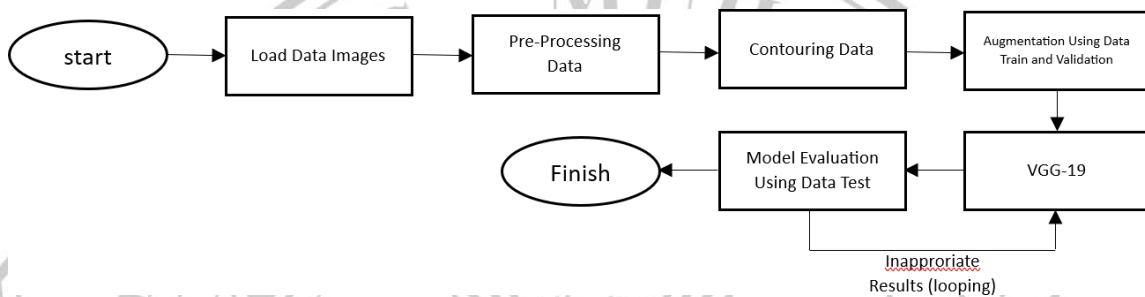


BAB III

METODE PENELITIAN

3.1. Tahapan Penelitian

Pada tahapan ini dijelaskan alur penelitian yang dilakukan untuk mengklasifikasi citra penyakit daun kopi menggunakan metode CNN dengan arsitektur model VGG-19 yang meliputi tahap pengumpulan data hingga tahap terakhir yaitu evaluasi model. Alur penelitian dapat dilihat pada gambar 3.1.



Gambar 3.1 Alur Penelitian

3.2. Dataset

Penelitian ini berfokus pada daun tanaman kopi yang dapat berpotensi oleh berbagai faktor seperti serangan hama, pola cuaca yang tidak dapat diprediksi, serta kemunculan jamur *Heileia Vastatrix*. Faktor-faktor inilah yang dapat menyebabkan perubahan pada daun tanaman kopi. Maka dari itu, peneliti melakukan klasifikasi untuk mendeteksi dan mengidentifikasi penyakit pada tanaman kopi pada daun. Situs web Kaggle menyediakan dataset untuk penelitian ini berjudul RoCoLe: Dataset Gambar Daun Kopi Robusta. Dataset ini terdiri dari 1560 gambar dan berisi tiga kelas: *healthy*, *rust*, dan *red spider mite*. Gambar 3.2 menunjukkan *sample* dataset dari tiap kelas.



Gambar 3.2 Sample Data Kelas *Healthy*, *Rust*, dan *Red Spider Mite*

3.3. Pembagian Dataset

Berdasarkan penelitian sebelumnya [2], agar jumlah dataset tiap kelas sama, dipilihlah 100 citra secara manual sesuai dengan kebutuhan penelitian tiap kelas. Kriteria data yang digunakan

adalah citra dengan fokus yang baik pada objek utama tanpa ada objek lain di sekitarnya. Pada penelitian ini, dataset dibagi menjadi data latih dengan rasio 80% dimana terdapat 240 citra, sedangkan data validasi sebagai data uji dengan rasio 20% dimana terdapat 60 citra.

3.4. Data Preprocessing

Data mentah diubah menjadi data latih melalui proses yang disebut *preprocessing* untuk meningkatkan akurasi model. Setelah mengimpor dataset dari *Google Drive* ke *Google Colaboratory*, data dibagi menjadi 240 gambar untuk data pelatihan, dan 60 gambar untuk data validasi. Gambar kemudian dikonversi ke ruang warna RGB dan diubah ukurannya menjadi 180 x 180 piksel untuk digunakan sebagai input untuk model CNN.

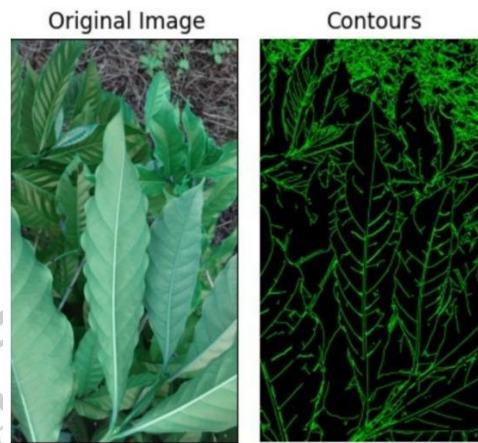
Dalam penelitian ini, normalisasi data merupakan salah satu tahapan yang umum dilakukan dalam *preprocessing*. Normalisasi data dilakukan dengan tujuan untuk mengubah skala data agar memiliki rentang atau distribusi yang konsisten. Hal ini memastikan bahwa semua fitur memiliki skala yang sama, sehingga tidak ada elemen yang mendominasi atau memiliki pengaruh yang lebih signifikan dibandingkan fitur lainnya. Normalisasi juga membantu mengurangi dampak *outlier* pada data. Teknik normalisasi yang digunakan dalam penelitian ini melibatkan normalisasi citra dengan menggunakan penskalaan *min-max*, membagi nilai piksel setiap lapisan dengan 255, yang menskalakan nilai ke internal 0-1, dengan 1 sebagai nilai piksel tertinggi dalam suatu gambar RGB [14].

3.5. Contouring

Kontur adalah garis melengkung yang menghubungkan titik-titik yang berdekatan dengan warna atau nilai yang sama. Ini adalah alat penting ketika ingin menganalisis bentuk dan mengidentifikasi objek. Kontur biasanya ditampilkan sebagai kumpulan titik lingkaran atau poligon. Metode CNN berhasil menangani tugas-tugas rumit, termasuk deteksi kontur, klasifikasi, dan segmentasi. Teknik ini bahkan dapat membedakan tepi dan kontur dari objek yang ditargetkan.

Pemrosesan kontur di *library* OpenCV terdiri dari dua fungsi: fungsi *find contour* yang berfungsi dapat memindai latar belakang data citra untuk objek, menangkap pola kontur dan membuat perkiraan, serta terdapat juga fungsi *draw contour* yang berfungsi untuk membuat katalog objek kontur dan memberdayakan user untuk menggambar data citra dengan ciri khas seperti warna dan lebar garis [20], [21], [22]. Penelitian ini menerapkan fungsi *draw contour* yang digunakan untuk menggambar garis *outline* pada citra, dan juga menerapkan fungsi *find contour*

yang digunakan untuk mengambil citra sebagai parameter dan mengembalikan daftar kontur yang ditemukan pada citra.



Gambar 3.3 Kontur Data

3.6. Augmentasi Data

Teknik augmentasi data dilakukan pada penelitian ini menggunakan *Image Data Generator* yang selain digunakan untuk menghindari *overfitting* pada data, juga digunakan untuk mengubah citra tanpa mengubah inti dari gambar tersebut [18]. Augmentasi dilakukan setelah data melalui *preprocessing*. Berikut pada Tabel 3.1 ditampilkan jenis augmentasi data yang dilakukan.

Tabel 3.1 Jenis Augmentasi Data

N o	Jenis Augmentasi	Deskripsi	Nilai
1	<i>Rotation</i>	Memutar citra dengan rentang 0 – 360°	45°
2	<i>Zoom</i>	Memperbesar citra dengan rentang 0 – 1	0.2
3	<i>Flipping</i>	Membalik citra dengan nilai <i>boolean</i> <i>True</i> atau <i>False</i>	<i>True</i>

3.7. Implementasi Model VGG-19

Penelitian ini menggunakan model VGG-19 yang terdiri dari beberapa lapisan seperti *Conv2D*, *MaxPooling2D*, *Flatten*, *Dense*, *Dropout*, serta menggunakan aktivasi ReLU dan *softmax* seperti yang ditampilkan pada Tabel 3.2.

Tabel 3.2 Metode yang Diajukan

Lapisan	Type	Output	Parameter
input_3	InputLayer (VGG-19)	(None, None, None, 3)	0
block1_conv1	Conv2D	(None, None, None, 64)	1792
block1_conv2	Conv2D	(None, None, None, 64)	36928
block1_pool	MaxPooling2D	(None, None, None, 64)	0
Block2_conv1	Conv2D	(None, None, None, 128)	73856
Block2_conv2	Conv2D	(None, None, None, 128)	147584
Block_pool	MaxPooling2D	(None, None, None, 128)	0
Block3_conv1	Conv2D	(None, None, None, 256)	295168
Block3_conv2	Conv2D	(None, None, None, 256)	590080
Block3_conv3	Conv2D	(None, None, None, 256)	590080
Block3_conv4	Conv2D	(None, None, None, 256)	590080
Block3_pool	MaxPooling2D	(None, None, None, 256)	0
Block4_conv1	Conv2D	(None, None, None, 512)	1180160
Block4_conv2	Conv2D	(None, None, None, 512)	2359808
Block4_conv3	Conv2D	(None, None, None, 512)	2359808
Block4_conv4	Conv2D	(None, None, None, 512)	2359808
Block4_pool	MaxPooling2D	(None, None, None, 512)	0
Block5_conv1	Conv2D	(None, None, None, 512)	2359808
Block5_conv2	Conv2D	(None, None, None, 512)	2359808
Block5_conv3	Conv2D	(None, None, None, 512)	2359808
Block5_conv4	Conv2D	(None, None, None, 512)	2359808
Block5_pool	MaxPooling2D	(None, None, None, 512)	0
Flatten_4	Flatten	(None, 25088)	0
Dense_12	ReLU	(None, 100)	2508900

Dropout_8	Dropout	(None, 100)	0
Dense_13	ReLU	(None, 50)	5050
Dropout_9	Dropout	(None, 50)	0
Dense_14	Softmax	(None, 3)	153

3.8. Confussion Matrix

Salah satu metode yang dilakukan pada penelitian ini adalah visualisasi *confussion matriks*, seperti akurasi, *recall*, dan *F1-Score* dapat digunakan untuk menilai kemandirian model. Keakuratan prediksi model diukur berdasarkan presentase seluruh observasi data yang positif [23].

Keempat rumus (1), (2), (3), (4) yang disebutkan dibawah ini dapat ditentukan dengan menggunakan *confussion matrix*, yang membantu mengevaluasi efisiensi model *deep learning* dengan membandingkan *output* dengan target *output*. Matriks ini dapat membantu peneliti dalam menciptakan model yang tepat dan efektif untuk mempelajari penyakit daun kopi. Matriks penilaian yang digunakan adalah *True Positive* (TP), *True Negative* (TN), *False Positive* (FP), dan *False Negative* (FN).

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

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

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

$$F1 - score = \frac{2 * Precision * Recall}{Precision + Recall} \quad (4)$$