

# Identifikasi Penyakit Pada Tanaman Kopi Berdasarkan Citra Daun Menggunakan Metode Convolution Neural Network

Ahmad Fatchurrachman <sup>\*1</sup>, Daniel Udjulawa<sup>2</sup>

Universitas Multi Data Palembang, Jl. Rajawali No. 14 Palembang, 0711-037600  
Jurusan Informatika, Universitas Multi Data Palembang, Palembang  
e-mail: <sup>\*1</sup>ahmadfatchurrachman@gmail.com, <sup>2</sup>daniel@mdp.ac.id

## Abstrak

Tanaman kopi biasa dibuat untuk minuman yang dihasilkan dari biji kopi yang sudah dihaluskan sehingga menjadi bubuk. Salah satu penyebab kualitas pada kopi menurun disebabkan oleh hama yang bisa menyerang dari daun, batang dan akar. Dalam penelitian ini bertujuan untuk mengidentifikasi penyakit tanaman kopi berdasarkan daun menggunakan metode Convolution Neural Network (CNN) dengan arsitektur ResNet-50 dengan optimizer Adam. Total data dari dataset adalah 1664 citra gambar, pada dataset tersebut terdapat data train sebanyak 1264 citra gambar dan data test sebanyak citra 400 gambar. Hasil tertinggi pada pelatihan di penelitian ini dengan menggunakan 60 epoch dan optimizer Adam dengan nilai probabilitas pada learning\_rate sebesar 0.0001 mendapatkan nilai probabilitas sebesar 0.9969 dan nilai terendah mendapatkan nilai probabilitas sebesar 0.4918. Hasil pengujian terhadap data test pada penelitian ini mendapatkan tingkat akurasi sebesar 99%.

**Kata kunci**— Adam; CNN; Kopi; Optimizer; ResNet-50

## Abstract

Coffee plants are usually made for drinks made from coffee beans that have been ground into powder. One of the causes of decreased coffee quality is caused by pests that can attack from the leaves, stems and roots. This study aims to identify coffee plant diseases based on leaves using the Convolution Neural Network (CNN) method with the ResNet-50 architecture with the Adam optimizer. The total data from the dataset is 1664 images, in the dataset there are 1264 train data images and 400 test images. The highest result in training in this study using 60 epochs and Adam's optimizer with a probability value of learning\_rate of 0.0001 getting a probability value of 0.9969 and the lowest value getting a probability value of 0.4918. The results of testing the test data in this study obtained an accuracy rate of 99%.

**Keywords**— Adam; CNN; Coffee; Optimizer; ResNet-50



This is an open-access article under the [CC-BY-CA](https://creativecommons.org/licenses/by-sa/4.0/) license

## 1. PENDAHULUAN

Kopi merupakan jenis tanaman yang di olah biji yang sudah di haluskan untuk dijadikan sebuah minuman yang banyak diminati oleh masyarakat. Ada beberapa jenis kopi diantaranya Arabika, Robusta, Luwak, Tubruk dan masih banyak lagi, namun kopi mengandung sedikit nutrisi tetapi mengandung lebih dari ribuan bahan kimia alami seperti karbohidrat,

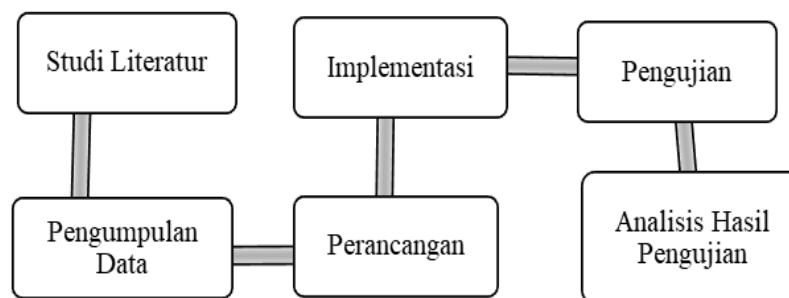
senyawa nitrogen, *lipid*, mineral, vitamin, senyawa fenolik, dan alkaloid. Kualitas kopi tergantung pada perawatan yang dilakukan, ketika tanaman kopi diserang hama maka akan menurunkan mutu pada kopi tersebut [1]. Hama tersebut dapat menyerang dari daun, batang, ataupun akar. Dalam dunia biologi, daun sering digunakan untuk menandakan kesehatan pada daun dengan melihat warna karena daun berkaitan terhadap kandungan *klorofil* dan sebagai pigmen yang dibutuhkan untuk proses kehidupan pada tumbuhan [1][2].

Dalam dunia teknologi jaman sekarang, *computer vision* memegang peran yang dapat dipelajari dan memahami secara visual, salah satu fitur di dalamnya yaitu dengan *image classification* [3]. Telah banyak para peneliti lain yang menggunakan metode *deep learning* untuk menganalisis data dengan sejumlah besar kasus berhasil dalam memproses gambar, mendeteksi suatu objek, pengenalan ucapan, dan masih banyak lagi [4]. Untuk identifikasi melalui media citra gambar dapat dilakukan dengan *Convolution Neural Network (CNN)*. Selain itu, *CNN* dapat digunakan untuk ekstraksi fitur *unsupervised* dan mempunyai 3 struktur lapisan dalam susunan utama yaitu *convolution layer*, *pooling layer*, dan *fully connection* [3][4][5]. Pada penelitian [3] bahwa *ResNet-50* model paling baik dalam mendeteksi jenis-jenis sampah mendapatkan nilai akurasi pada *train* 78% dan 90%, sedangkan pada *validation* sebesar 75% dan 80%. Dalam mendeteksi *Colorectal Cancer* di penelitian [6] bahwa *resnet-50* memberikan kinerja paling andal untuk nilai akurasi klasifikasi di atas 80%. Dalam mengklasifikasi penyakit pada strawberry menggunakan *ResNet-50* mencapai 100% di penelitian [7] untuk kasus pada pucuk, daun dan buah, 98% untuk jamur abu-abu dan 98% untuk jamur tepung.

Dalam penelitian ini dilakukan dengan model *CNN* dan arsitektur *ResNet-50* untuk menganalisa citra daun kopi. Penelitian ini ditujukan untuk mengetahui tingkat akurasi dalam mengidentifikasi berpenyakit atau tidak pada tanaman kopi berdasarkan citra daun dengan menggunakan metode *CNN* dan arsitektur *ResNet-50*.

## 2. METODE PENELITIAN

Identifikasi penyakit pada tanaman kopi ini, menggunakan metode penelitian yang dilakukan dalam beberapa tahapan. Tahapan dimulai dari studi literatur, pengumpulan data, perancangan, implementasi, pengujian dan analisis hasil pengujian. Untuk proses tahapan tersebut dapat dilihat pada gambar 1.



Gambar 1. Metode Penelitian

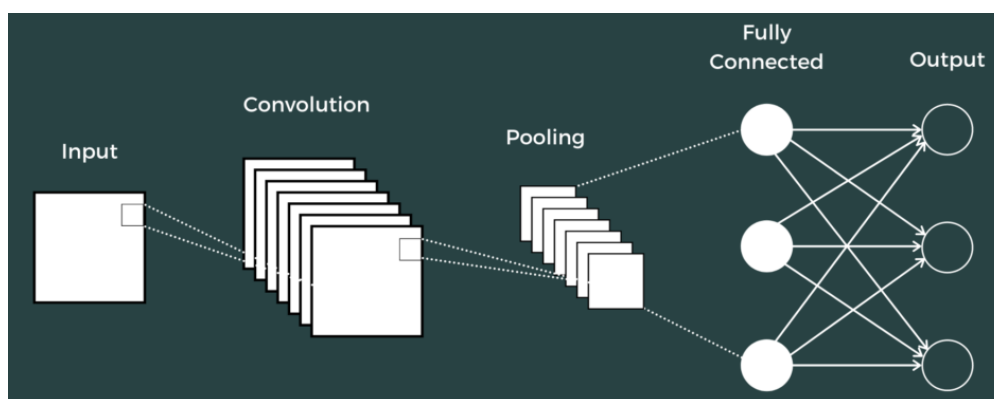
### 2.1 Studi Literatur

Tahap awal ini dilakukan dengan mencari literatur berupa buku dan jurnal yang berkaitan dengan identifikasi penyakit pada tanaman kopi melalui citra daun dengan menggunakan metode *CNN* dan arsitektur *ResNet50*.

1) Kopi merupakan sebuah jenis dari tanaman yang membutuhkan peneduh sehingga penyerapan intensitas cahaya tidak maksimal. Tanaman kopi banyak dimanfaatkan pada buahnya dan akan

diproses menjadi minuman yang sangat disukai. Setelah air dan teh, kopi menjadi minuman paling disukai masyarakat dunia [8].

2) Convolution Neural Network merupakan salah satu dari deep learning dan mempunyai struktur yang dapat mengolah data dua dimensi dan mempunyai lapisan masuk, lapisan keluar dan lapisan tersembunyi yang berisi convolutional layer, pooling layer dan fully connected layers [4]. Berikut model Convolution Neural Network dapat dilihat pada gambar 2.



Gambar 2. Model Convolution Neural Network [8]

*Convolution Layer* merupakan lapisan dan blok utama dari CNN yang digunakan dalam menghitung *output* dari *neuron* yang terhubung ke *local region* dari masukan citra [10]. *Convolution Layer* dianggap sebagai *filter* bagian *input* untuk meningkatkan pola yang diberikan. Menggunakan *filter* pada setiap *neuron* dengan cara menggeser untuk perkalian antara region kecil yang terhubung ke masukan citra dan dilanjutkan dalam melakukan operasi konvolusi pada keluaran lapisan.

*Pooling Layer* merupakan salah satu proses lapisan untuk menurunkan ukuran citra untuk memudahkan operasi pengolahan citra dan mempercepat komputasi untuk lapisan berikutnya [11]. *Pooling layer* ini bertujuan untuk mengurangi jumlah parameter dan proses komputasi, serta mencegah terjadinya akurasi tinggi pada prediksi pada data *training* namun tidak dapat memprediksi data *testing* atau dikenal dengan *overfitting*. Ada beberapa teknik untuk melakukan proses ini salah satunya yaitu mengambil nilai *maximum* dari sub citra atau disebut *Max pooling*. *Max pooling* ini dapat memecah *output* dari *relu* menjadi beberapa *grid* yang kecil lalu menggunakan nilai maksimum dari setiap *grid* untuk membangun matriks citra yang direduksi.

*Full Connected Layer* salah satu lapisan yang terdiri beberapa *node*. Jumlah Layer ini digunakan untuk menggabungkan ekstrasi fitur ke dalam kelas yang diperbarui dan di ulangi dalam dalam perkalian acak sehingga memiliki bobot dan bias untuk pendekatan kelas. Proses ini bisa membutuhkan waktu yang lama tergantung pada banyaknya fitur yang dihasilkan [10].

3) Augmentasi: Augmentasi merupakan teknik yang penting dari computer vision dalam manipulasi sebuah citra gambar sehingga mesin mampu mengenali berbagai citra gambar sekaligus menambah data secara tidak langsung [12]. Flip merupakan salah satu jenis dari augmentasi yang digunakan untuk mengubah posisi citra gambar secara vertikal ataupun horizontal. Salah satu jenis dari augmentasi lainnya yaitu *random rotation* yang berfungsi untuk merotasi sebuah citra, namun rotasi tersebut diberikan nilai derajat yang bisa kita tentukan sehingga citra gambar akan di rotasi maksimal derajat yang telah kita tentukan.

4) *ResNet-50*: *ResNet-50* merupakan salah satu arsitektur dari CNN yang memiliki konsep *shortcut connection* yang digunakan untuk mencegah sistem dari hilangnya banyak informasi di saat *training* dilakukan [5]. Arsitektur *ResNet-50* dapat di lihat pada gambar 3.

layer name	output size	18-layer	34-layer	50-layer	101-layer	152-layer
conv1	112×112	7×7, 64, stride 2				
		3×3 max pool, stride 2				
conv2.x	56×56	$\begin{bmatrix} 3 \times 3, 64 \\ 3 \times 3, 64 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 64 \\ 3 \times 3, 64 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$
conv3.x	28×28	$\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 4$	$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 4$	$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 4$	$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 8$
conv4.x	14×14	$\begin{bmatrix} 3 \times 3, 256 \\ 3 \times 3, 256 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 256 \\ 3 \times 3, 256 \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 23$	$\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 36$
conv5.x	7×7	$\begin{bmatrix} 3 \times 3, 512 \\ 3 \times 3, 512 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 512 \\ 3 \times 3, 512 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$
	1×1	average pool, 1000-d fc, softmax				
FLOPs		$1.8 \times 10^9$	$3.6 \times 10^9$	$3.8 \times 10^9$	$7.6 \times 10^9$	$11.3 \times 10^9$

Gambar 3. Arsitektur *ResNet-50*

Ada beberapa langkah dalam konvolusi diawali pada konvolusi pertama dengan ukuran 7\*7 dengan *filter* ukuran 64 kernel dengan *stride* 2 dapat memberi satu lapisan.

Pada konvolusi berikutnya ada kernel 1\*1 dengan *filter* ukuran 64, kernel 3\*3 dengan *filter* ukuran 64 dan kernel yang terakhir 1\*1 dengan *filter* ukuran 256. Ketiga lapisan tersebut diulangi sebanyak tiga kali sehingga mendapatkan 9 lapisan dan akan melihat *max pool* dengan *stride* 2.

Langkah selanjutnya ada kernel 1\*1 dengan *filter* ukuran 128, kernel 3\*3 dengan *filter* ukuran 128, dan kernel 1\*1 dengan *filter* ukuran 512. Langkah ini akan di ulangi sebanyak empat kali sehingga mendapatkan 12 lapisan.

Langkah selanjutnya kernel 1\*1 dengan *filter* ukuran 256, kernel 3\*3 dengan *filter* ukuran 256, dan 1\*1 dengan *filter* ukuran 1024. Langkah ini akan diulangi sebanyak enam kali sehingga mendapatkan 18 lapisan.

Langkah terakhir kernel 1\*1 dengan *filter* 512, kernel 3\*3 dengan *filter* 512, dan kernel 1\*1 dengan *filter* 2048. Langkah akhir ini di ulangi sebanyak tiga kali sehingga mendapatkan 9 lapisan.

Setelah itu melakukan *average pool* dan lapisan yang terhubung penuh yang berisi 1000 *node*. Dilanjutkan dengan memberi 1 lapisan pada fungsi *softmax* yang digunakan pada lapisan keluaran sehingga memiliki nilai probabilitas yang nyata. Jika lapisan yang didapatkan ditotalkan dari langkah yang telah dilakukan maka didapatkan 50 lapisan

5) *Optimizer*: *Optimizer* adalah sebuah algoritma dari salah satu fungsi yang sangat berguna untuk mempercepat proses pelatihan untuk menentukan bobot yang optimal dengan cara meminimalkan *loss* dan memaksimalkan akurasi [10]. *Optimizer* mempunyai nilai pada *learning rate* untuk menentukan performa pelatihan secara lambat dan cepat [11]. Dalam penelitian ini menggunakan *optimizer* Adam.

Adam merupakan sebuah *optimizer* dengan teknik komputasi yang efisien dan sederhana untuk memperkirakan tingkat pembelajaran adaptif untuk parameter yang terlibat dalam pelatihan *gradien* dengan kebutuhan memori yang kecil untuk optimasi *stochastic* dan tidak berubah untuk *rescalling* diagonal dari *gradien* untuk dapat menangani masalah-masalah seperti data yang banyak, gradien yang tidak memadai, dan *nonstasioner* yang membutuhkan penyetelan kecil [13]

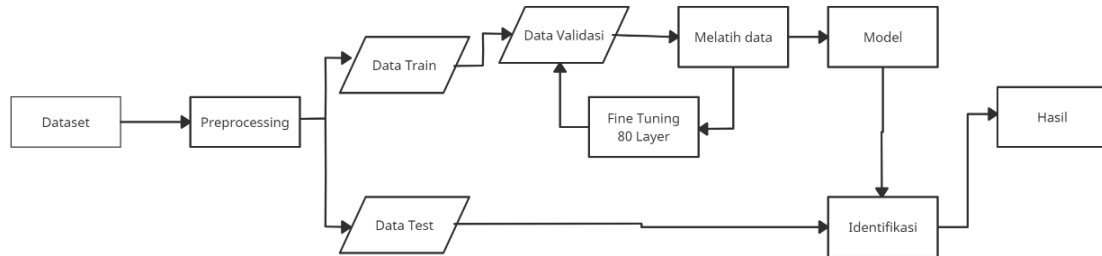
## 2. 2 Pengumpulan Data

Ditahap ini melakukan pengumpulan data berupa dataset *Coffee leaf diseases* yang di ambil dari [14], data tersebut dapat di akses atau bersifat publik dan berukuran 2048 x 1024 *pixel*. Pada dataset tersebut terdapat pembagian data berupa 400 citra daun pada data *test* dan 1264 citra daun pada data *train*, selain itu ada sebuah *file* sebagai catatan untuk pemilah dari terindikasi jenis

penyakit pada daun kopi dan daun yang masih sehat.

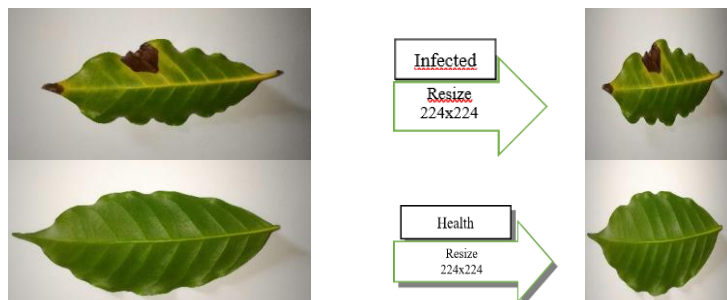
### 2. 3 Perancangan

Pada penelitian ini diperlukan sebuah rancangan yang digunakan untuk penelitian yaitu metode *CNN* dan arsitektur *ResNet-50*. Perancangan tersebut dapat dilihat di gambar 4.



Gambar 4. Perancangan Sistem

Dimulai dari pengumpulan dataset dan dilakukan *preprocessing* dengan mengubah citra gambar dari 2048 x 1024 *pixel* menjadi 224x224 *pixel* dan dilanjutkan untuk memilah citra daun berpenyakit dan yang sehat dari *file* catatan yang telah disediakan. proses tersebut pada gambar 5.



Gambar 5. *Preprocessing* citra daun

Langkah selanjutnya melakukan augmentasi dimulai dari jenis *shear\_range*, *rotation\_range*, *fill\_mode*, *zoom\_range*, *brighness\_range*, *width\_shift\_range*, *height\_shift\_range*, *horizontal\_flip*, *vertical\_flip*, dan *validation\_split* akan digunakan pada data *train* di penelitian ini. Dilanjutkan dengan melatih data *train* dan data *validasi* dengan membekukan semua *layer* sebanyak 20 *epoch*, kemudian dilanjutkan melatih data dengan melakukan *fine tuning* dan hanya 80 diawal *layer* yang akan dibekukan *layer* tersebut dengan 40 *epoch*. Model yang akan dilatih menggunakan 32 dan 16 pada *hidden layer* masing-masing diberi fungsi aktivasi *ReLU* serta diberi *dropout* 0.4 dan 2 *layer* untuk output dengan fungsi aktivasi *softmax*. Rincian pada model dengan jumlah parameter yang digunakan pada penelitian ini dapat dilihat pada tabel 1.

Tabel 1. Parameter Model *ResNet-50*

Type Layer	Output Shape	Parameter
<i>ResNet50</i>	(None,7,7,2048)	23587712
<i>flatten (Flatten)</i>	(None,2048)	0
<i>dense (Dense)</i>	(None,32)	65568
<i>dense (Dropout)</i>	(None,32)	0
<i>dense_1 (Dense)</i>	(None,16)	528
<i>dense_1 (Dropout)</i>	(None,16)	0

<i>batch_normalization</i> ( <i>BatchNormalization</i> )	(None,16)	64
<i>dense_2 (Dense)</i>	(None,2)	34

Ditahap akhir ini melakukan penyimpanan terhadap model yang sudah dilatih dengan format *file* h5 dan akan dilakukan identifikasi menggunakan model tersebut terhadap data *test* yang telah disediakan untuk mendapatkan hasil yang dibutuhkan.

#### 2. 4 Implementasi

Pada tahapan ini dilakukan mengimplementasi algoritma untuk penelitian yang telah dirancang agar sistem dapat membedakan *health* dan *infected* menggunakan data yang ada sebelumnya dan melakukan operasi perhitungan dalam bentuk kode program untuk menghitung akurasi dan *confusion matrix* pada *ResNet-50*

#### 2. 5 Pengujian

Ditahap ini dilakukan pengujian terhadap data *test* dengan program yang sudah dibuat, sistem tersebut mengumpulkan data yang diperoleh dan menerapkan hasil dari identifikasi untuk mendapatkan *accuracy*, *precision*, *recall*, dan *F1-Score*.

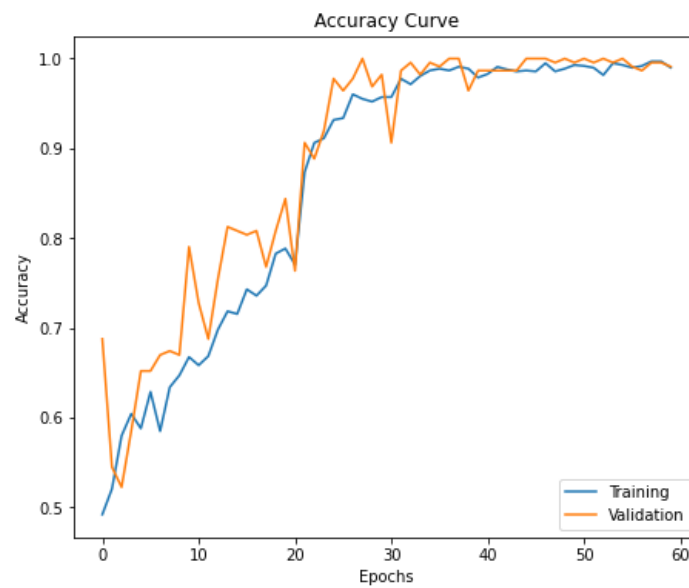
#### 2. 5 Pengujian

Pada tahap ini dilakukan analisis dan menghitung hasil yang telah di uji untuk mendapatkan tingkat keberhasilan dari metode yang digunakan dengan *Confusion Matrix* untuk menghitung nilai *precision*, *recall*, dan *F1-Score*.

*Precision* adalah jumlah total dari sampel positif terklasifikasikan sebagai benar dibagi dengan jumlah total sampel positif yang diprediksi. *Recall* didefinisikan sebagai rasio jumlah total positif yang terklasifikasikan bernilai benar dibagi dengan jumlah total positif. *Accuracy* adalah jumlah prediksi yang benar dengan jumlah total prediksi. *F1-Score* didefinisikan sebagai rata-rata *harmonic* antara *presisi* dan *Recall*.

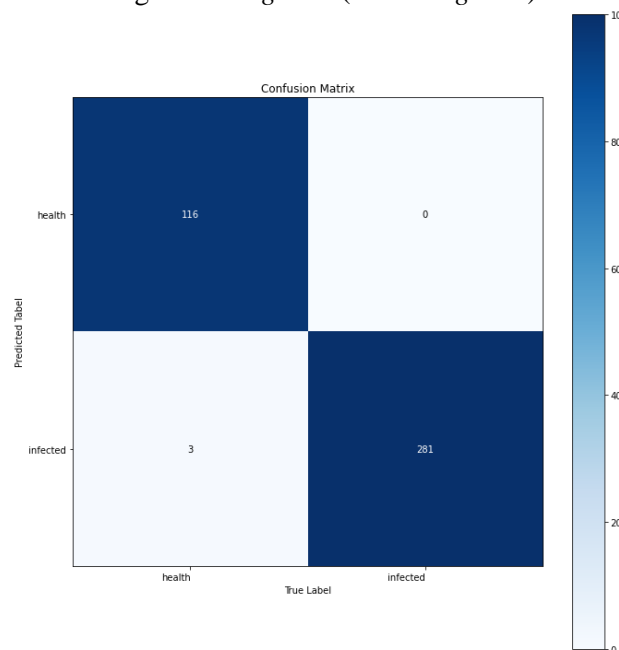
### 3. HASIL DAN PEMBAHASAN

Berdasarkan hasil *training* dengan *epoch* pertama didapatkan tingkat akurasi dengan nilai *probability* sebesar 0.4918 dan mengalami peningkatan hingga *epoch* ke 20 mencapai 0.7883 dan dilanjutkan membuat pelatihan baru dengan pembekuan 80 *layer* dengan 40 *epoch* menghasilkan di *epoch* pertama mendapatkan tingkat akurasi sebesar 0.7704 sampai ke *epoch* 40 menghasilkan tingkat akurasi 0.9899. Hasil tertinggi dari akurasi dari *epoch* 21 sampai 60 dengan nilai *probability* yaitu 0.9969 dan terendah mendapatkan nilai 0.5204. Hasil pelatihan dari *epoch* 1 dan 60 tersebut dapat dilihat pada gambar 5.



Gambar 5. Hasil Pelatihan Model

Berdasarkan hasil *testing* yang telah dilakukan dapat dilihat dengan *confusion matrix* pada gambar 6. terdapat jumlah data positif yang diprediksi benar (*True Positive*) memiliki hasil sebesar 116 gambar, jumlah data *positive* yang diprediksi salah (*False Positive*) sebesar 3, jumlah data negatif yang diprediksi salah oleh sistem (*True Negative*) memiliki hasil 281 gambar, jumlah data positif yang diprediksi sebagai data *negative* (*False Negative*) memiliki hasil 0 gambar.



Gambar 6. Hasil Test dengan *Confusion Matrix*

Adapun Hasil perhitungan dapat dilihat pada tabel 2 dengan *accuracy* mendapatkan rata-rata dengan nilai *probability* 0.99, *precision* mendapatkan rata-rata dengan nilai *probability* 0.98, *recall* mendapatkan rata-rata dengan nilai *probability* 0.99, dan *f1-score* mendapatkan rata-rata dengan nilai *probability* 0.99. Pada bagian *support*, terdapat *health* dengan 116 citra dan *infected* 284 citra. Untuk bagian *Macro AVG* 36 didapatkan dari rata-rata dari *precision* dengan nilai *probability* 0.99, *recall* dengan nilai *probability* 0.99, *F1-Score* yaitu 0.99 dan untuk hasil dari

*weighted avg* mendapatkan nilai dengan *probability* 0.99, lalu bagian xxx dapat di isi dengan *Precision*, *Recall* ataupun *F1-Score* dapat dilihat dari persamaan (1).

$$WeightedAVG = \left( \left( \left( \frac{Support_0}{TotalDataset} \right) \times 100\% \right) \times xxx_0 \right) + \left( \left( \left( \frac{Support_1}{TotalDataset} \right) \times 100\% \right) \times xxx_1 \right) \quad (1)$$

Tabel 2 Hasil Pengujian

No	Class	Precision	Recall	F1-score	Support
1	Health	0.97	1.00	0.99	116
2	Infected	1.00	0.99	0.99	284
<i>Accuracy</i>				0.99	400
<i>Macro avg</i>		0.99	0.99	0.99	400
<i>Weighted avg</i>		0.99	0.99	0.99	400

#### 4. KESIMPULAN

Berdasarkan penelitian yang telah dilakukan, maka dapat disimpulkan bahwa model *ResNet-50* dapat diterapkan untuk identifikasi penyakit daun tanaman kopi dan hasil dari pengujian pada penelitian ini menggunakan metode *CNN* dengan model arsitektur *ResNet-50* dan *optimizer* Adam mendapatkan nilai akurasi dengan rata-rata 99%.

#### 5. SARAN

Dari hasil yang telah diuji dalam penelitian ini, ada beberapa saran untuk penelitian selanjutnya dengan melakukan penelitian pada akar ataupun batang untuk mengetahui tingkat akurasi dalam mengidentifikasi penyakit pada tanaman kopi.

#### UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih yang utama kepada keluarga yang sudah membantu, dosen pembimbing yang selalu membimbing dan memberikan pembelajaran, serta teman – teman seperjuangan yang saling membantu.

#### DAFTAR PUSTAKA

- [1] Sugiarti, L. “Identifikasi hama dan penyakit pada tanaman kopi di kebun percobaan fakultas pertanian universitas winaya mukti”. *Jurnal Agro Wiralodra*, vol. 1, pp. 16-22, Juni 2017.
- [2] Dharmadewi, A. “Analisis kandungan klorofil pada beberapa jenis sayuran hijau sebagai alternatif bahan dasar food supplement”. *Jurnal Emasains: Jurnal Edukasi Matematika dan Sains*, vol. 9, pp. 171-176, September 2020.



- [3] Stephen, Raymond, & Santoso, H. “Aplikasi convolution neural network untuk mendeteksi jenis-jenis sampah. *Jurnal Manajemen Sistem Informasi dan Telematika*”, vol. 10, pp. 122-130, Oktober 2019.
- [4] Ridhovan, A., & Suharso, A. “Penerapan metode residual network (resnet) dalam klasifikasi penyakit pada daun gandum”. *JUPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)*, vol. 7, pp. 59-65, Maret 2022.
- [5] Nashrullah, F., Wibowo, S. A., & Budiman, G. “Investigasi Parameter Epoch Pada Arsitektur ResNet-50 Untuk Klasifikasi Pornografi”. *Journal of Computer, Electronic, and Telecommunication*, vol. 1, pp. 1-8, Juli 2020.
- [6] Sarwindaa, D., Paradisaa, R. H., Bustamama, A., & Anggiab, P. “Deep Learning in Image Classification using Residual Network (ResNet) Variants for Detection of Colorectal Cancer”, vol. 179, pp. 423-431, 2021.
- [7] Jia-Rong, X., Pei-Che, C., Hung-Yi, W., Quoc-Hung, P., Yeh, J.-L. A., & Hou, M.-K. “Detection of Strawberry Diseases Using a Convolutional Neural Network”. *Plants*, pp. 1-14, 2020.
- [8] Cornelis, C. M. “The Impact of Caffeine and Coffee on Human Health. *Nutrients Journal*”, vol. 11, pp. 1-4, Februari 2019.
- [9] M. Peng, C. Wang, T. Chen, G. Liu and X. Fu. “Dual temporal scale convolutional neural network for micro-expression recognition,” *Front. Psychol.*, vol. 8, p. 1–12, 2017.
- [10] Al Rivan, M. E., & Hartoyo, S. “Klasifikasi Isyarat Bahasa Indonesia Menggunakan Metode Convolutional Neural Network”. *Jurnal Teknik Informatika dan Sistem Informasi*, vol. 8, pp. 364-373, Agustus 2022.
- [11] Miranda, N. D., Novamizanti, L., & Rizal, S. “Convolutional Neural Network pada klasifikasi sidik jari menggunakan resnet-50”. *Jurnal Teknik Informatika (JUTIF)*, vol. 2, pp. 61-68, Desember 2020.
- [12] A. Setiawan, “Augmentasi Data Pada Computer Vision”, *Data Folks Indonesia*, 2021. [Online]. Available: <https://medium.com/data-folks-indonesia/augmentasi-data-pada-computer-vision-45c5ebe10e8f>. [Accessed 30 September 2022].
- [13] Yaqub, M., Feng, J., Zia, M. S., Arshid, K., Jia, K., Rehman, Z. U., & Mehmood, A. (2020). “State-of-the-Art CNN Optimizer for Brain Tumor Segmentation in Magnetic Resonance Images”. *Brain Sciences*, vol. 10, pp. 1-20, 2020.
- [14] M. Odhiambo, “Coffee leaf diseases“, *Coffee leaf diseases*, 2020. [Online]. Available: <https://www.kaggle.com/badasstechie/coffee-leaf-diseases>. [Accessed 30 september 2022]