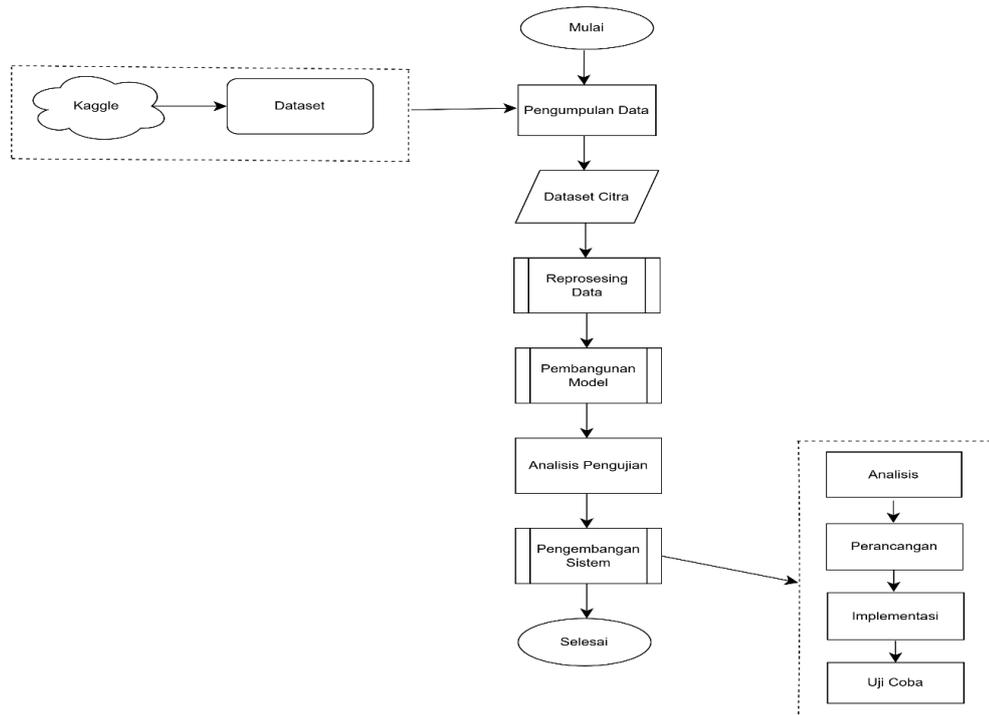


BAB III METODOLOGI PENELITIAN

3.1 Metodologi Penelitian

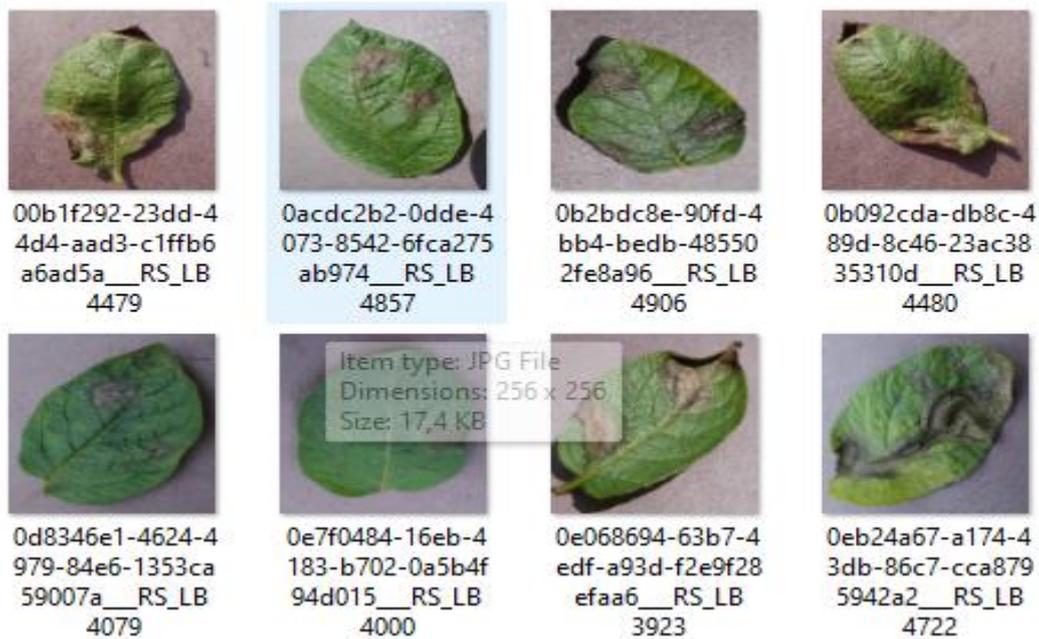
Penelitian yang dilakukan penulis menggunakan metode kuantitatif tipe eksperimental menggunakan dataset sekunder. Berikut tahapan pada penelitiannya:



Gambar 3. 1 Tahapan Penelitian

3.1.1 Pengumpulan Data

Proses pengumpulan data pada penelitian ini menggunakan data sekunder yang bersumber dari website kaggle dengan nama PlantVillage Dataset yang menyediakan sumber data terbuka. Pada website ini dipilih dataset kentang dengan jumlah klasifikasi 3 class. Data ini berjumlah 2.152, dan memiliki class, yaitu busuk daun (*Late Blight*), kering daun (*Early Blight*), dan sehat (*Healty*). *Late Blight* memiliki 1.000 citra, *Early Blight* memiliki 1.000 citra, *Healty* memiliki 152 citra.



Gambar 3. 2 Citra Daun Kentang

3.1.2 Data Preprocessing

Pembagian dataset pada penelitian ini dijelaskan pada Tabel 3.1 berikut:

Tabel 3. 1 Pembagian data latih dan data Uji

No.	Nama	Jenis	Jumlah
1.	Citra Kentang Early Blight	Training	701
2.	Citra Kentang Healty	Training	665
3.	Citra Kentang Late Blight	Training	700
4.	Citra Kentang Early Blight	Testing	150
5.	Citra Kentang Healty	Testing	143
6.	Citra Kentang Late Blight	Testing	150
7.	Citra Kentang Early Blight	Validation	150
8.	Citra Kentang Healty	Validation	142
9.	Citra Kentang Late Blight	Validation	150

Setelah dilakukan pembagian data, kemudian dilakukan proses selanjutnya:

a. Augmentasi Data

Proses augmentasi data citra daun untuk proses training dilakukan untuk mencegah terjadinya overfitting (memiliki kinerja baik selama pelatihan, tetapi buruk pada data baru). Beberapa teknik yang dilakukan pada proses augmentasi data dapat dilihat pada tabel 2 berikut:

Tabel 3. 2 Teknik pada Proses Data Augmentasi

No.	Parameter	Nilai
1.	Rescale	1.0/255.0
2.	Rotation range	20
3.	Width shift range	0.2
4.	Height shift range	0.2
5.	Shear range	0.2
6.	Zoom range	0.2

Hasil dari proses tabel diatas dapat dilihat pada gambar 3.3 Berikut:



Gambar 3. 3 Citra Daun Kentang yang sudah di Augentasi

3.2 Pembangunan Model CNN

Proses Pembangunan model CNN ini merupakan proses melakukan pembelajaran mesin untuk menemukan pola didalam data. Tahapan yang pertama yaitu menggunakan data citra yang sudah melalui tahap *preprocessing* data untuk melakukan proses *training*. Setelah proses *training* dilakukan, maka akan menghasilkan model.

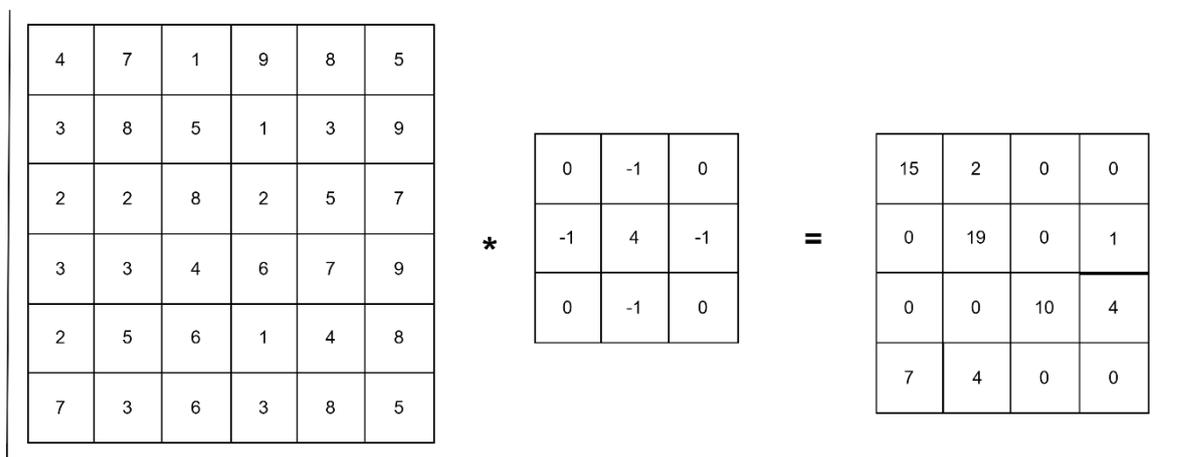
3.2.1 Input Image

Tahapan pertama yaitu memasukkan data citra daun ke input layer untuk melakukan pelatihan terhadap model, pada layer ini citra gambar dikonversi kedala matriks 3 dimensi dengan ukuran Panjang x lebar x 3 channels RGB (*Red, Green, Blue*).

3.2.2 Convolution

Proses convolusi berfungsi untuk mengektrasi fitur-fitur (feature map) yang ada pada cira dengan menggunakan filter. Penelitian ini menggunakan filter size berukuran 3x3 pixel,

dengan stride 1 dan tanpa padding. Pada CNN sendiri filter size juga disebut sebagai kernel atau waight yang nilainya diinisialisasi secara acak. Contoh prosesnya dapat dilihat pada gambar 3.4 berikut.



Gambar 3. 4 Contoh Proses Konvolusi

Pada ilustrasi gambar diatas terdapat input citra berukuran 6x6 yang akan dikonvolusi menggunakan filter size berukuran 3x3 yang akan dihasilkan feature map berukuran 4x4. Operasi konvolusi dilakukan dengan menggeser kernel konvolusi ixel per pixel. Hasil konvouisi disimpan didalam matrix yang baru. Proses perhitungannya adalah sebagai berikut $(0 \times 4) + (-1 \times 7) + (0 \times 1) + (-1 \times 3) + (4 \times 8) + (-1 \times 5) + (-1 \times 2) + (0 \times 8) = 15$. Dari proses tersebut menghasilkan nilai 15 yang akan menempati 1 buah sel di cira yang baru. Setelah iu filter digeser lagi kekanan dan kebawah hinggga terbenuk 1 citra baru.

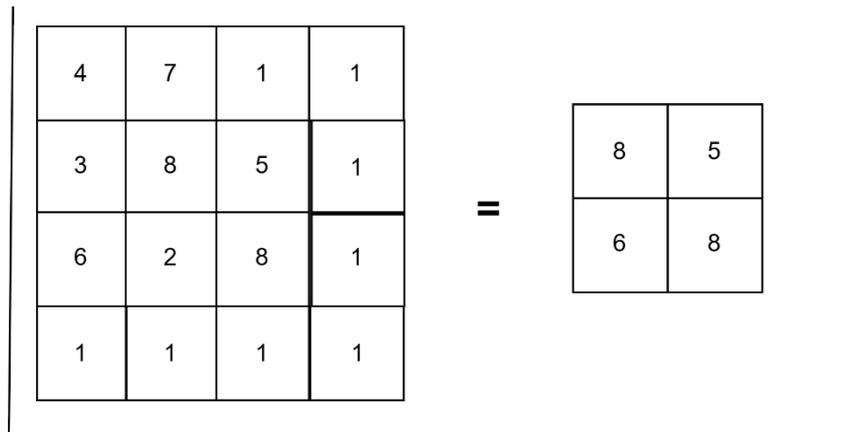
3.2.3 Activation function

Untuk mendapatkan objek dalam citra diperluka pemisahan objek dengan latar belakang pada objek. Pada penelitian ini activaation function ReLU digunakan untuk menentukan aktif tidaknya neuron pada neural network sehingga hanya neuron yang berhubungan dengan objek saja yang dipilih.

3.2.4 Pooling Layer

Pooling layer berfungsi untuk mengurangi ukuran spasial dari cira dan mengurangi jumlah parameter dan perhitungan dalam neural network. Pada lapisan ini menerima input

dari hasil feature mapp dari hasil konvolusi pada convoution layer. Penelitian ini menggunakan max pooling 2x2 sehingga citra yang dihasilkan lebih kecil. Sebagai contoh pada gambar 4. Dibawah merupakan input max pooling, proses ini menghasilkan ouput dengan nilai yang terbesar dari semua nilai input, ada kasus ini 8 adalah nilai terbsar sehingga nilai tersebut lahy yang diambil.



Gambar 3. 5 Contoh proses pooling

Berdasarkan gambar 3.5 dengan elemen berwarna hijau nilai terbesarnya adalah 8 maka nilai yang digunakan untuk input citra baru adalah 8 begitupun untuk elemen kolom kedua dengan nilai tersebar adalah 5 makan input cira barunya adalah 5, begitupun seterusnya. Proses komputasi yang terjadi pada pooling layer dijelaskan seperti pada tabel 3.3 berikut

Tabel 3. 3 Proses Pooling Layer

No	Sel	Nilai Area Matriks	Hasil
1	F(1,1)	4,7,3,8	8
2	F(1,2)	1,1,5,1	5
3	F(1,3)	6,2,1,1	6
4	F(1,4)	8,1,1,1	8

3.2.5 Fully Connected Layer

tahapan selanjutnya adalah fully connected layer, tahapan awal adalah mengubah data matriks 3 dimensi pada tahap konvolusi menjadi satu dimensi vector (flatten). Koputasi fully connected layer diilustrasikan seperti tabel 3.4 berikut:

Tabel 3. 4 Proses flattening

Data Matriks			Data Vektor (x)							
6	4	9								
7	3	5	6	4	9	7	3	5	1	2
1	2	7								

Data vektor tersebut akan melakukan proses perhitungan seperti pada persamaan 3.1, apabila dihitung secara manual akan terlihat seperti berikut :

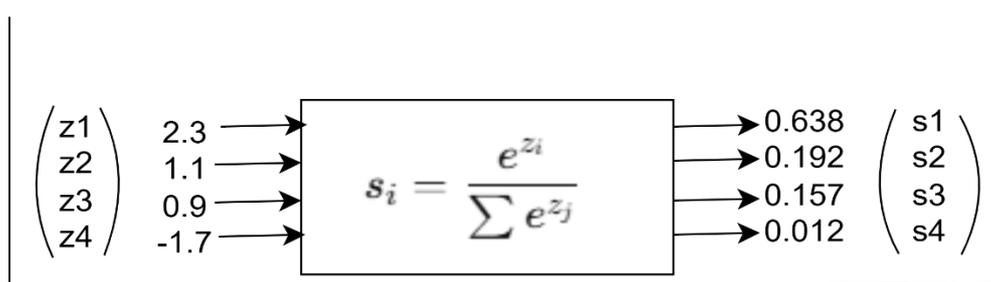
$$y = x_1w_1 + x_2w_2 + x_3w_3 + x_4w_4 + x_5w_5 + x_6w_6 + x_7w_7 + x_8w_8 + x_9w_9$$

$$y = (6 \cdot 0,5) + (4 \cdot 0,71) + (9 \cdot 0,25) + (7 \cdot 0,79) + (3 \cdot 6,58) + (5 \cdot -1,25) + (1 \cdot 3,11) + (2 \cdot -2,5) + (7 \cdot 5,20)$$

$$y = 3+2,84+2,25+5,53+19,74+(-6,25)+3,11+(-5)+36,4 \quad y = 61,72 \text{ (Situngkir, 2022)}$$

3.2.6 Output Layer

Output Layer merupakan lapisan yang merepresentasikan hasil akhir dari proses klasifikasi. Tujuan dari layer ini adalah untuk memprediksi output klasifikasi dari segi nilai probabilitas, dimana nilai probabilitas kelas terbesar merupakan output prediksi kelas yang diperoleh. Penelitian ini menggunakan fungsi aktivasi Softmax, proses yang terjadi diilustrasikan seperti Gambar 3.6 berikut :



Gambar 3. 6 Proses Klasifikasi Menggunakan Softmax

Keterangan:

Z = Layer Output sebelum softmax

S = Probabilitas

3.2.7 Membuat Rencana Kombinasi Pengujian

pada tahap ini dilakukan perbandingan beberapa parameter pada arsitektur CNN agar didapatkan model yang terbaik. Adapun parameter-parameter yang dilakukan perbandingan adalah nilai epoch, jenis oprimizer dan skenario Convoluinal Layer. Kombinasi yang akan diujikan ada peneitian ini dapat diliha pada tabel 3.5 berikut ini.

Tabel 3. 5 Kombinasi Pengujian

No	Hyperparameter	Parameter	Parameter	Parameter
1	Optimizer	Adam	RMSProp	SGD
2	Epoch	15	25	50
3	Convolutional Layer	4	4	4

3.2.7 Rencana Pengujian dan Indikaor Keberhasilan Penelitian

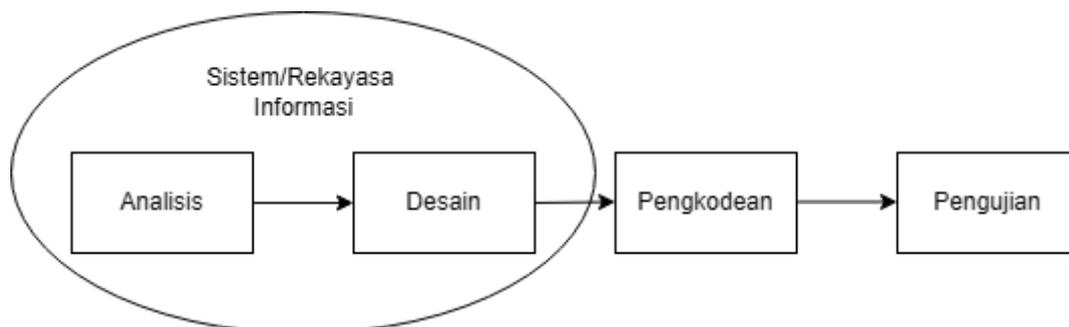
Parameter pertama yang akan dilakukan perbandingan untuk memperoleh arsitekur terbaik adalah parameter *Optimizer*. Pada penelitian ini jenis Optimizer yang akan dilakukan perbandingan adalah Adam, RMSProp dan SGD. Hasil training CNN menunjukkan bahwa pemilihan jenis optimizer akan memberikan ingkat akurasi dan losses yang berbeda-beda (Wikarta, Pramono, & Ariatedja, 2020). Parameter selanjutnya yang akan dilakukan perbandingan adalah epoch karena penguunaan epoch sangat berpengaruh terhadap akurasi yang dihasilkan (Wasil, Harianto, & Fathurrahman, 2022). Parameter terakhir yang akan dilakukan perbandingan adalah layer konvolusi dengan perbandingan layer sebanyak 4 dan 5. Berdasarkan parameter yang diujikan pada tabel 3.6 menghasilkan kombinasi sebanyak 18 kombinasi yang dapat dilihat pada tabel 3.6 berikut :

Tabel 3. 6 Rencana Pengujian Kombinasi Hyperparameter dan Opimizer

No	Optimizer	Epoch	Convolution Layer	Akurasi
1	Adam	15	4	87,83%
2	Adam	15	5	
3	Adam	25	4	
4	Adam	25	5	
5	Adam	50	4	
6	Adam	50	5	
7	RMSprop	15	4	
8	RMSprop	15	5	
9	RMSprop	25	4	
10	RMSprop	25	5	
11	RMSprop	50	4	
12	RMSprop	50	5	
13	SGD	15	4	
14	SGD	15	5	
15	SGD	25	4	
16	SGD	25	5	
17	SGD	50	4	
18	SGD	50	5	

3.3 Pengembangan perangkat lunak

Pengembangan perangkat lunak dilakukan dengan menggunakan metode waterfall, metode ini memiliki pendekatan sekuensial dengan proses yang sistematis. Adapun metode pengembangan system yang digunakan adalah metode waterfall yang dimulai dari Analisis, Desain, Pengkodean dan Pengujian.



Gambar 3. 7 Model Waterfall
sumber (s & shalahuiddi 2014)

3.3.1 analisis kebutuhan perangkat lunak

Proses pengumpulan kebutuhan dilakukan secara intensif untuk menspesifikasikan kebutuhan system agar dapat dengan mudah dipahami dan dapat dibangun sesuai dengan kebutuhan.

3.3.2 Desain

Tahap ini mentranslasi kebutuhan system dari tahap analisis kebutuhan system ke representasi desain agar dapat diimplementasikan menjadi program selanjutnya.

3.3.3 Pengkodean

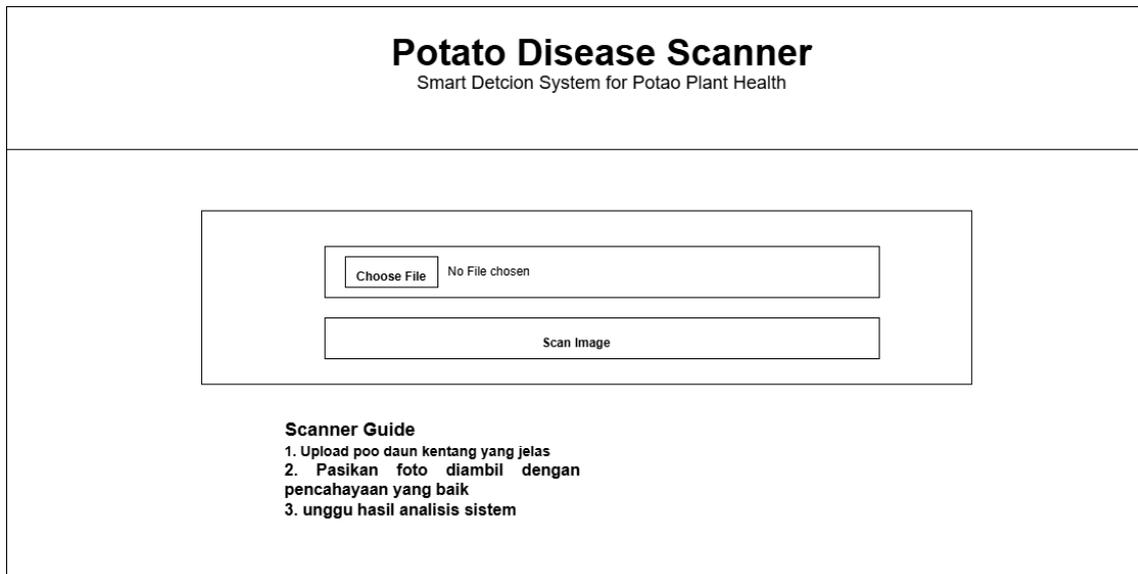
Desain harus ditranslasikan kedalam program menggunakan Bahasa pemrograman python. Hasil dari tahapan ini program computer sesuai dengan desain yang telah dibuat pada tahap desain system.

3.3.4 Pengujian

Pengujian focus pada fungsional system untuk memastikan keluaran system telah sesuai dengan kebutuhan pengguna.

3.4 Rancangan Antarmuka Aplikasi

Berdasarkan hasil analisis kebutuhan system, diartikan rancangan antarmuka aplikasi yang di tampilkan pada gambar 3.5 berikut:



Gambar 3. 8 Rancangan antarmuka aplikasi

Pada halaman utama ini user dapat melakukan penimoutan citra dengan menekan tombol choose file, kemudian pilih citra yang ingin di klasifikasikan. Seelah gambar daun kentang di tampilkan user dapat melakukan pemeriksaan apakah daun kentang yang dipilih terinfksi penyakit atau normal dengan menekan tombol scan image sehingga diampilkan hasil klasifikasinya. Aplikasi hanya bisa memproses citra yang berekstensi .jpg, .png, .jpeg saja.

3.5 Uji Coba

Setelah perangkat lunak dibangun akan dilakukan pengujian. Pengujian dengan melakukan pengujian pada semua fungsi untuk menemukan error dan memastikan masukan yang telah ditetapkan memberikan keluaran yang diharapkan. Tabel rencana pengujian dapat dilihat pada tabel 3.7 berikut:

Tabel 3. 7 Rencana Pengujian

Skenario ke	Test Case	Ekspetasi Hasil
1	Input Citra Normal .jpg	Berhasil
2	Input Citra Berpenyakit .png	Berhasil
3	Input Citra .jpeg	Berhasil

Tabel 3.7 merupakan skenario dari pengujian yang akan dilakukan oleh siste. Apabila citra yang di inpukan bertipe jpg, png dan jpeg, maka sistem akan mampu melakukan prses klasifikasi dengan benar, apabila sistem diberi input citra beripe selain itu, maka citra tidak akan bisa di klasifikasi.

Tabel 3. 8 Rencana pengujian Augmentasi

No.	Gambar	Jenis Penyakit	Klasifikasi	Kesimpulan
1.				
2.				
3.				
4.				
5.				

Tabel 3.8 Adalah rencana pengujian untuk citra gambar yang sudah di augmentasi menggunakan rotasi dengan besar 40 derajat dan 70 deraja, brightness ditambah 30% dan dikurangi 30%. Apabila menginputkan cira yang telah diaugmentasi, maka sistem akan mampu melakukan proses klasifikasi. Hasil proses klasfikasi akan disimpulkan dalam bentuk akurasi.