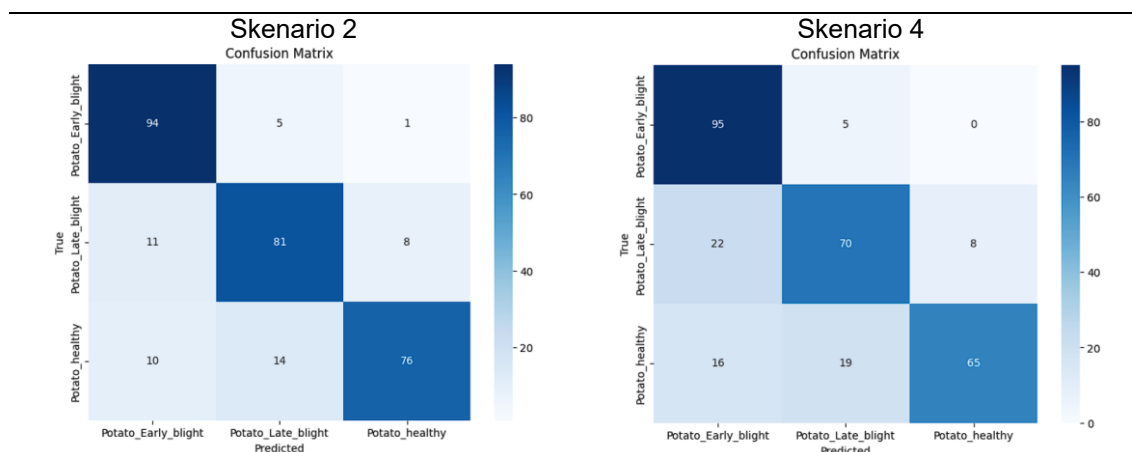
Tabel 5. Hasil Evaluasi Pengujian Semua Skenario

Skenario	Kategori	Presicion	Recall	F1-Score	Accuracy
Skenario 1	Daun Sehat	98%	89%	93%	92%
	Daun <i>Early Blight</i>	91%	94%	93%	
	Daun <i>Late Blight</i>	88%	93%	90%	
Skenario 2	Daun Sehat	89%	76%	82%	84%
	Daun <i>Early Blight</i>	82%	94%	87%	
	Daun <i>Late Blight</i>	81%	81%	81%	
Skenario 3	Daun Sehat	94%	92%	93%	92%
	Daun <i>Early Blight</i>	92%	96%	94%	
	Daun <i>Late Blight</i>	91%	89%	90%	
Skenario 4	Daun Sehat	89%	65%	75%	77%
	Daun <i>Early Blight</i>	71%	95%	82%	
	Daun <i>Late Blight</i>	74%	70%	72%	

Berdasarkan *confusion matrix* yang ditunjukkan pada tabel 6 dapat dilihat bahwa skenario 1 dan skenario 3 mampu mengklasifikasi dan membedakan daun sehat dan daun berpenyakit serta memiliki tingkat kesalahan yang rendah dibandingkan skenario 2 dan skenario 4 yang dimana skenario 2 dan skenario 4 masih banyak kesalahan pada mengklasifikasi daun sehat sebagai daun yang memiliki penyakit, skenario 2 mengklasifikasi daun sehat 76 dan skenario 4 mengklasifikasi daun sehat 65. Selain itu pemilihan *hyperparameter*, khususnya pada jenis optimasi lebih berpengaruh dalam menghasilkan akurasi yang tinggi dibandingkan variasi *batch size*.

Tabel 6. Confusion Matrix Semua Pengujian





4.10. Pembahasan

Hasil penelitian menunjukkan bahwa ResNet50 yang diimplementasikan dengan CBAM memiliki performa yang sangat baik dalam mengenali pola penyakit daun tanaman kentang, yang ditunjukkan oleh nilai *accuracy* yang didapatkan pada skenario pertama dan skenario ketiga. Dengan nilai *accuracy* sebesar 92% pada kedua skenario tersebut, namun memiliki perbedaan pada nilai *precision*, nilai *recall* dan nilai F1-Score yang dimana nilai *precision* dari skenario 1 lebih besar dibandingkan skenario 3 dan untuk nilai *recall* dan nilai F1-Score dari skenario 3 lebih besar dibandingkan skenario 1.

Jika dibandingkan dengan penelitian sebelumnya, performa model pada penelitian ini menunjukkan peningkatan pada model ResNet50. Penelitian yang dilakukan [12] yang menggunakan VGG16, InceptionV3 dan ResNet50 Hasil pengujian model tersebut didapatkan nilai hasil akurasi pada model CNN mencapai 80%, model VGG16 mencapai 95%, model InceptionV3 mencapai 78% dan model ResNet50 mencapai 78%. Meskipun model masih kurang mampu menyaingi VGG16 tetapi ResNet50 yang diimplementasikan dengan CBAM dapat menunjukkan peningkatan akurasi pada ResNet50 tanpa CBAM yang sebelumnya 78% menjadi 92%.

Pada penelitian [16] yang menggunakan CNN *baseline* dalam mengidentifikasi penyakit daun tanaman kentang menghasilkan hasil akurasi 82% pada validasi namun hasil evaluasi menunjukkan *overfitting* pada akurasi *training set* lebih tinggi dibandingkan *validation set* yang disebabkan oleh pada kelas yang memiliki gambar yang mirip. Sementara itu, pada penelitian ini menggunakan ResNet50 yang diimplementasi CBAM memiliki keunggulan dalam mendapatkan hasil yang lebih baik yaitu 94% pada validasi, Penelitian ini telah menambahkan 3 *callback* yang berfungsi untuk mencegah terjadinya *overfitting* pada saat *training dan validation* seperti *EarlyStopping*, *ReduceLROnPlateau* dan *ModelCheckpoint* dan Penelitian ini menggunakan CBAM sebagai membantu model fokus pada kelas penyakit yang memiliki visual mirip seperti *early blight* dan *late blight*.

Meskipun demikian, penelitian ini memiliki keterbatasan pada letak modul CBAM. Pada penelitian ini modul CBAM hanya diletakkan setelah base model ResNet50 selesai melakukan *pre-trained* yang berarti modul CBAM hanya bekerja pada hasil *feature map* terakhir. Selain itu, dataset yang digunakan masih berada pada latar belakang lingkungan yang terkontrol seperti latar belakang yang masih polos (belum dalam kondisi lapangan). Hal ini dapat menjadi tantangan untuk penelitian selanjutnya dalam pengembangan selanjutnya untuk memperluas cakupan model dalam mengenali bentuk citra penyakit daun. Jika dikembangkan lebih luas model ini berpotensi dapat membantu bidang pertanian dalam mengklasifikasi penyakit daun tanaman untuk meningkatkan kualitas tanaman dan hasil panen secara efektif.

5. Simpulan

Berdasarkan hasil pengujian terhadap empat skenario yang memvariasikan *optimizer* dan *batch size*, dapat disimpulkan bahwa pemilihan algoritma optimasi memegang peranan paling krusial dalam menentukan performa model ResNet50+CBAM. Optimasi Adam terbukti jauh lebih baik dibandingkan optimasi SGD, di mana Skenario 1 dan 3 menguji menggunakan optimasi Adam konsisten mencapai akurasi tertinggi sebesar 92%, sedangkan skenario yang menguji

menggunakan optimasi SGD tertinggal signifikan dengan performa terendah pada Skenario 4 yang hanya mencapai 77%. Kelemahan fatal SGD terlihat jelas pada ketidakmampuannya mendeteksi kelas daun sehat secara akurat, khususnya pada Skenario 4 yang mencatat nilai evaluasi *recall* terendah sebesar 65%. Oleh karena itu, Skenario 3 (Adam dengan Batch Size 32) ditetapkan sebagai model terbaik karena tidak hanya menghasilkan akurasi maksimal, tetapi juga menawarkan keseimbangan paling optimal antara stabilitas pelatihan dan efisiensi komputasi dibandingkan skenario lainnya.

Daftar Referensi

- [1] S. Lestari and K. I. Nauval, "Implementasi Deteksi Objek Penyakit Daun Kentang dengan Metode Convolutional Neural Network," *J. Apl. Teknol. Inf. dan Manaj.*, vol. 3, no. 2, pp. 136–149, 2022, doi: 10.31102/jatim.v3i2.1576.
- [2] R. Gunawan *et al.*, "Potato leaf disease classification with transfer learning using CNN optimization of MobilenetV2 architecture," *J. Comput. Sci. Inf. Technol.*, vol. 6, no. 2, pp. 254–258, 2025, doi: 10.37859/coscitech.v6i2.8599.
- [3] A. M. Lesmana, R. P. Fadhillah, and C. Rozikin, "Identifikasi penyakit pada citra daun kentang menggunakan Convolutional Neural Network (CNN)," *J. Sains dan Inform.*, vol. 8, no. 1, pp. 21–30, 2022, doi: 10.34128/jsi.v8i1.377.
- [4] S. Ghandi, I. M. Nugroho, and Y. R. Ramadhan, "Penerapan metode convolution neural network (CNN) dalam aplikasi pendeteksi penyakit daun tanaman kentang berbasis android," *J. Mhs. Tek. Inform.*, vol. 8, no. 5, pp. 8701–8708, 2024, doi: 10.36040/jati.v8i5.10769.
- [5] I. Nita, A. N. Putra, A. W. Sektiono, S. Riza, and K. Sigit, "Analisis kelayakan deteksi cepat penyakit hawar daun tanaman kentang pada fase akhir menggunakan UAV," *J. Hama Penyakit Tumbuh.*, vol. 11, no. 3, pp. 109–120, 2023, doi: 10.21776/ub.jurnalhpt.2023.011.3.2.
- [6] M. Muchtar and R. A. Muchtar, "Integrasi fitur warna, tekstur dan renji fraktal untuk klasifikasi penyakit daun kentang menggunakan linear discriminant analysis 1," *J. Mnemon.*, vol. 7, no. 1, pp. 77–84, 2024.
- [7] M. D. Pranatha, M. A. Maricar, and G. H. Setiawan, "Implementasi arsitektural Resnet-34 dalam klasifikasi gambar penyakit pada daun kentang," *J. Teknol. Dan Sist. Inf. Bisnis*, vol. 6, no. 3, pp. 575–580, 2024, doi: 10.47233/jteksis.v6i3.1431.
- [8] R. A. Boimau and Y. R. Kaesmetan, "Klasifikasi Citra Digital Bumbu dan Rempah Dengan Algoritma Convolutional Neural Network (CNN)," *Repeater Publ. Tek. Inform. dan Jar.*, vol. 2, no. 3, pp. 26–34, Jun. 2024, doi: 10.62951/repeater.v2i3.81.
- [9] M. R. Wibisono, I. Ma, and M. Defriani, "Batik motif classification using CNN with Resnet-50 architecture," *RISTEC Res. Inf. Syst. Technol.*, vol. 6, no. 1, pp. 41–59, 2025.
- [10] J. Ismail, L. Tanti, and Wanayumini, "Development of skin cancer pigment image classification using a combination of MobilenetV2 and CBAM," *J. Ilmu Pengetah. dan Teknol. Komput.*, vol. 10, no. 4, pp. 770–780, 2025, doi: 10.33480/jitk.v10i4.6541.
- [11] M. A. Rahman, M. S. Ullah, R. K. Devnath, and T. H. Chy, "Cotton Leaf Disease Detection : An Integration of CBAM with Deep Learning Approaches," *Int. J. Comput. Appl.*, vol. 186, no. 67, pp. 1–8, 2025, doi: 10.5120/ijca2025924487.
- [12] J. Rozaqi, A. Sunyoto, and R. Arief, "Implementasi transfer learning pada algoritma Convolutional Neural Network untuk identifikasi penyakit daun kentang," *Procedia Eng. Life Sci.*, vol. 1, no. 1, 2021, doi: 10.21070/pels.v1i1.820.
- [13] C. Zhang, S. Wang, C. Wang, H. Wang, Y. Du, and Z. Zong, "Research on a potato leaf disease diagnosis system based on deep learning," *Agriculture*, vol. 15, no. 4, pp. 1–24, Feb. 2025, doi: 10.3390/agriculture15040424.
- [14] T. H. Pinem and Z. P. Putra, "Evaluasi kinerja algoritma klasifikasi deep learning dalam prediksi diabetes," *J. Ilm. Fifo*, vol. 17, no. 1, pp. 17–28, 2025, doi: 10.22441/fifo.2025.v17i1.003.
- [15] B. Ramadhan and S. F. Pane, "Pengaruh hyperparameter tuning untuk efektivitas pada pendekatan hybrid dalam mendiagnosis stres dan depresi : tinjauan studi literatur," *J. Tekno Insentif*, vol. 18, no. 2, pp. 104–118, 2024, doi: 10.36787/jti.v18i2.1516.
- [16] K. Prasetyo and R. Mahendra, "Analisis kinerja Convolutional Neural Networks baseline untuk identifikasi jenis jenis penyakit kentang," *MALCOM Indones. J. Mach. Learn. Comput. Sci.*, vol. 5, no. 2, pp. 609–615, 2025, doi: 10.57152/malcom.v5i2.1722.