

# **KLASIFIKASI PENYAKIT TANAMAN PADA DAUN KENTANG DENGAN METODE CONVOLUTIONAL NEURAL NETWORK ARSITEKTUR MOBILENETV1**

Disusun Oleh :

1. Muhammad Irpan (211351094) Malam A
2. An-Nada Zakiyya Nabila (211351017) Malam A
3. Fathan Naufal Rosidin (211351054) Malam A



**PROGRAM STUDI TEKNIK INFORMATIKA  
SEKOLAH TINGGI TEKNOLOGI WASTUKANCANA  
PURWAKARTA**

**2024**

# DAFTAR ISI

<b>Daftar Tabel .....</b>	<b>3</b>
<b>Daftar Gambar.....</b>	<b>4</b>
<b>I. Business Understanding .....</b>	<b>5</b>
<b>II. Data Understanding .....</b>	<b>7</b>
<b>III. Data Preparation .....</b>	<b>9</b>

## DAFTAR TABEL

Table 1. Explore Data .....	8
-----------------------------	---

**DAFTAR GAMBAR**

Gambar 3. Data Preparation 1 ..... 10

Gambar 3. Data Preparation 2 ..... 11

Gambar 3. Data Preparation 3 ..... 12

Gambar 3. Data Preparation 4 ..... 12

Gambar 3. Data Preparation 5 ..... 13

## I. Business Understanding

### A. Latar Belakang Masalah

Kentang merupakan jenis umbi yang mengandung banyak karbohidrat. Selain itu kentang dalam bahasa kimia *Solanum Tubersum L*, menjadi sumber pangan yang tumbuh di Negara Indonesia pada dataran tinggi. Kentang juga kaya akan karbohidrat, vitamin C, dan serat yang menjadi komoditas penting di Indonesia. Pada tahun 2012, produksi kentang nasional sebesar 1.069 juta ton yang menjadikannya komoditi keempat di dunia setelah padi, jagung, dan gandum. Sayangnya produksi kentang di Indonesia sering terkendala oleh penyakit tanaman yang menjadi tantangan dalam memenuhi kebutuhan kentang nasional. Pengembangan budidaya kentang yang lebih baik sangat diperlukan untuk menjaga berkelanjutan produksi. (Rosid et al., 2024)

Terjadinya penyakit pada tanaman kentang akan berdampak buruk pada produksi pertanian. Jika penyakit kentang terlambat terdeteksi dengan waktu yang telah di jadwalkan maka akan terjadi peningkatan penyakit tanaman kentang. Daun tanaman seringkali menjadi indikator awal adanya masalah kesehatan pada tanaman. Penyakit seperti bercak daun awal (*early blight*) dan hawar daun (*late blight*) yang ditandai dengan bercak-bercak kecoklatan atau pembusukan dapat menurunkan kualitas kentang secara signifikan. Keragaman gejala penyakit pada kentang membuat identifikasi menjadi tantangan bagi petani. (Beno et al., 2022)

Dalam menangani yang ada pada penyakit daun kentang ini tidak hanya dilakukan di bidang pertanian saja melainkan bidang teknologi pun ikut andil, salah satunya adalah pemanfaatan bidang informatika dalam mengidentifikasi penyakit yang ada pada tanaman kentang dengan menggunakan *image processing* atau biasa disebut pengolahan citra digital. Dengan memanfaatkan teknologi pengolahan gambar, kita bisa mendiagnosis penyakit tanaman kentang secara lebih akurat dan cepat. Hal ini sangat penting untuk meningkatkan produktivitas dan mengurangi akibat serangan penyakit. (Rozaqi et al., 2021).

Penelitian sebelumnya yang dilakukan oleh AM Lesmana, dkk (2022). Berdasarkan penelitiannya dapat ditemukan penyakit pada tanaman daun kentang melalui gambar, dengan epoch 50 dan batch size 32 memperoleh akurasi 97,90% dan loss 0,0390, metode

deep learning CNN model arsitektur MobileNetV1 dapat dengan akurat menemukan gambar penyakit pada daun kentang. Berdasarkan penjelasan masalah yang telah dipaparkan, penulis ingin mengklasifikasikan tanaman kentang berdasarkan citra pada daun dan batang menggunakan metode CNN dengan model *MobileNetV1*, teknologi ini diharapkan dapat menghasilkan klasifikasi penyakit pada tanaman kentang dengan memanfaatkan data gambar daun kentang.

## **B. Tujuan**

Tujuan dari penelitian klasifikasi penyakit tanaman jagung menggunakan metode CNN dengan model arsitektur *MobileNetv1* yaitu dengan mengidentifikasi jenis penyakit tanaman jagung dengan cara pengambilan gambar pada daun dapat memudahkan petani untuk mendeteksi jenis penyakit yang ada secara akurat dan cepat terlebih lagi tidak membutuhkan biaya yang besar.

## **C. Solusi**

Solusi yang akan di lakukan dari penelitian ini agar mencapai tujuan, sebagai berikut:

### **1. Pengumpulan Data**

Mengumpulkan gambar daun tanaman kentang yang terinfeksi berbagai jenis penyakit dan juga daun yang sehat. Data ini dapat di ambil dari sumber data public atau dengan mengambil gambar langsung di lapangan.

### **2. Preprocessing Data**

Memprocessing semua gambar memiliki ukuran yang sama agar dapat di proses oleh CNN, menambah variasi pada gambar, seperti rotasi, pencahayaan, zoom, serta mernormalisasi kan piksel gambar agar memudahkan untuk pelatihan.

### **3. Pembelajaran Mesin**

Model yang dipilih yaitu Convolutional Neural Network (CNN) dan arsitektur *MobileNetV1* untuk mendeteksi kondisi daun kentang yang terkena penyakit dan yang tidak terkena penyakit.

### **4. Validasi Model**

Untuk menilai kinerja model dalam mendeteksi kondisi daun kentang maka dilakukan analisis menggunakan grafik loss dan akurasi selama proses pelatihan serta *confusion matrix* setelah model selesai dilatih. Grafik loss membantu memantau bagaimana model mengurangi kesalahan selama pelatihan, sedangkan grafik akurasi menunjukkan kemampuan model dalam mengenali daun sehat dan daun sakit dengan benar. Jika terdapat perbedaan besar antara hasil training dan validation dapat menjadi indikasi adanya overfitting atau

underfitting. *Confusion matrix* digunakan untuk mengukur akurasi prediksi pada kelas daun sehat dan daun sakit, termasuk matrix penting seperti presisi, recall, dan F1-score.

## 5. Pengembangan Aplikasi

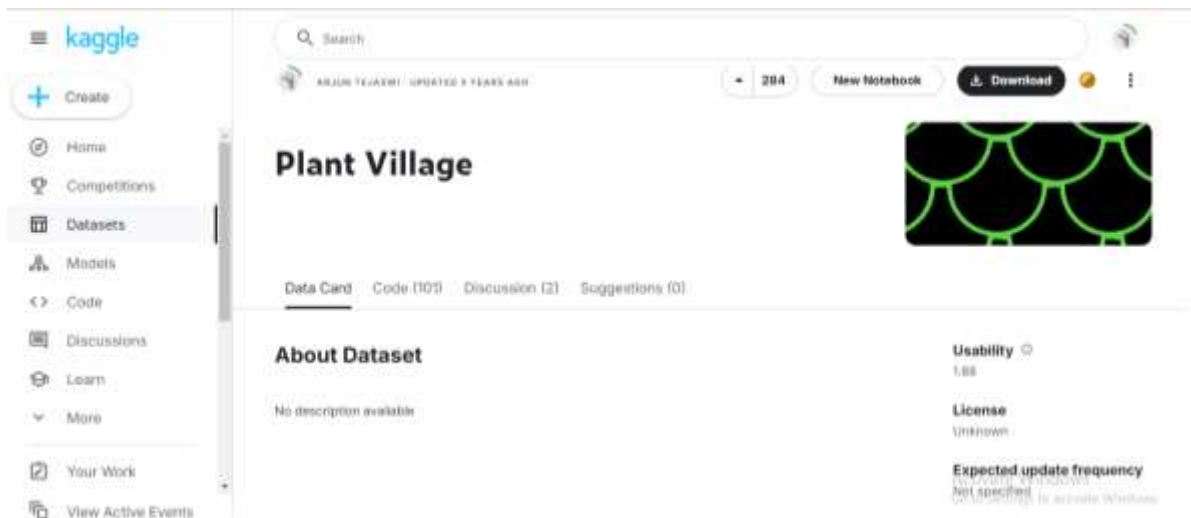
Pada tahap *deployment* model, *Streamlit* dipakai untuk menguji model yang sudah dibuat.

## D. Pengguna

Pengguna utama dari sistem ini yaitu petani yang membutuhkan sistem untuk mendeteksi penyakit pada tanaman secara cepat dan mudah. Dengan adanya aplikasi berbasis gambar petani bisa mengambil foto daun tanaman yang dicurigai terinfeksi penyakit, dan sistem akan memberikan hasil diagnosis serta rekomendasi perawatan. Selain itu mahasiswa juga bisa mempelajari sistem ini untuk latihan atau proyek akhir mereka. Mereka dapat belajar cara membangun model deep learning dalam konteks nyata yang berkaitan dengan pertanian.

# II. Data Understanding

## A. Sumber Data

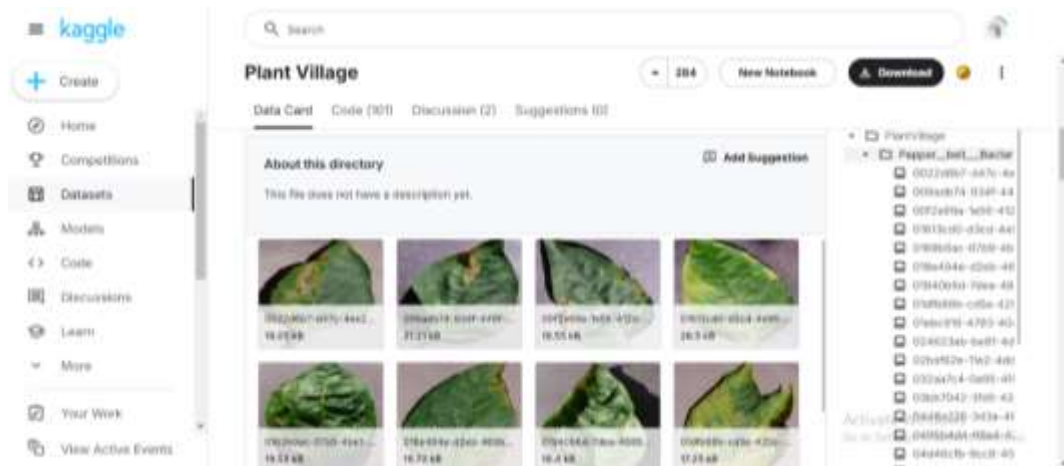


Gambar 1. Sumber Data 1

Link Dataset (<https://www.kaggle.com/datasets/arjuntejaswi/plant-village>)

Dataset yang digunakan diambil dari kaggle, dataset ini mencakup gambar daun dari berbagai jenis tanaman termasuk tanaman kentang dan lainnya. Gambar dalam dataset memiliki beberapa daun dalam kondisi sehat sementara yang lain menunjukkan gejala penyakit tertentu. Dataset ini berasal dari pemilik yang bernama Arjun Tejaswi.

## B. Deskripsi Data




Gambar 2. Dataset 1

Dataset ini memiliki 1.506 data dengan format *JPEG*. Dataset tersebut sudah terbagi berdasarkan penyakitnya seperti early blight, late blight, dan healthy. Ukuran dalam gambar dataset ini adalah 244x244 piksel. Dataset ini berisi beberapa kelas penyakit tanaman serta kategori daun sehat untuk berbagai jenis tanaman termasuk daun kentang.



## C. Explore Data

Berikut adalah contoh gambar yang akan di klasifikasi dalam penyakit tanaman kentang.

Table 1. Explore Data

NO	Gambar	Keterangan
1.		Daun Early Blight (hawar dini) adalah penyakit yang memiliki bercak coklat berbentuk lingkaran kosentris pada daun yang sering dikelilingi oleh area kuning. Biasanya bercak ini muncul pada daun yang lebih tua.



2.		Daun Late Blight (busuk daun) adalah penyakit yang memiliki bercak coklat gelap atau abu-abu kehijauan yang dikelilingi oleh area hijau muda. Bercak ini berkembang menjadi lesi besar yang bawah.
3.		Daun Healty adalah daun yang sehat berwarna hijau cerah tanpa adanya bercak atau perubahan warna. Warna hijau ini menunjukkan bahwa tanaman dapat melakukan fotosintesis secara optimal.

#### **D. Kualitas Data**

Dataset citra daun kentang ini memiliki kualitas yang baik. Semua citra memiliki resolusi yang baik dan detail yang jelas, memastikan integritas data untuk analisis. variasi dalam pencahayaan dan sudut pengambilan gambar telah di kelola dengan baik, sehingga tidak memengaruhi warna dan tekstur daun. Dengan kualitas data yang baik ini, model yang dikembangkan akan efektif dalam mendeteksi kondisi dari daun kentang.

### **III. Data Preparation**

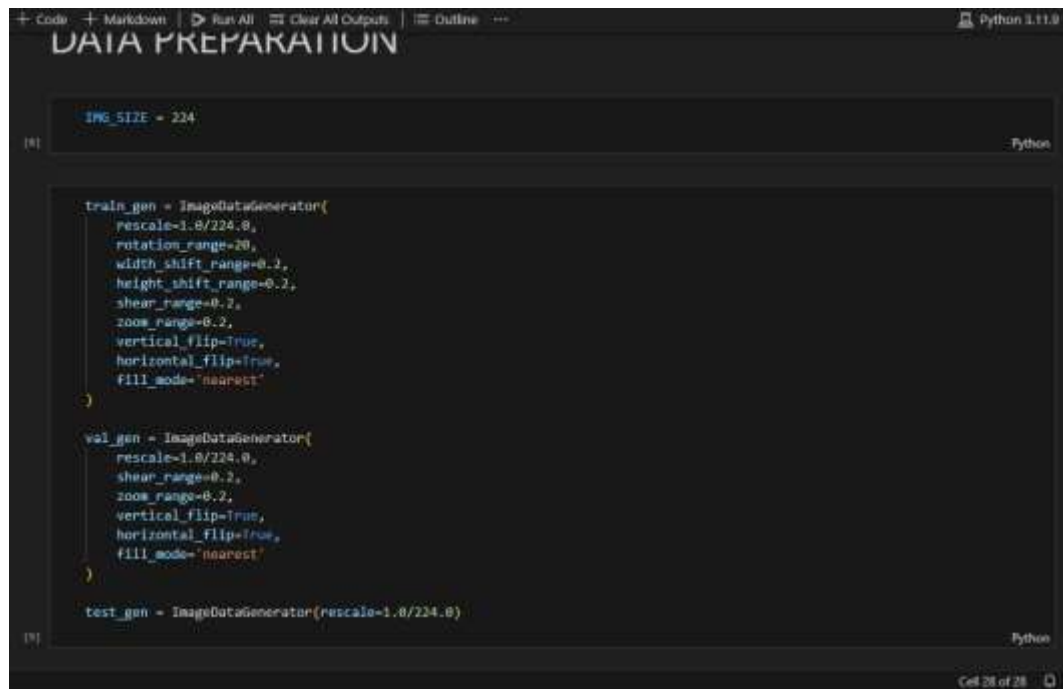
#### **A. Pemilihan Data**

Pada tahap ini, dipilih citra daun kentang yang benar-benar sesuai untuk tujuan penelitian. Hanya citra yang jelas, tidak buram, dan memiliki resolusi baik yang digunakan, sehingga model dapat menganalisis detail dengan lebih akurat. Kami memastikan jumlah citra antara daun sehat dan sakit seimbang, agar model bisa belajar mengenali kedua kondisi tanpa bias. Selain itu, citra yang dipilih mencakup variasi pencahayaan, sudut, dan

kondisi daun yang berbeda, sehingga model lebih siap menghadapi data nyata dengan berbagai kondisi.

## B. Data Preparation

Pada proyek ini, kita menggunakan Image Data Generator dari keras untuk mempersiapkan gambar daun kentang sebagai data pelatihan, validasi, dan uji. gambar-gambar ini akan melalui proses yang disebut data *augmentation* atau pengayaan data, yang bertujuan membuat model lebih kuat terhadap variasi.



```
DATA PREPARATION

IMG_SIZE = 224

train_gen = ImageDataGenerator(
    rescale=1.0/224.0,
    rotation_range=20,
    width_shift_range=0.2,
    height_shift_range=0.2,
    shear_range=0.2,
    zoom_range=0.2,
    vertical_flip=True,
    horizontal_flip=True,
    fill_mode='nearest'
)

val_gen = ImageDataGenerator(
    rescale=1.0/224.0,
    shear_range=0.2,
    zoom_range=0.2,
    vertical_flip=True,
    horizontal_flip=True,
    fill_mode='nearest'
)

test_gen = ImageDataGenerator(rescale=1.0/224.0)
```

Gambar 3. Data Preparation 1

Proses ini dilakukan dengan membuat versi baru dari setiap gambar pelatihan melalui beberapa transformasi acak, seperti:

- Normalisasi adalah proses mengubah nilai piksel agar berada dalam rentang antara 0 dan 1, sehingga model lebih mudah untuk dioptimalkan. Dengan menggunakan kode ***rescale=1.0/224.0*** yang bertujuan untuk membagi setiap piksel dalam gambar dengan 224.0, menurunkan nilai piksel ke rentang antara 0 dan 1. Normalisasi ini diterapkan pada `train_gen`, `val_gen`, dan `test_gen`.
- Rotasi dan geser dilakukan untuk menambah variasi sudut dan posisi objek pada gambar. Ini membantu model mengenali objek (daun kentang) dari berbagai sudut dan posisi. Dengan kode merotasi yaitu ***rotation\_range=20*** untuk Menentukan bahwa

gambar bisa diputar hingga  $\pm 20$  derajat secara acak, dan kode geser *width\_shift\_range=0.2* dan *height\_shift\_range=0.2* untuk Menerapkan pergeseran horizontal (lebar) dan vertikal (tinggi) hingga 20% dari ukuran gambar.

- Penskalaan (Zoom) memperbesar atau memperkecil gambar, sementara *shear* memiringkan gambar, menambahkan lebih banyak variasi pada data. Dengan kode *zoom\_range=0.2* yang berfungsi untuk memperbesar atau mengecilkan gambar secara acak dalam kisaran 20%, dan kode *shear\_range=0.2* untuk Menerapkan kemiringan hingga 20% untuk menghasilkan gambar dalam perspektif yang berbeda.
- Pembalikan secara vertikal dan horizontal memperkenalkan lebih banyak variasi posisi daun. dengan kode *vertical\_flip=True* untuk membalik gambar secara vertikal, dan kode *horizontal\_flip=True* untuk membalik gambar secara horizontal.
- Pengisian ruang kosong digunakan untuk mengisi area yang kosong setelah transformasi (seperti rotasi atau geser) agar gambar tetap terlihat alami. Dengan kode *fill\_mode='nearest'* yang Dimana Jika ada ruang kosong yang tercipta karena transformasi, pengaturan ini mengisi area tersebut dengan warna piksel terdekat, menjaga tampilan gambar tetap penuh.

Langkah selanjutnya yaitu memuat gambar dari folder sebagai data pelatihan validasi, dan uji menggunakan generator `flow_from_directory`.

```
train_data = train_gen.flow_from_directory(  
    train_dir,  
    target_size=(IMG_SIZE, IMG_SIZE),  
    class_mode='categorical',  
    shuffle=True  
)
```

Gambar 3. Data Preparation 2

Memuat data pelatihan dari folder yang berisi gambar dengan menggunakan data generator *train\_gen* dengan kode awal yaitu:

- *train\_data = train\_gen.flow\_from\_directory(...)* yang didalamnya berisikan perintah *train\_dir*, untuk penjaluran ke folder yang berisi data pelatihan.
- *target\_size=(IMG\_SIZE, IMG\_SIZE)*, untuk mengubah ukuran gambar menjadi 224x224 sesuai dengan ukuran foto yang sebelumnya sudah di tentukan agar konsisten dengan ukuran input yang diinginkan oleh model.

- ***class\_mode='categorical'*** untuk Mengatur mode kelas menjadi *categorical*, yang cocok untuk masalah klasifikasi dengan beberapa kelas (misalnya, mendeteksi beberapa jenis penyakit).
- ***shuffle=True*** untuk Mengacak data pelatihan sehingga model akan melihat gambar-gambar dalam urutan yang acak di setiap *epoch*, yang membantu dalam mencegah *overfitting*.

```
val_data = val_gen.flow_from_directory(
    val_dir,
    target_size=(IMG_SIZE, IMG_SIZE),
    class_mode='categorical',
    shuffle=True
)
```

Gambar 3. Data Preparation 3

Setelah memuat data pelatihan, selanjutnya memuat data validasi dengan menggunakan perintah

“***Val\_data = val\_gen.flow\_from\_directory(...)***” kode ini bertujuan untuk memuat gambar dari direktori yang ditunjukkan oleh *train\_dir*. Dan sisa kodenya sama seperti saat memuat data pelatihan Dimana semua gambar diubah menjadi ukuran 224x244 piksel agar konsisten, dan mengacak data pelatihan dan validasi untuk membantu model lebih baik dalam mempelajari data daun kentang, sedangkan data pengujian tidak di acak untuk menjaga konsisten hasil.

```
test_data = test_gen.flow_from_directory(
    test_dir,
    target_size=(IMG_SIZE, IMG_SIZE),
    class_mode='categorical',
    shuffle=False
)
```

Gambar 3. Data Preparation 4

Selanjutnya melanjutkan pemuatan data pada bagian data uji, ***test\_data*** ini bermaksud menyimpan data gambar pengujian yang telah diproses oleh *test\_gen*. ***test\_gen.flow\_from\_directory()*** disini bertujuan untuk memuat gambar dari direktori ***test\_dir***, yang akan digunakan untuk menguji performa akhir model. Setelah memuat gambar dari directory *test\_dir* lalu masuk ke tahap perubahan ukuran gambar seperti sebelumnya dengan ukuran 224x224 pixel agar konsisten dengan data pelatihan dan validasi. Yang berbeda di kode ini adalah data pengujian tidak idacak, sehingga urutannya tetap. Ini memastikan bahwa hasil pengujian model konsisten.

```
--- Found 1586 images belonging to 3 classes.  
Found 323 images belonging to 3 classes.  
Found 323 images belonging to 3 classes.
```

*Gambar 3. Data Preparation 5*

Output yang didapat menunjukkan bahwa di direktori pelatihan ( `train_dir` ), terdapat total 1506 gambar yang terbagi ke dalam 3 kelas yaitu: Daun Early Blight, Daun Late Blight, Daun Healty. Lalu untuk data validasinya ada 323 gambar untuk validasi dalam 3 kelas barusan, dan 323 gambar di baris terakhir menunjukkan direktori pengujian dalam 3 kelas.