

# Pemodelan Klasifikasi Penyakit Daun Tanaman Tomat dengan Convolutional Neural Network Algorithm

Goklas Henry Agus Panjaitan\*, Frengki Simatupang

Fakultas Vokasi, Teknologi Komputer, Institut Teknologi Del, Laguboti, Indonesia

Email: <sup>1,\*</sup>[goklas.panjaitan@del.ac.id](mailto:goklas.panjaitan@del.ac.id), <sup>2</sup>[frengki.simatupang@del.ac.id](mailto:frengki.simatupang@del.ac.id)

Email Penulis Korespondensi: [goklas.panjaitan@del.ac.id](mailto:goklas.panjaitan@del.ac.id)

**Abstrak**—Tomat merupakan tanaman hortikultura yang memiliki nilai ekonomi tinggi dan banyak ditanam oleh masyarakat Indonesia sebagai negara agraris. Produktivitas tomat mengalami peningkatan dalam tahun 2021-2022, namun dalam proses perawatan tanaman sering sekali terjadi kendala dalam mengidentifikasi jenis penyakit yang terdapat pada daun tanaman tomat karena memiliki gejala yang hampir mirip dan sulit dibedakan. Daun tanaman tomat merupakan bagian yang memiliki peran dan fungsi penting dalam proses pertumbuhan tanaman. Untuk menangani masalah penyakit daun tomat ini diperlukan sebuah metode yang bisa digunakan dengan mudah melakukan deteksi penyakit daun tomat dengan menggunakan teknologi tertentu. Sebelum hal itu dapat dilakukan diperlukan sebuah pemodelan klasifikasi tanaman daun tomat. Pemodelan ini bertujuan untuk menghasilkan model klasifikasi daun tomat yang baik sehingga dapat digunakan nantinya untuk sistem deteksi penyakit dengan teknologi IoT. Pemodelan klasifikasi penyakit tanaman tomat menggunakan algoritma Convolutional Neural Network. Tahapan pembuatan pemodelan dilakukan mulai dari analisis sumber dataset, pra-proses data, pembentukan model, pelatihan model, evaluasi model, terakhir diperoleh hasil model. Dataset yang digunakan untuk implementasi adalah dataset Tomato yang terdiri dari 18.162 gambar. Akurasi data training lebih tinggi daripada akurasi data validation yang artinya bahwa data yang digunakan untuk model adalah baik. Hasil akurasi data training adalah 94,06% yang menunjukkan bahwa data training baik digunakan sebagai model. Kemudian hasil loss function pada data training adalah 7,8% yang menunjukkan bahwa data training baik digunakan sebagai model.

**Kata Kunci:** Convolutional Neural Network; Klasifikasi Penyakit; Daun Tomat; Dataset PlantVillage; Validation Accuracy

**Abstract**—Tomatoes constitute a horticultural crop with significant economic value and are extensively cultivated by the people of Indonesia, an agricultural country. Tomato productivity is anticipated to increase in 2021-2022. However, issues often arise during the plant care process when attempting to identify the type of disease affecting the leaves of tomato plants, owing to the nearly identical symptoms that make differentiation challenging. Tomato plant leaves play a crucial role in the plant growth process. Addressing the challenge of tomato leaf diseases necessitates a method that employs specific technology for easy detection. Before implementing this method, a classification model for tomato leaf plants must be developed. The goal of this modeling is to create an effective tomato leaf classification model for subsequent use in a disease detection system that utilizes IoT technology. The modeling of tomato plant disease classification involves the application of the Convolutional Neural Network algorithm. The modeling process encompasses data source analysis, data preprocessing, model formation, model training, model evaluation, and the subsequent acquisition of model results. The dataset utilized for implementation comprises 18,162 images of tomatoes. The accuracy of the training data surpasses that of the validation data, indicating the reliability of the data used for model development. The accuracy of the training data is recorded at 94.06%, signifying its suitability for modeling. Additionally, the loss function result on the training data is 7.8%, further affirming the quality of the training data for model development.

**Keywords:** Convolutional Neural Network; Disease Classification; Tomato Leaves; Dataset PlantVillage; Validation Accuracy

## 1. PENDAHULUAN

Tomat adalah tanaman hortikultura dengan nilai ekonomi tinggi dan banyak dibudidayakan oleh masyarakat Indonesia sebagai negara agraris [1]. Tanaman Tomat (*Solanum lycopersicum* syn. *Lycopersicon esculentum*) diketahui tumbuh dengan mudah di iklim tropis Indonesia dan memiliki nilai yang tinggi secara ekonomi, menjadi bahan baku utama dalam industri makanan. Menurut Badan Pusat Statistik (BPS) Indonesia, produksi tomat di Indonesia mencapai 1,12 juta ton pada tahun 2021, menunjukkan peningkatan sebesar 0,21% dari tahun sebelumnya sebanyak 1,11 juta ton (BPS 2022). Salah satu alasan tomat tetap menjadi komoditas panen terbesar di Indonesia adalah iklim yang cocok untuk budidaya tomat [2]. Menurut data BPS tahun 2023, produktivitas tomat mengalami peningkatan antara 2021 dan 2022. Namun, dalam proses budidaya terdapat tantangan dalam mengidentifikasi penyakit daun karena gejalanya serupa dan sulit dibedakan. Daun tomat memiliki peran penting dalam pertumbuhan tanaman, dan penyakit yang mengenai daun dapat menjadi faktor signifikan yang mempengaruhi pertumbuhan tanaman [3]. Jika daun terkena penyakit, hal ini dapat membahayakan kondisi tanaman dan berdampak pada hasil panen.

Beberapa penyakit yang terjadi pada daun tanaman adalah sebagai berikut [4]: *Leaf blotch* dan *Early Blight*, *Leaf Spot*, *Target Spot*. *Leaf Disease* adalah penyakit yang terjadi pada daun tanaman yang dapat mempengaruhi kualitas suatu tanaman. Proses deteksi dini perlu dilakukan pada daun untuk mengetahui penyakit yang terjadi pada daun tanaman. Tujuan deteksi penyakit daun pada tanaman adalah [4]: untuk mengukur daerah daun yang terkena penyakit, untuk mengidentifikasi batas daerah yang terkena penyakit, untuk mengetahui warna daerah pada daun yang terserang penyakit, untuk mengetahui penyebab penyakit yang sebenarnya yang terjadi pada daun seperti disebabkan oleh serangga, karat, atau nematoda.

Identifikasi penyakit pada tanaman merupakan salah hal yang dapat mencegah hilangnya hasil dan kuantitas hasil pertanian. Secara umum, sangat sulit untuk memantau penyakit tanaman secara manual karena membutuhkan banyak pekerjaan, keahlian dalam penyakit tanaman dan membutuhkan proses yang berkelanjutan. Salah satu hal untuk

mengetahui penyakit yang terjadi pada daun tanaman adalah melalui deteksi penyakit pada daun tanaman melalui image processing.

Dalam beberapa dekade terakhir, perkembangan teknologi di bidang kecerdasan buatan, khususnya pengolahan citra, telah secara signifikan memajukan pendekatan terhadap permasalahan di berbagai bidang, termasuk pertanian. Salah satu teknologi yang dominan dalam pengolahan citra adalah Convolutional Neural Network (CNN). CNN merupakan suatu algoritma atau metode yang digunakan pada deep learning yang sangat banyak digunakan dalam pengolahan gambar [5]. CNN dipresentasikan dalam bentuk tiga dimensi atau didesain untuk mengolah data tiga dimensi yaitu dimensi dari masukan gambar (height dan weight) dan depth (kedalaman) [6].

Penggunaan CNN dalam klasifikasi penyakit tanaman tomat membuka kemungkinan pendekatan yang lebih akurat dan otomatis dalam mengidentifikasi gejala penyakit tanaman tersebut. Pentingnya pemodelan klasifikasi penyakit tanaman tomat terletak pada dampak positifnya terhadap produktivitas pertanian dan kesejahteraan sosial. Penyakit tanaman tomat dapat menyebabkan penurunan hasil, bahkan dalam beberapa kasus kerugian total. Dengan sistem yang dapat mendeteksi penyakit tanaman tomat dengan cepat dan akurat, petani dapat segera mengambil tindakan pencegahan yang diperlukan, seperti memastikan pengendalian yang tepat dengan pestisida atau mengisolasi tanaman yang terinfeksi untuk mencegah penyebaran penyakit ke tanaman lain.

Keuntungan utama penggunaan CNN untuk pemodelan klasifikasi penyakit tanaman tomat terletak pada kemampuannya untuk secara otomatis mengekstraksi fitur-fitur penting dari gambar tanaman. CNN dapat mempelajari fitur-fitur hierarkis ini melalui konvolusi dan penggabungan lapisan, serupa dengan cara manusia memproses informasi visual. Sebagai contoh, CNN dapat mendeteksi pola tekstur, warna, dan bentuk yang mungkin sulit diidentifikasi menggunakan metode tradisional. CNN telah menjadi algoritma pilihan dalam menyelesaikan masalah klasifikasi gambar. Beberapa penelitian telah menggunakan Convolutional Neural Networks (CNN) dalam berbagai kasus klasifikasi gambar, termasuk klasifikasi citra CT scan kanker paru-paru menggunakan Contrast Stretching pada CNN dengan EfficientNet-B0 [7], klasifikasi citra sinar-X pneumonia dan non-pneumonia [8][9][10], klasifikasi gambar daging sapi dan babi [11], [12][13], klasifikasi citra penyakit daun [14], klasifikasi citra stroke [15], dan klasifikasi serangan hama pada pertanian [16]. Oleh karena itu, CNN sangat cocok untuk pengolahan pola citra [3].

Beberapa penelitian telah membahas klasifikasi penyakit tanaman tomat dengan menggunakan metode CNN. Dalam sebuah penelitian [17], algoritma CNN berhasil mengklasifikasikan dataset daun tomat dengan 10.000 gambar, mencapai akurasi sebesar 94,1%, sedangkan ResNet 50 menghasilkan 88,4%. Penelitian lain [18] menganalisis daun tomat yang terkena penyakit dengan menggunakan algoritma CNN, mencapai akurasi sebesar 99,64%. Penelitian [19] menggunakan kecerdasan buatan untuk mengklasifikasikan penyakit tanaman tomat, mencapai akurasi pelatihan sebesar 98% dan akurasi validasi sebesar 82% berdasarkan dataset dengan 10 kelas penyakit. Penelitian lain [20] berfokus pada identifikasi penyakit Leaf Mold pada daun tomat menggunakan model DENSENET121 berbasis transfer learning. Dalam penelitian lainnya [21], enam model CNN digunakan untuk mengklasifikasikan gambar daun tomat yang sakit, dan pendekatan ensemble CNN digunakan. Hasilnya menunjukkan bahwa Ensemble CNN mencapai akurasi sekitar 90,83%, dengan nilai presisi dan recall rata-rata di atas 0,8 hingga 1,0. Secara keseluruhan, metode ensemble CNN terbukti lebih baik daripada satu model CNN saja. Terakhir, sebuah penelitian [1] bertujuan untuk mengidentifikasi penyakit pada tanaman tomat melalui citra daun menggunakan deep learning dengan metode Convolutional Neural Network. Dua arsitektur CNN, LeNet-5 standar dan LeNet-5 yang dimodifikasi, digunakan untuk deteksi penyakit. LeNet-5 standar berhasil mendeteksi penyakit pada daun tomat dengan akurasi 90%, sedangkan LeNet-5 yang dimodifikasi, dengan penyesuaian lapisan jaringan dan parameter, mencapai akurasi optimal sebesar 95%.

Penelitian [3] menggunakan metode SVM (Support Vector Machine) dan CNN untuk menentukan penyakit yang memengaruhi daun tomat. Pengujian dilakukan dengan 200 sampel citra daun tomat, dengan 160 citra sebagai data latih dan 40 citra sebagai data uji. Hasil pengujian menunjukkan bahwa metode CNN memiliki persentase akurasi rata-rata sebesar 97,5%, presisi sebesar 95,45%, recall sebesar 95%, dan tingkat kesalahan sebesar 5%. Sementara itu, SVM menghasilkan akurasi rata-rata sebesar 95%, presisi sebesar 90,83%, recall sebesar 90%, dan tingkat kesalahan sebesar 10%. Berdasarkan hasil pengujian, dapat dikatakan bahwa dalam penelitian ini, CNN adalah klasifikasi yang lebih baik dibandingkan SVM.

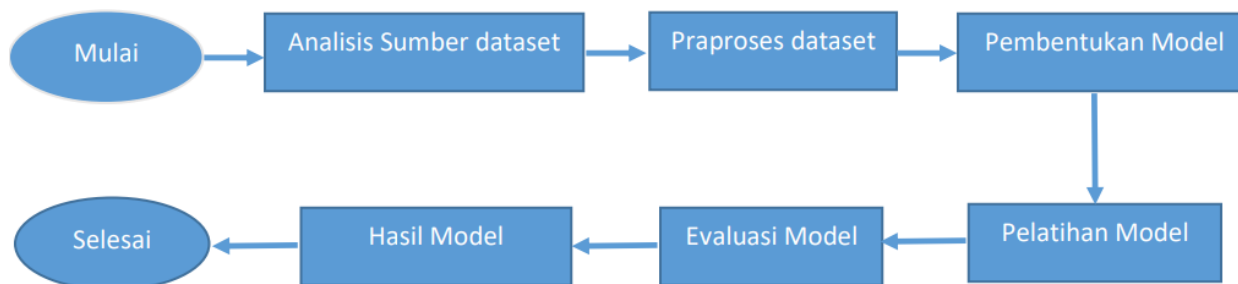
Dalam penelitian berjudul "Depthwise separable convolution architectures for plant disease classification" [22], penyakit tanaman diklasifikasikan berdasarkan gambar daun menggunakan algoritma CNN dengan model arsitektur konvolusi yang terpisah. Para peneliti memperkenalkan arsitektur Modified MobileNet, membandingkannya dengan jaringan AlexNet dan VGG. Proses pelatihan dan pengujian dilakukan pada dataset PlantVillage yang tersedia secara umum, terdiri dari 82.161 gambar dengan 55 kelas penyakit tanaman dan kondisi sehat. Arsitektur jaringan Reduced MobileNet mencapai akurasi klasifikasi sebesar 98,34% dengan parameter 29 kali lebih sedikit dibandingkan VGG dan 6 kali lebih rendah dari MobileNet.

Penelitian lain [2] berfokus pada klasifikasi kesehatan daun tanaman tomat, menguji berbagai model untuk mengidentifikasi tanaman tomat yang terkena penyakit. Metode yang diusulkan menggunakan pendekatan CNN dengan arsitektur VGG, MobileNet, dan Inception V3. Data citra berasal dari dataset Plant Disease Classification Merged, yang berisi berbagai kategori gambar yang digunakan dalam eksperimen. Hasil eksperimen menunjukkan bahwa masing-masing model mencapai kinerja akurasi sebesar 98%, 93%, dan 88% untuk Inception V3, VGG, dan MobileNet. Temuan menunjukkan bahwa model dengan kinerja terbaik dalam memproses data adalah Inception V3, diikuti oleh VGG dan MobileNet. Meskipun demikian, MobileNet tetap efektif dan efisien dalam menjalankan model jauh lebih baik daripada Inception V3 dan VGG.

Pada penelitian ini penulis akan lebih fokus untuk membuat model klasifikasi untuk penyakit tanaman tomat dengan menggunakan metode CNN, karena berdasarkan penelitian sebelumnya bahwa metode CNN ini cocok dan baik digunakan untuk pembentukan model berupa citra. Tujuan dari pembentukan model klasifikasi penyakit tanaman tomat ini akan digunakan untuk penelitian berikutnya untuk sistem deteksi penyakit tanaman tomat berbasis teknologi IoT yang sangat kompleks. Dalam penelitian ini penulis juga akan menggunakan dataset pengujian yang cukup besar yaitu 18.162 gambar.

## 2. METODOLOGI PENELITIAN

### 2.1 Tahapan Penelitian



**Gambar 1.** Tahapan Penelitian

Dalam melakukan penelitian ini penulis mencoba membuat tahapan penelitian untuk melakukan pembuatan pemodelan untuk klasifikasi tanaman Tomat seperti pada Gambar 1 diatas. Pembentukan model bertujuan untuk membentuk sebuah mode dan label yang digunakan untuk pengujian data pada aplikasi berkelanjutan. Proses untuk pembentukan model yang siap untuk digunakan ini dapat dilakukan mulai dari analisis sumber dataset, praproses dataset, pembentukan model, pelatihan model, evaluasi model, sampai model diperoleh. Model yang sudah dibentuk ini pada akhirnya dapat digunakan sebagai pemanfaatan untuk pengembangan pada aplikasi berikutnya.

### 2.2 Analisis Sumber Data

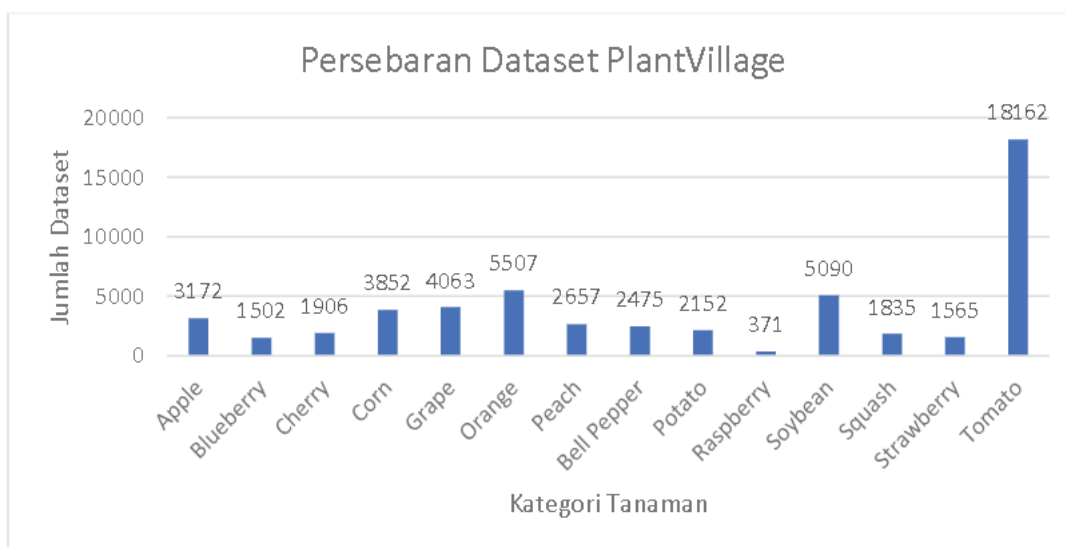
Pada penelitian ini akan menggunakan *open dataset PlantVillage* [23] yang diambil dari *repository spMohanty* (<https://github.com/spMohanty/PlantVillage-Dataset>). Adapun pemilihan *dataset PlantVillage* ini karena menyajikan data yang sesuai dengan kebutuhan penelitian ini. Selain itu, *dataset PlantVillage* juga sudah digunakan pada banyak penelitian. *Dataset PlantVillage* terdiri dari 54.303 gambar daun tanaman yang sehat dan yang tidak sehat yang dibagi menjadi 38 klasifikasi berdasarkan spesies tanaman dan penyakit yang dapat dilihat pada Tabel 1 berikut.

**Tabel 1.** Detail Klasifikasi *Dataset PlantVillage*

No	Jenis Tanaman	Jumlah Klasifikasi	Nama Klasifikasi
1	<i>Apple</i>	4	<i>Apple Apple scab</i> <i>Apple Black rot</i> <i>Apple Cedar appl rust</i> <i>Apple healthy</i>
2	<i>Blueberry</i>	1	<i>Blueberry healthy</i>
3	<i>Cherry</i>	2	<i>Cherry Healthy</i> <i>Cherry Powdery_mildew</i>
4	<i>Corn</i>	4	<i>Corn cercospora_leaf_spot</i> <i>Gray_leaf_spot</i> <i>Corn Common_rust</i> <i>Corn healthy</i> <i>Corn Northern_Leaf_Blight</i>
5	<i>Grape</i>	4	<i>Grape Black_rot</i> <i>Grape Esca_(Black_Measles)</i> <i>Grape Healthy</i> <i>Grape Leaf Blight</i>
6	<i>Orange</i>	1	<i>Orange Haunglongbing</i>
7	<i>Peach</i>	2	<i>Peach Bacterial spot</i> <i>Peach Healthy</i>
8	<i>Pepper</i>	2	<i>Pepper Bell Bacterial_spot</i> <i>Pepper Bell healthy</i>
9	<i>Potato</i>	3	<i>Potato Early blight</i> <i>Potato healthy</i> <i>Potato Late blight</i>

10	<i>Raspberry</i>	1	<i>Raspberry healthy</i>
11	<i>Soybean</i>	1	<i>Soybean healthy</i>
12	<i>Squash</i>	1	<i>Squash Powdery mildew</i>
13	<i>Strawberry</i>	2	<i>Strawberry Healthy</i> <i>Strawberry leaf scorch</i>
14	<i>Tomato</i>	10	<i>Tomato Bacterial spot</i> <i>Tomato Early blight</i> <i>Tomato healthy</i> <i>Tomato Late_blight</i> <i>Tomato Leaf_mold</i> <i>Tomato Septoria_leaf_spot</i> <i>Tomato Spider mites Two spotted spider mite</i> <i>Tomato Target Spot</i> <i>Tomato Tomato mosaic virus</i> <i>Tomato Tomato Yellow Leaf Curl Virus</i>

Tabel 1 diatas menunjukkan bahwa klasifikasi paling banyak terdapat pada tanaman tomat yaitu sebanyak 10 klasifikasi dan klasifikasi paling sedikit adalah tanaman *Blueberry*, *Orange*, *Raspberry*, *Soybean* dan *Squash* yaitu hanya ada 1 klasifikasi. *Dataset PlantVillage* menunjukkan persebaran dataset pada setiap tanaman yang dapat kita lihat dari gambar 2 berikut.



**Gambar 2.** Persebaran dataset PlantVillage

Pada gambar 2 tersebut menunjukkan bahwa jumlah dataset terbanyak terdapat pada tanaman tomat dan jumlah dataset paling kecil adalah *raspberry*. Pada penelitian ini penulis akan menggunakan dataset khusus pada tanaman tomat, karena pada dataset tanaman tomat yang paling banyak memiliki klasifikasi dan memiliki jumlah dataset terbanyak yaitu sebanyak 18.162 data. Jadi, dataset yang digunakan tidak keseluruhan dataset, tetapi hanya fokus pada satu jenis tanaman saja untuk memudahkan proses pengujian.

### 2.3 Praproses Dataset

Pada tahapan praproses dataset adalah mempersiapkan data dalam penelitian ini. Berdasarkan bahasan sebelumnya tentang analisis sumber data, data yang akan digunakan adalah *dataset PlantVillage*. Secara lebih rinci, yang dilakukan dalam tahap praproses yaitu melakukan seleksi dataset dan menentukan data yang digunakan untuk pelatihan dan pengujian yang dijelaskan berikut ini.

#### a. Melakukan Seleksi Dataset

Seleksi dataset dilakukan karena data asli dari *PlantVillage* terlalu besar untuk diproses dengan spesifikasi *device* yang terbatas. Data tersebut berjumlah 54.303 gambar. Selain itu, pada penelitian ini, hanya fokus pada satu jenis tanaman untuk memudahkan proses pengujian dan memilih dataset pada tanaman tomat. Alasan pemilihan dataset tomat, karena pada dataset tanaman tomat yang paling banyak memiliki klasifikasi (10 klasifikasi) dibandingkan tanaman yang lain dan memiliki jumlah dataset terbanyak yaitu sebanyak 18.162 data.

#### b. Menentukan Data *Training* dan Data *Testing*

Setelah dilakukan seleksi dataset pada tahapan sebelumnya, maka selanjutnya akan dilakukan penentuan data terhadap data *training* dan data *testing*. Adapun pemilihan data *training* dan data *testing* tidak dilakukan secara acak,

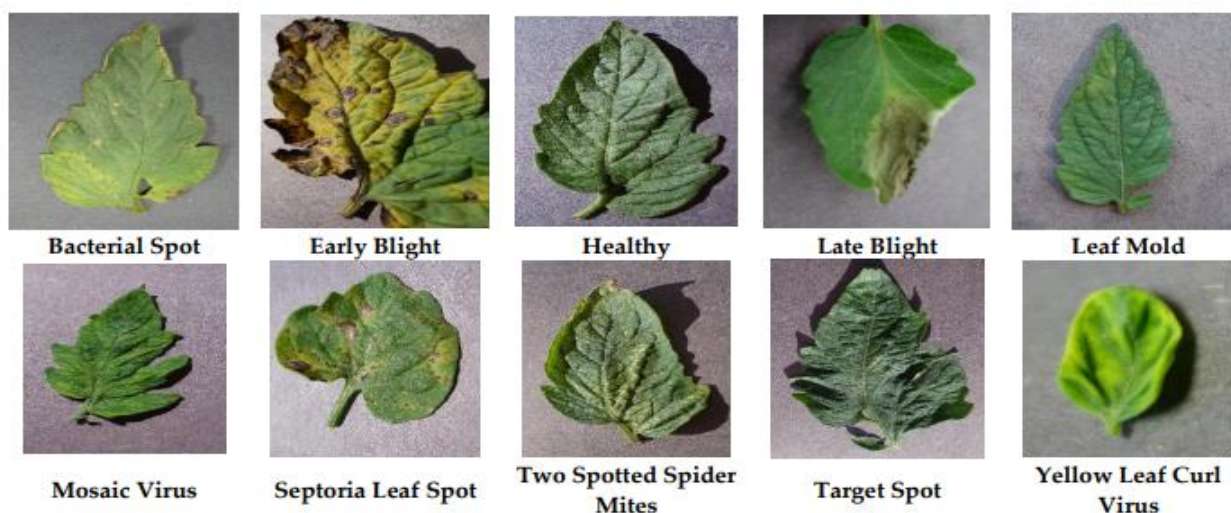
tetapi pada *dataset PlantVillage* sudah dilakukan sebelumnya pembagian data untuk data *training* dan data *testing* yang dapat dilihat pada Tabel 2 berikut.

**Tabel 2.** Data *Training* dan Data *Testing*

<i>Dataset Clasification</i>	Data <i>Training</i>	Data <i>Testing</i>
<i>Tomato Bacterial Spot</i>	1702	425
<i>Tomato Early Blight</i>	800	200
<i>Tomato Late Blight</i>	1527	382
<i>Tomato Leaf Mold</i>	761	191
<i>Tomato Septoria Leaf Spot</i>	1417	354
<i>Tomato Spider Mites</i>	1341	335
<i>Tomato Target Spot</i>	1123	281
<i>Tomato Mosaic Virus</i>	299	74
<i>Tomato Yellow Leaf Curl Virus</i>	4286	1071
<i>Tomato Healthy</i>	1273	318

Pada Tabel 2 menunjukkan bahwa dataset tomat paling banyak adalah dengan klasifikasi *Tomato\_Yellow\_Leaf\_Curl\_Virus* dan dataset paling sedikit adalah *Tomato\_Mosaic\_Virus*.

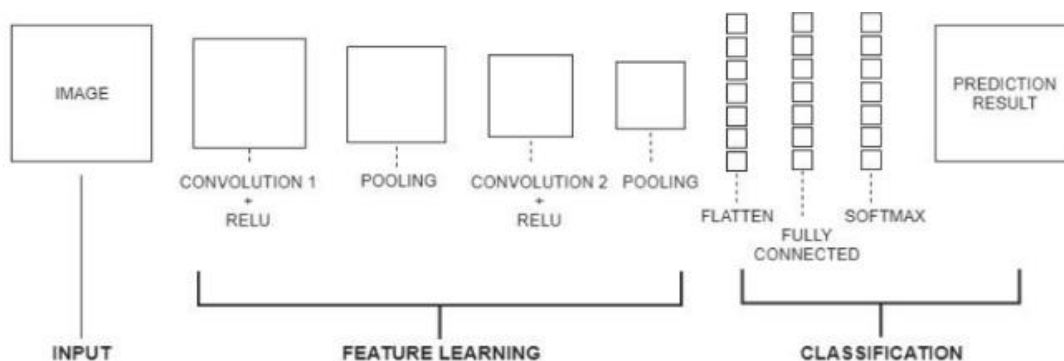
Pada gambar 3 merupakan gambar klasifikasi penyakit daun tomat [24] yang terdiri dari 10 jenis penyakit daun tomat yang akan menjadi data *training* dan data *testing*.



**Gambar 3.** Klasifikasi penyakit daun tomat

## 2.4 Pembentukan Model

Pada tahapan ini adalah tahapan untuk pembentukan model dengan menerapkan algoritma CNN dan *library Tensorflow*. Model yang terbentuk akan digunakan untuk proses prediksi pada pengujian. Sebelum melakukan pembentukan model dengan menggunakan algoritma CNN, terlebih dahulu akan mendefinisikan label pada dataset untuk menentukan klasifikasi penyakit tanaman. Setelah mendefinisikan label, kemudian menerapkan algoritma CNN untuk pembentukan model dengan menggunakan library yang terdapat pada *Tensorflow*. Keluaran dari tahapan pembentukan model adalah model dengan ekstensi *.h5* yang digunakan selanjutnya untuk proses pelatihan. Secara umum CNN terdiri dari dua lapisan umum yaitu lapisan *feature learning* dan lapisan *classification* seperti yang ditunjukkan pada gambar 4 dibawah [25].



**Gambar 4.** Arsitektur CNN

Setiap bagian dari lapisan CNN dijelaskan berikut ini.

a. Feature Learning

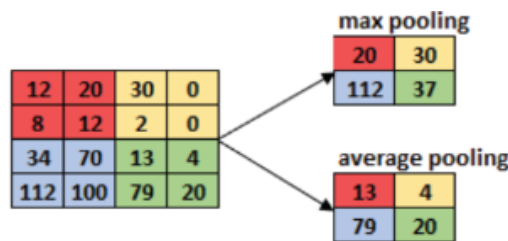
Lapisan-lapisan dalam Feature Learning bertujuan untuk menerjemahkan atau mengubah suatu masukan menjadi fitur berdasarkan karakteristik dari masukan tersebut dalam format vektor angka. Lapisan Feature Learning ini dibagi menjadi dua bagian: lapisan konvolusional dan lapisan pooling.

1. Lapisan Konvolusional

Lapisan ini terdiri atas dua bagian: lapisan konvolusional dan Rectified Linear Unit (ReLU). Lapisan konvolusional menghitung keluaran dari neuron yang terhubung ke wilayah sekitar masukan. Setiap lapisan yang terhubung melakukan perhitungan dot antara nilai dan daerah lokal yang merupakan bagian kecil yang terhubung ke dalam nilai isi masukan gambar. Sementara itu, ReLU menghilangkan gradien yang menghilang dengan mengimplementasikan elemen pada fungsi aktivasi, yaitu  $f(x) = \max(0, x)$ , yang berarti elemen pada aktivasi akan dilakukan ketika posisi elemen mencapai batas 0.

2. Lapisan Pooling

Lapisan pooling adalah lapisan yang mengurangi dimensi dari peta fitur sehingga proses komputasi semakin dipercepat karena nilai yang perlu diubah semakin sedikit, dan memberikan solusi terhadap overfitting (keadaan di mana data dilatih untuk mendapatkan data terbaik). Umumnya, metode pooling yang paling sering diimplementasikan adalah average pooling atau max pooling. Max pooling bertugas mencari nilai maksimum pada setiap pergerakan filter, sementara average pooling menghitung nilai rata-rata dari setiap filter, seperti yang ditunjukkan pada Gambar 5.



Gambar 5. Max Pooling dan Average Pooling

b. Classification

Pada lapisan *classification* bertujuan untuk melakukan klasifikasi pada setiap neuron yang telah melewati proses ekstraksi fitur pada tahapan sebelumnya. Lapisan ini dibagi atas 3 bagian yaitu *flatten*, *fully-connected* dan *softmax* yang dijelaskan sebagai berikut.

1. Flatten

Flatten merupakan bagian yang membentuk kembali feature map menjadi suatu vektor agar dapat digunakan sebagai masukan dari lapisan *fully-connected*.

2. Fully-connected

Lapisan *fully-connected* merupakan lapisan yang akan menghitung skor kelas. Masing-masing *neuron* akan melakukan koneksi terhadap keseluruhan nilai di dalam isi gambar.

3. Softmax

*Softmax* bertujuan untuk melakukan kalkulasi probabilitas dari setiap objek kelas terhadap semua kelas yang berfungsi untuk melakukan klasifikasi objek kelas dengan nilai keluaran probabilitas diantara nilai 0 hingga 1, dan jumlah semua probabilitas akan sama dengan satu. *Softmax* adalah lapisan terakhir pada jaringan CNN yang menghitung nilai skor probabilitas actual setiap label kelas.

## 2.5 Pelatihan Model

Pada tahapan ini adalah tahapan untuk melakukan pelatihan pada model yang telah terbentuk untuk menguji keakuratan model tersebut dalam proses prediksi. Pelatihan pada model yang terbentuk dengan menerapkan algoritma CNN menggunakan jumlah epochs dan *batch size* untuk menyimpan hasil pelatihan model dengan metrik akurasi yang terbaik setiap iterasinya. *Optimizer* yang digunakan untuk perbaikan model adalah dengan mengadopsi metode *Adam Optimizer*. Pada tahapan ini, keluaran yang dihasilkan adalah nilai *loss (loss function)* dan akurasi (*accuracy*) yang bertujuan untuk mendapatkan nilai akurasi dari setiap model yang telah melewati pelatihan.

## 2.6 Evaluasi Model

Pada tahapan ini adalah bagian dari tahapan sebelumnya yaitu pelatihan model. Keluaran dari pelatihan model adalah nilai *loss (loss function)* dan akurasi (*accuracy*) yang akan dianalisis untuk proses evaluasi. Evaluasi dilakukan dengan membandingkan *accuracy* dan *loss* yang ditunjukkan pada grafik untuk melihat lebih jelas hasil perbandingan *accuracy* dan *loss* dari model yang telah dilatih. Jika hasil dari pelatihan masih menunjukkan akurasi dibawah 90%, maka selanjutnya akan dilakukan kembali proses pelatihan model sampai model mendapatkan akurasi diatas 90%.

### 3. HASIL DAN PEMBAHASAN

Pembentukan model pelatihan yang digunakan untuk prediksi pada data pengujian diterapkan menggunakan algoritma CNN. Dalam penelitian ini penulis menerapkan CNN untuk memodelkan klasifikasi penyakit daun tanaman tomat. Tahapan-tahapan untuk pembentukan model dapat dilihat pada Gambar 1. Arsitektur CNN tersebut adalah lapisan konvolusi, lapisan *pooling*, dan lapisan *fully connected* untuk mengekstraksi fitur citra daun tomat yang terinfeksi penyakit. Proses pelatihan dataset dengan menggunakan dataset 18.162 gambar. Evaluasi kinerja model akan diuji menggunakan metrik akurasi. Berikut penjelasan implementasi metode CNN dalam melakukan pemodelan klasifikasi penyakit daun tomat mulai dari Praproses *dataset*, Pembentukan *model* dengan CNN, Pelatihan *model* dan Evaluasi *model*.

#### 3.1 Praproses Dataset

Pada implementasi praproses dataset, dataset yang akan diimplementasikan adalah dataset tomat yang terdiri dari 18.162 gambar seperti dijelaskan pada analisis sumber data. Dataset gambar telah disimpan sesuai dengan klasifikasi penyakit pada tanaman tomat. Klasifikasi pada tanaman tomat terdapat 10 klasifikasi, sehingga terbentuk 10 folder dataset. Praproses dataset yang diimplementasikan adalah gambar akan di-load dan diubah dalam bentuk *image array* dan mendefinisikan label berdasarkan klasifikasi pada tanaman tomat. Implementasi praproses dataset menggunakan python, *library open-cv* dan *mumpy*. Implementasi untuk *me-load dataset* dilakukan dengan konversi gambar menjadi *image array* serta mendefinisikan label klasifikasi sesuai dengan nama folder dataset. Pertama, program akan mendefinisikan list folder pada dataset yang disimpan. Kemudian setiap list folder akan berisi dataset gambar yang digunakan untuk pelatihan. Setiap gambar yang terdapat pada *list folder* akan disimpan dalam *array* daftar gambar (*image list*). *Array image list* yang berisi data gambar dalam ekstensi *.jpg* atau *.JPG* akan dikonversi menjadi *image array* dengan menggunakan *library open-cv*. Kemudian label klasifikasi disimpan dalam *array* label yang diambil dari penamaan folder (penamaan folder telah diproses sebelumnya sesuai data klasifikasi pada dataset *PlantVillage*).

#### 3.2 Pembentukan Model

Setelah melakukan implementasi praproses data, yang melibatkan memuat seluruh dataset yang digunakan untuk pelatihan dan mengkonversi semua gambar ke dalam bentuk larik gambar, langkah berikutnya adalah pembentukan model dengan menerapkan algoritma CNN berdasarkan arsitektur CNN sebagaimana dijelaskan dalam Gambar 4. Ini diimplementasikan menggunakan Python dan perpustakaan TensorFlow. Implementasi arsitektur CNN dalam pembangunan model yang digunakan dalam pelatihan. Pertama, proses yang dilakukan adalah menentukan parameter-parameter di dalam algoritma CNN. Masukan gambar pada algoritma CNN ini adalah gambar dengan ukuran 256 x 256. Kemudian, untuk batch size dengan ukuran 32. Batch size adalah jumlah sampel yang akan dikerjakan dalam memperbaharui parameter model dalam arsitektur neural network. Kemudian terdapat jumlah filter yang menjadi masukan dalam proses konvolusi dengan 3 proses konvolusi dan 3 proses pooling layer. Tahapan pertama pada lapisan konvolusi memiliki jumlah filter 32, pada tahapan kedua lapisan konvolusi menggunakan jumlah filter 64 dan konvolusi ketiga digunakan jumlah filter sebanyak 128. Pada setiap proses tahapan konvolusi akan memanfaatkan fungsi aktivasi RELU. Fungsi aktivasi RELU digunakan untuk melakukan perubahan nilai yang minus hasil dari proses konvolusi yang ditunjukkan melalui matriks. Fungsi aktivasi ini melakukan "*threshol*" dari 0 sampai tak terbatas. Setelah itu pada tahapan konvolusi akan menggunakan *zero padding (same)*. *Zero padding* merupakan nilai ukur piksel yang memiliki nilai 0 yang dijumlahkan pada setiap sisi masukan. Setelah itu, kernel yang digunakan adalah dengan ukuran 3x3. Kernel merupakan suatu matriks untuk melakukan perhitungan dan proses deteksi terhadap pola yang didapatkan pada tahapan *convolution*. Untuk ukuran pooling-nya adalah 2. Pooling adalah lapisan yang melakukan pengurangan dimensi pada feature map. Pembuatan layer dropout yang digunakan adalah dengan nilai 0.25 pada setiap proses konvolusi dan 0.5 ketika proses konvolusi berakhir. Setelah melewati tahapan konvolusi, nilai pooling akan di-update menjadi vektor dua dimensi. Pada lapisan dense memiliki nilai vektor 1024 yang mengimplementasikan fungsi aktifasi RELU dan *softmax* yang bertujuan untuk melakukan klasifikasi kelas.

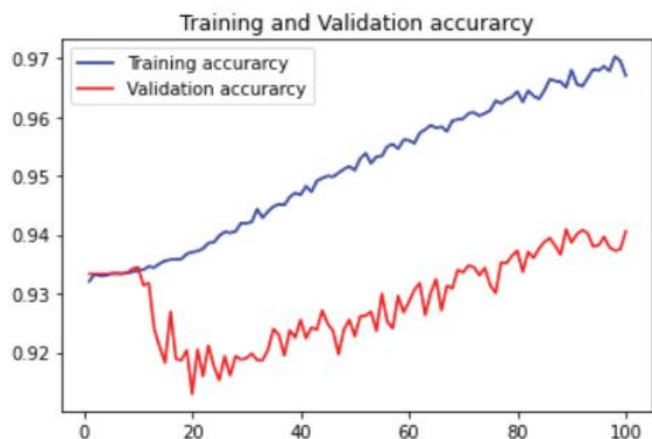
#### 3.3 Pelatihan Model

Setelah melakukan implementasi pembentukan model, selanjutnya adalah model yang telah terbentuk akan dilatih dengan menggunakan *Adam optimizer*. Implementasi pelatihan model tentang pelatihan *model* dengan menggunakan *library Tensorflow*. Pertama sebelum melakukan pelatihan model, proses yang dilakukan adalah proses augmentasi gambar yaitu proses untuk pembangkitan data gambar sebelum melakukan pelatihan. Proses augmentasi gambar yang dilakukan adalah proses rotasi gambar dan *flipping* gambar (membalikkan) gambar secara horizontal. Proses pada tahapan kedua adalah memisahkan data menjadi data training dan data testing untuk tujuan validasi. Pemisahan data dilakukan secara *random*. Data yang digunakan diambil dari *image list* dan *label list* pada saat pertama kali melakukan praproses dataset. Dalam proses data training akan menggunakan ukuran pengujian 0.2 dan nilai random state 42. Ketiga, proses yang dilakukan adalah melatih model dengan menggunakan metode Adam optimizer untuk menghitung nilai *loss* dan *accuracy*. *Model fit generator* bertujuan untuk melakukan proses pembangkitan data dari data training dan data validation pada setiap nilai *epoch* sampai nilai semua *epoch* mencapai jumlah sampel. Nilai batch size adalah 32 artinya pada proses data training akan mengambil data secara random sebesar 32 data dari keseluruhan sampel dataset pada setiap *epoch* sampai keseluruhan *epoch* mencapai batas maksimum sampel. Penentuan nilai dari *epoch* pada pelatihan data adalah 25 yang

artinya proses pengulangan dari pelatihan data pada setiap sesi melewati 25 kali untuk menghitung error yang paling minimum dan mendapatkan performansi yang baik dari model yang dibangun.

### 3.4 Analisis Evaluasi Model

Untuk mengevaluasi hasil statistik dalam memprediksi sebuah model dan menghitung akurasi model prediksi yang diimplementasikan, metode K-Fold Cross Validation digunakan. Berdasarkan implementasi K-Fold Cross Validation pada data pelatihan menggunakan dataset Plain Village, akurasi data pelatihan dan data validasi diperoleh seperti yang ditunjukkan pada Gambar 10. Gambar 10 menjelaskan bahwa akurasi data pelatihan lebih tinggi daripada akurasi data validasi, menunjukkan kesesuaian data yang digunakan untuk model. Akurasi data pelatihan adalah 94,06%, menunjukkan bahwa data pelatihan cocok digunakan sebagai model dan lebih baik dari model yang digunakan dalam penelitian sebelumnya.



Gambar 6. Training dan Validation Accuracy



Score for fold 1: loss of 7.801330707601686; accuracy of 94.05527710914612%

Gambar 7. Training dan Validation Loss

Pada Gambar 7 menjelaskan bahwa *loss function* data *training* lebih rendah daripada *loss function* data *validation* yang artinya bahwa data yang digunakan untuk model memiliki performa yang lebih baik, tetapi model tidak dapat menggeneralisasi dengan benar pada data yang tidak diketahui. Hasil *loss function* pada data training adalah 7.8% yang menunjukkan bahwa data *training* baik digunakan sebagai *model*.

## 4. KESIMPULAN

Dalam penelitian ini, penulis mengembangkan model klasifikasi penyakit tanaman menggunakan algoritma Convolutional Neural Network (CNN). Proses pembuatan model mencakup langkah-langkah mulai dari analisis sumber dataset, pra-pemrosesan data, pembentukan model, pelatihan model, evaluasi model, hingga akhirnya memperoleh hasil model. Dataset yang diimplementasikan dalam penelitian ini adalah dataset Tomato, terdiri dari 18.162 gambar, dengan 10 klasifikasi untuk tanaman Tomat. Pelatihan model dilakukan menggunakan perpustakaan Tensorflow. Selama proses pelatihan, digunakan ukuran pengujian sebesar 0,2 dan nilai random state sebesar 42. Penulis melatih model menggunakan metode optimizer Adam untuk menghitung nilai loss dan akurasi. Untuk menilai hasil statistik dalam memprediksi model dan menghitung akurasi model prediksi, digunakan metode K-Fold Cross Validation. Berdasarkan hasil implementasi K-

Fold Cross Validation pada data pelatihan menggunakan dataset Plain Village, diperoleh akurasi data pelatihan dan akurasi data validasi. Akurasi data pelatihan lebih tinggi daripada akurasi data validasi, menunjukkan kesesuaian data yang digunakan untuk model. Akurasi data pelatihan sebesar 94,06%, menunjukkan bahwa data pelatihan cocok digunakan sebagai model. Selain itu, hasil fungsi loss pada data pelatihan sebesar 7,8%, menunjukkan bahwa data pelatihan cocok digunakan sebagai model. Model ini dapat digunakan lebih lanjut untuk penelitian dalam sistem berbasis IoT untuk deteksi penyakit daun pada tanaman tomat

## REFERENCES

- [1] R. C. Sigitta, R. H. Saputra, and F. Fathulloh, "Deteksi Penyakit Tomat melalui Citra Daun menggunakan Metode Convolutional Neural Network," *Avitec*, vol. 5, no. 1, p. 43, 2023, doi: 10.28989/avitec.v5i1.1404.
- [2] P. Palupiningsih, A. R. Sujiwanto, and R. R. B. P. Prawirodirjo, "Analisis Perbandingan Performa Model Klasifikasi Kesehatan Daun Tomat menggunakan arsitektur VGG, MobileNet, dan Inception V3," *J. Ilmu Komput. dan Agri-Informatika*, vol. 10, no. 1, pp. 98–110, 2023, doi: 10.29244/jika.10.1.98-110.
- [3] F. Felix, S. Faisal, T. F. M. Butarbutar, and P. Sirait, "Implementasi CNN dan SVM untuk Identifikasi Penyakit Tomat via Daun," *J. SIFO Mikroskil*, vol. 20, no. 2, pp. 117–134, 2019, doi: 10.55601/jsm.v20i2.670.
- [4] C. Senthilkumar and M. Kamarasan, "a Survey on Leaf Disease Detection Using Image Processing Techniques," vol. 5, no. 7, pp. 889–900, 2018, [Online]. Available: [www.jetir.org](http://www.jetir.org)
- [5] S. Pouyanfar *et al.*, "Dynamic Sampling in Convolutional Neural Networks for Imbalanced Data Classification," *Proc. - IEEE 1st Conf. Multimed. Inf. Process. Retrieval, MIPR 2018*, pp. 112–117, 2018, doi: 10.1109/MIPR.2018.00027.
- [6] A. Saxena, "An Introduction to Convolutional Neural Networks," *Int. J. Res. Appl. Sci. Eng. Technol.*, vol. 10, no. 12, pp. 943–947, 2022, doi: 10.22214/ijraset.2022.47789.
- [7] A. Salam, F. Yanto, S. Agustian, and S. Ramadhani, "Perbandingan Klasifikasi Citra CT-Scan Kanker Paru-Paru Menggunakan Contrast Stretching Pada CNN dengan EfficientNet-B0," *KLIK Kaji. Ilm. Inform. dan Komput.*, vol. 4, no. 3, pp. 1341–1351, 2023, doi: 10.30865/klik.v4i3.1514.
- [8] A. A. Hipzi, G. W. Wirianto, I. Made, and B. Sukmadana, "Klasifikasi Pneumonia Pada Augmentasi Citra X-Ray Paru-Paru Menggunakan Metode Convolution Neural Network (Cnn)," 2023, [Online]. Available: <http://eprints.unram.ac.id/id/eprint/39247>
- [9] I. M. D. Maysanjaya, "Klasifikasi Pneumonia pada Citra X-rays Paru-paru dengan Convolutional Neural Network," *J. Nas. Tek. Elektro dan Teknol. Inf.*, vol. 9, no. 2, pp. 190–195, 2020, doi: 10.22146/jnteti.v9i2.66.
- [10] J. Yopento, E. Ernawati, and F. F. Coastera, "Identifikasi Pneumonia Pada Citra X-Ray Paru-Paru Menggunakan Metode Convolutional Neural Network (CNN) Berdasarkan Ekstraksi Fitur Sobel," *Rekursif J. Inform.*, vol. 10, no. 1, pp. 40–47, 2022, doi: 10.33369/rekursif.v10i1.17247.
- [11] S. Lasniari, J. Jasril, S. Sanjaya, F. Yanto, and M. Affandes, "Klasifikasi Citra Daging Babi dan Daging Sapi Menggunakan Deep Learning Arsitektur ResNet-50 dengan Augmentasi Citra," *J. Sist. Komput. dan Inform.*, vol. 3, no. 4, p. 450, 2022, doi: 10.30865/json.v3i4.4167.
- [12] H. Almirza, S. Sanjaya, L. Handayani, and F. Syafria, "Klasifikasi Daging Sapi dan Daging Babi Menggunakan Convolutional Neural Network EfficientNet-B0 dengan Augmentasi Citra," *KLIK Kaji. Ilm. Inform. dan Komput.*, vol. 3, no. 6, pp. 1013–1021, 2023, doi: 10.30865/klik.v3i6.910.
- [13] I. A. DLY, Jasril, S. Sanjaya, L. Handayani, and F. Yanto, "Klasifikasi Citra Daging Sapi dan Babi Menggunakan CNN Alexnet dan Augmentasi Data," *J. Inf. Syst. Res.*, vol. 4, no. 4, pp. 1176–1185, 2023, doi: 10.47065/josh.v4i4.3702.
- [14] D. Putri Ayuni, Jasril, M. Irsyad, F. Yanto, and S. Sanjaya, "Augmentasi Data Pada Implementasi Convolutional Neural Network Arsitektur Efficientnet-B3 Untuk Klasifikasi Penyakit Daun Padi," *Zo. J. Sist. Inf.*, vol. 5, no. 2, pp. 239–249, 2023, doi: 10.31849/zn.v5i2.13874.
- [15] N. H. Putri, Jasril, M. Irsyad, S. Agustian, and F. Yanto, "Klasifikasi Citra Stroke Menggunakan Augmentasi dan Convolutional Neural Network EfficientNet-B0," *J. Media Inform. Budidarma*, vol. 7, no. 2, pp. 650–658, 2023, doi: 10.30865/mib.v7i2.5981.
- [16] A. Akram, K. Fayakun, and H. Ramza, "Klasifikasi Hama Serangga pada Pertanian Menggunakan Metode Convolutional Neural Network," *Build. Informatics, Technol. Sci.*, vol. 5, no. 2, pp. 397–406, 2023, doi: 10.47065/bits.v5i2.4063.
- [17] M. Agarwal, A. Singh, S. Arjaria, A. Sinha, and S. Gupta, "ToLeD: Tomato Leaf Disease Detection using Convolution Neural Network," *Procedia Comput. Sci.*, vol. 167, no. 2019, pp. 293–301, 2020, doi: 10.1016/j.procs.2020.03.225.
- [18] S. Ashok, G. Kishore, V. Rajesh, S. Suchitra, S. G. Gino Sophia, and B. Pavithra, "Tomato leaf disease detection using deep learning techniques," *Proc. 5th Int. Conf. Commun. Electron. Syst. ICCES 2020*, no. Icces, pp. 979–983, 2020, doi: 10.1109/ICCES48766.2020.09137986.
- [19] R. Soekarta, N. Nurdjan, and A. Syah, "Klasifikasi Penyakit Tanaman Tomat Menggunakan Metode Convolutional Neural Network (CNN)," *Insect (Informatics Secur. J. Tek. Inform.)*, vol. 8, no. 2, pp. 143–151, 2023, doi: 10.33506/insect.v8i2.2356.
- [20] N. Awalia, "Identifikasi Penyakit Leaf Mold Pada Daun Tomat Menggunakan Model Densenet121 Berbasis Transfer Learning," *J. Ilm. Ilmu Komput.*, vol. 8, no. 1, pp. 49–52, 2022, doi: 10.35329/jiik.v8i1.212.
- [21] A. Lawi, N. S. Intizhami, R. Mukhtarom, and S. Amir, "Klasifikasi Penyakit Citra Daun Tanaman Tomat Dengan Ensemble Convolutional Neural Network," *Sntei*, pp. 207–212, 2022.
- [22] K. KC, Z. Yin, M. Wu, and Z. Wu, "Depthwise separable convolution architectures for plant disease classification," *Comput. Electron. Agric.*, vol. 165, no. December 2018, p. 104948, 2019, doi: 10.1016/j.compag.2019.104948.
- [23] D. P. Hughes and M. Salathe, "An open access repository of images on plant health to enable the development of mobile disease diagnostics," 2015, [Online]. Available: <http://arxiv.org/abs/1511.08060>
- [24] O. Attallah, "Tomato Leaf Disease Classification via Compact Convolutional Neural Networks with Transfer Learning and Feature Selection," *Horticulturae*, vol. 9, no. 2, 2023, doi: 10.3390/horticulturae9020149.
- [25] S. Albawi, T. A. Mohammed, and S. Al-Zawi, "Understanding of a convolutional neural network," *Proc. 2017 Int. Conf. Eng. Technol. ICET 2017*, vol. 2018-Janua, pp. 1–6, 2017, doi: 10.1109/ICEngTechnol.2017.8308186.