

**KLASIFIKASI PENYAKIT PADA DAUN MANGGA DENGAN
MENGUNAKAN METODE CONVOLUTIONAL NEURAL
NETWORK (CNN)**

SKRIPSI

**ABID TONDI NUGRAHA
191401001**



**PROGRAM STUDI S-1 ILMU KOMPUTER
FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI
UNIVERSITAS SUMATERA UTARA
MEDAN
2023**

**KLASIFIKASI PENYAKIT PADA DAUN MANGGA DENGAN
MENGUNAKAN METODE CONVOLUTIONAL NEURAL
NETWORK (CNN)**

SKRIPSI

Diajukan untuk melengkapi tugas dan memenuhi syarat memperoleh ijazah

Sarjana Ilmu Komputer

ABID TONDI NUGRAHA

191401001



**PROGRAM STUDI S-1 ILMU KOMPUTER
FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI
UNIVERSITAS SUMATERA UTARA
MEDAN
2023**

PERSETUJUAN

Judul : KLASIFIKASI PENYAKIT PADA DAUN
MANGGA DENGAN MENGGUNAKAN METODE
CONVOLUTIONAL NEURAL NETWORK (CNN)

Kategori : SKRIPSI

Nama : ABID TONDI NUGRAHA

Nomor Induk Mahasiswa : 191401001

Program Studi : SARJANA (S-1) ILMU KOMPUTER

Fakultas : ILMU KOMPUTER DAN TEKNOLOGI
INFORMASI

Medan, 06 Desember 2023

Komisi Pembimbing :

Dosen Pembimbing II



Desilia Selvida, S.Kom., M.Kom.
NIP 198912052020012001

Dosen Pembimbing I



Dr. Ir. Elviawaty Muisa Zamzami, S.T., M.T., M.M.
NIP 197007162005012002

Diketahui/disetujui oleh

Program Studi S1 Ilmu Komputer



Dr. Amalia S.T., M.T.
NIP 197812212014042001

PERNYATAAN**KLASIFIKASI PENYAKIT PADA DAUN MANGGA DENGAN
MENGUNAKAN METODE CONVOLUTIONAL NEURAL
NETWORK (CNN)****SKRIPSI**

Saya mengakui bahwa skripsi ini adalah hasil karya saya sendiri, kecuali beberapa kutipan dan ringkasan yang masing-masing telah disebutkan sumbernya

Medan, Desember 2023



Abid Tondi Nugraha

191401001

PENGHARGAAN

Alhamdulillah, Segala puji bagi Allah SWT. Tuhan Semesta Alam, yang telah memberikan kenikmatan berupa Iman, Islam, keselamatan, rahmat, dan juga kesehatan serta kekuatan kepada penulis, sehingga penulis dapat menyelesaikan penyusunan skripsi ini sebagai syarat untuk memperoleh gelar Sarjana Komputer di Program Studi S-1 Ilmu Komputer, Fakultas Ilmu Komputer dan Teknologi Informasi, Universitas Sumatera Utara.

Selanjutnya, *shalawat* dan *salam* kepada Baginda Nabi Muhammad SAW. yang telah mengajarkan ajaran Islam dengan sempurna dan menyempurnakan sifat & akhlak manusia di muka bumi ini.

Dengan kerendahan hati dan rasa syukur, penulis ingin menyampaikan rasa hormat dan terima kasih yang sebesar-besarnya kepada:

1. Bapak Prof. Dr. Muryanto Amin, S.Sos, M.Si selaku Rektor Universitas Sumatera Utara
2. Ibu Dr. Maya Silvi Lydia, B.Sc., M.Sc., selaku Dekan Fakultas Ilmu Komputer dan Teknologi Informasi, Universitas Sumatera Utara.
3. Ibu Dr. Amalia, ST. MT., selaku Kepala Program Studi S-1 Ilmu Komputer, Fakultas Ilmu Komputer dan Teknologi Informasi, Universitas Sumatera Utara.
4. Ibu Anandhini Medianty Nababan, S.Kom., M.T., selaku Dosen Pembimbing Akademik yang telah banyak memberikan bimbingan, saran dan dukungan akademisi selama kuliah hingga pengerjaan skripsi ini.
5. Ibu Dr. Ir. Elviawaty Muisa Zamzami, S.T., M.T., M.M., selaku Dosen Pembimbing I yang telah banyak memberikan bimbingan, saran, masukan dan dukungan dalam pengerjaan skripsi ini.
6. Ibu Desilia Selvida, S.Kom., M.Kom., selaku Dosen Pembimbing II yang telah banyak memberikan bimbingan, saran, masukan dan dukungan dalam pengerjaan skripsi ini.

7. Dr. T. Henny Febriana Harumy, S.Kom., M.Kom., selaku Dosen Penguji I yang telah memberikan arahan, saran, masukan dan dukungan dalam pengerjaan skripsi ini.
8. Dr. Amalia, ST., M.T., selaku Dosen Penguji II yang telah memberikan arahan, saran, masukan dan dukungan dalam pengerjaan skripsi ini.
9. Seluruh tenaga pengajar dan pegawai Program Studi S-1 Ilmu Komputer, Fakultas Ilmu Komputer dan Teknologi Informasi, Universitas Sumatera Utara.
10. Keluarga tersayang yaitu Ayahanda Hadi Susiono, M.Pd dan Ibunda Nurhati Ritonga, M.A., serta saudara kandung penulis, yaitu Kakanda Asri Maulida Ramadhani, M.Pd., Abangda Arif Faisaluddin, S.P., dan Kakanda Khairani Zahra, S.AP.. Terima kasih untuk segala dukungan doa, nasihat, motivasi, dukungan dan kerja kerasnya kepada penulis hingga saat ini.
11. Teman-teman seperjuangan skripsi S1 Ilmu Komputer USU stambuk 2019, terkhusus untuk Kom A, yang sudah menemani dan melengkapi penulis dalam proses belajar di program studi S-1 Ilmu Komputer USU.
12. Keluarga besar IMILKOM USU, PEMA Fasilkom-TI USU dan UKMI Al-Khuwarizmi Fasilkom-TI USU, yang telah memberikan pengalaman organisasi kepada penulis selama satu periode kepengurusan.
13. Dan semua pihak yang telah banyak membantu dan mendukung penulis yang tidak bisa disebutkan satu per satu.

Semoga seluruh kebaikan yang telah diberikan akan kembali kepada pihak yang memberi dengan tulus juga dari Allah SWT. Semoga skripsi ini dapat memberikan manfaat yang baik kedepannya.

Medan, Desember 2023



Penulis

ABSTRAK

Mangga (*Mangifera Indica*) merupakan salah satu tanaman yang sangat populer dan disukai oleh masyarakat Indonesia. Adanya serangan hama, virus atau jamur pada tanaman mangga menghasilkan kondisi buah mangga yang tidak maksimal dan busuk, yang memberikan kerugian pada produktif buah, maka dibutuhkanlah sebuah sistem yang mampu mengidentifikasi jenis penyakit pada tanaman mangga dengan melakukan klasifikasi gambar penyakit pada daun mangga. Sistem klasifikasi penyakit pada daun mangga dibangun dengan menggunakan Metode *Convolutional Neural Network* dan berbasis website, menggunakan *Library Flask*. Metode *Convolution Neural Network* dengan arsitektur VGG-16 mampu mendeteksi gambar penyakit pada daun mangga sangat baik, dengan proses pelatihan model sebanyak 15 epoch & 8 steps untuk jumlah data sebanyak 1500 data, dengan durasi waktu 13.225 detik atau setara dengan 3 jam 40 Menit 55 detik, serta hasil akurasi yang diperoleh selama proses pelatihan secara keseluruhan adalah mencapai 0,986.

Kata Kunci: Penyakit Tanaman Mangga, Klasifikasi Gambar, *Convolutional Neural Network*, *Website*

CLASSIFICATION OF DISEASES ON MANGO LEAVES USING THE CONVOLUTIONAL NEURAL NETWORK (CNN) METHOD

ABSTRACT

Mango (*Mangifera Indica*) is one of the plants that is very popular and favored by the people of Indonesia. The attack of pests, viruses or fungi on mango plants results in the condition of mango fruit that is not optimal and rotten, which provides a loss in fruit productivity, so a system is needed that is able to identify the type of disease in mango plants by classifying disease images on mango leaves. The disease classification system on mango leaves is built using the Convolutional Neural Network Method and is web-based, using the Flask Library. The Convolution Neural Network method with VGG-16 architecture is able to detect disease images on mango leaves very well, with a model training process of 15 epochs & 8 steps for the amount of data as much as 1500 data, with a duration of 13,225 seconds or equivalent to 3 hours 40 minutes 55 seconds, and the accuracy results obtained during the training process as a whole is 0.986.

Keywords: *Mango Plant Disease, Image Classification, Convolutional Neural Network, Website*

DAFTAR ISI

PERSETUJUAN.....	ii
PERNYATAAN.....	iii
PENGHARGAAN.....	iv
ABSTRAK	vi
DAFTAR ISI.....	viii
DAFTAR TABEL	xi
DAFTAR GAMBAR.....	xii
DAFTAR LAMPIRAN	xiv
 BAB 1 PENDAHULUAN	 1
1.1. Latar Belakang Masalah.....	1
1.2. Rumusan Masalah.....	3
1.3. Batasan Masalah.....	3
1.4. Tujuan Penelitian	3
1.5. Manfaat Penelitian	3
1.6. Metodologi Penelitian.....	4
1.7. Sistematika Penelitian	4
 BAB 2 LANDASAN TEORI	 6
2.1. Mangga	6
2.1.1. Deskripsi Mangga	6
2.1.2. Klasifikasi Tanaman Mangga	7
2.1.3. Penyakit pada Tanaman Mangga	7

2.2. Machine Learning	11
2.2.1. <i>Supervised Learning</i>	11
2.2.2. <i>Unsupervised Learning</i>	11
2.2.3. <i>Reinforcement Learning</i>	12
2.3. Deep Learning	12
2.4. Convolutional Neural Network	13
2.4.1. <i>CNN Layer</i>	14
2.4.2. <i>Visual Geometry Group (VGG) 16</i>	18
2.5. Penelitian Relevan	20
 BAB 3 ANALISIS DAN PERANCANGAN.....	 22
3. 1. Analisis Sistem	22
3.1.1. Analisis Masalah.....	22
3.1.2. Analisis Data.....	23
3.1.3. Analisis Kebutuhan.....	24
3.2 Arsitektur Umum	24
3.3. Pemodelan Sistem.....	26
3.3.1. <i>Use Case Diagram</i>	26
3.3.2. <i>Activity Diagram</i>	27
3.3.3. <i>Sequence Diagram</i>	28
3.3.4. <i>Flowchart (Diagram Alir)</i>	28
3.4 Perancangan Antarmuka.....	30
3.4.1. Desain Halaman Beranda.....	31
3.4.2. Desain Halaman Deteksi.....	32
3.4.3. Desain Halaman Bantuan.....	34

3.4.4. Desain Halaman Tentang.....	35
BAB 4 IMPLEMENTASI DAN PENGUJIAN.....	36
4.1. Implementasi Sistem	36
4.1.1. Halaman Beranda.....	36
4.1.2. Halaman Deteksi.....	37
4.1.3. Halaman Bantuan.....	38
4.1.4. Halaman Tentang.....	39
4.2. Pengujian Sistem	40
4.2.1. Tahap Pelatihan Model	40
4.2.2. Tahap Integrasi	52
BAB 5 KESIMPULAN DAN SARAN	58
5.1. Kesimpulan.....	58
5.2. Saran	59
DAFTAR PUSTAKA	60

DAFTAR TABEL

Tabel 4. 1. Hasil <i>Precision</i> , <i>Recall</i> dan <i>F1-Score</i>	43
Tabel 4. 2. Hasil dari <i>Accuracy</i> , <i>Loss</i> & <i>Training Time</i>	44
Tabel 4. 3. Hasil dari Percobaan Prediksi.....	45
Tabel 4. 4. Hasil Pengujian.....	55

DAFTAR GAMBAR

Gambar 2. 1 (a) Daun dan Buah Mangga, (b) Pohon Mangga.....	7
Gambar 2. 2 Kutu Putih.....	8
Gambar 2. 3 Embun Jelaga.....	9
Gambar 2. 4 Antraknosa.....	10
Gambar 2. 5 Penyakit Gloeosporium	10
Gambar 2. 6 Kelas dari <i>Machine Learning</i> (Goodfellow et al. 2016).....	11
Gambar 2. 7 Arsitektur dari CNN (Ghimire et al., 2022).....	13
Gambar 2. 8 Cara Kerja dari CNN	14
Gambar 2. 9 Perhitungan <i>Convolutional Layer</i>	15
Gambar 2. 10 Jenis-Jenis <i>Pooling</i>	16
Gambar 2. 13 Bentuk Kurva dari <i>ReLU Function</i>	17
Gambar 2. 14 <i>Fully Connected Layer</i>	18
Gambar 2. 15 Bentuk Lapisan VGG16	19
Gambar 3. 1 Diagram Ishikawa	22
Gambar 3. 2 Jumlah data masing-masing kelas untuk <i>training</i> dan <i>testing</i>	23
Gambar 3. 3 Beberapa Gambar Daun dari <i>Dataset Training</i>	23
Gambar 3. 4 Arsitektur Umum Sistem.....	25
Gambar 3. 5 <i>Use Case Diagram</i> Sistem.....	26
Gambar 3. 6 <i>Activity Diagram</i> Sistem.....	27
Gambar 3. 7 <i>Sequence Diagram</i> Sistem.....	28
Gambar 3. 8 <i>Flowchart</i> Pelatihan Model CNN.....	29
Gambar 3. 9 <i>Flowchart</i> Sistem.....	30
Gambar 3. 10 Rancangan Tampilan Beranda.....	31
Gambar 3. 11 Rancangan Tampilan Deteksi	32
Gambar 3. 12 Rancangan Tampilan Hasil Deteksi.....	33
Gambar 3. 13 Rancangan Tampilan Bantuan.....	34
Gambar 3. 14 Rancangan Tampilan Tentang	35
Gambar 4. 1 Halaman Beranda.....	36

Gambar 4. 2 Halaman Deteksi.....	37
Gambar 4. 3 Halaman Hasil Deteksi	38
Gambar 4. 4 Halaman Bantuan	39
Gambar 4. 5 Halaman Tentang.....	40
Gambar 4. 6 Kode Program Koneksi <i>Dataset</i>	40
Gambar 4. 7 Kode Program Akses Data <i>Train</i>	41
Gambar 4. 8 Lapisan pada Pelatihan Model.....	42
Gambar 4. 9 Kode Program Mulai Pelatihan	46
Gambar 4. 10 Proses Pelatihan Model.....	47
Gambar 4. 11 Hasil Akhir Pelatihan Model	48
Gambar 4. 12 Hasil <i>Confusion Matrix</i>	48
Gambar 4. 13 Grafik <i>Accuracy & Loss</i>	49
Gambar 4. 14 Memanggil <i>Library</i> yang dibutuhkan.....	52
Gambar 4. 15 Tahap Deklarasi (Insialisasi)	52
Gambar 4. 16 Fungsi Memanggil Model.....	53
Gambar 4. 17 Fungsi Utama Integrasi.....	53
Gambar 4. 18 Fungsi Mengirim Hasil dari Model	54
Gambar 4. 19 Kode HTML Menampilkan Hasil dari Model	54

DAFTAR LAMPIRAN

	Halaman
<i>Curriculum Vitae</i>	A-1

BAB 1

PENDAHULUAN

1.1. Latar Belakang Masalah

Mangga (*Mangifera Indica*) merupakan salah satu tanaman yang sangat populer dan disukai oleh masyarakat Indonesia. Banyak orang menyukai atau memelihara tanaman mangga dikarenakan memiliki buah nya yang enak, manis, kaya vitamin dan mineral yang menyehatkan tubuh, serta buah mangga dapat berubah menjadi sebuah inovasi makanan atau minuman seperti jus mangga, manisan mangga, buah kaleng mangga dll. (Nurmila, 2019).

Selain itu, mudahnya memelihara tanaman mangga menjadi alasan masyarakat biasa dan petani untuk menanam tanaman mangga. Tanaman mangga pun telah menyebar ke beberapa daerah di Indonesia. (Purbawesa, 2019).

Namun, ada beberapa faktor yang membuat masyarakat dan petani resah ketika memelihara atau membudidayakan tanaman mangga. Salah satu nya adalah adanya serangan hama, virus atau jamur pada tanaman mangga sehingga menghasilkan kondisi buah mangga yang tidak maksimal dan busuk, yang memberikan kerugian pada produktif buah. (Herdiyanto et al. 2019).

Tanaman mangga yang rusak atau buah yang tidak jadi dipanen merupakan efek dari serangan jamur dan virus, sehingga mengganggu produksi buah mangga. (Siregar, 2017 dalam Rohmatika, 2022)

Salah satu penyakit pada tanaman mangga adalah penyakit hama Kutu Putih dan penyakit Embun Jelaga. Kedua penyakit tersebut menimbulkan suatu gejala pada permukaan daun mangga.

Hama Kutu Putih (*Pseudococcus*) merupakan serangga hama pertanian yang paling mengerikan di daerah tropis (Herren, 1981) dan resiko yang meningkat pada buah mangga (Karar et al., 2006). Umumnya, hama kutu putih ditandai dengan banyaknya gumpalan benang lilin berwarna putih di permukaan daun, sehingga membuat permukaan daun menjadi seperti keputihan. Hama kutu putih merupakan penyebab dari tumbuhnya penyakit embun jelaga. (BPPSDMP, 2020).

Penyakit Embun Jelaga adalah sebuah penyakit yang disebabkan jamur *Capnodium sp.* Penyakit embun jelaga dapat ditandai dengan adanya lapisan warna hitam dan lengket, yang menutupi permukaan daun mangga akibat adanya Jamur *Capnodium sp.* Jamur tersebut tumbuh disebabkan adanya serangan dari serangga atau wereng yang mengisap gula/madu dan sisa-sisa gula/madu tersebut menempel pada permukaan daun, sehingga menjadi lokasi tumbuhnya jamur *Capnodium sp.* (R. Elqassas, 2018). Penyakit embun jelaga dapat menghambat kinerja fotosintesis pada tanaman mangga.

Berdasarkan penjelasan diatas, maka dibutuhkanlah sebuah sistem yang mampu mengidentifikasi jenis penyakit pada tanaman mangga dengan melakukan klasifikasi gambar penyakit pada daun mangga, sehingga masyarakat dan petani dapat mengetahui jenis penyakit yang diderita mangga dan cara menanggulangnya.

Convolutional Neural Network (CNN) merupakan sebuah metode dari algoritma *Deep Learning* yang dirancang untuk mengenal dan mendeteksi sebuah objek di dalam sebuah citra digital dan dapat digunakan untuk fitur Pengenalan Wajah (*Face Recognition*), Klasifikasi Gambar (*Image Classified*), dll. (Nisa, et al. 2021)

Metode CNN mengklasifikasikan gambar dengan memproses gambar yang di-input, lalu membagi gambar tersebut menjadi gambar yang lebih kecil namun dengan nilai konvolusi yang sama. Terakhir, menyimpan hasil dari gambar-gambar kecil tersebut ke dalam suatu *array* baru, yang kemudian digunakan oleh model untuk prediksi. (Herlambang, 2020).

Berdasarkan pemaparan dan penjelasan sebelumnya, maka dari itu, penulis bermaksud untuk membuat skripsi dengan judul “Klasifikasi Penyakit pada Daun Mangga dengan Menggunakan Metode Convolutional Neural Network (CNN)”.

1.2. Rumusan Masalah

Berdasarkan pemaparan latar belakang tersebut, masih banyak orang yang belum mengenali penyakit pada daun mangga secara umum, sehingga menghasilkan buah mangga yang tidak maksimal dan busuk. Maka, dibutuhkanlah suatu sistem yang dapat mengenali jenis penyakit yang ada di daun mangga.

1.3. Batasan Masalah

Adapun beberapa batasan masalah dalam penelitian ini adalah:

1. Data yang digunakan adalah berupa citra daun mangga.
2. Bahasa pemrograman yang digunakan dalam pembuatan sistem adalah Python dan HTML.
3. Menggunakan metode *Convolutional Neural Network*.
4. Sistem yang dirancang adalah sistem yang berbasis *Website*.
5. Kategori penyakit pada daun mangga yang digunakan dalam penelitian ini adalah Antraknosa, Embun Jelaga, *Gleosporium* dan Kutu Putih.

1.4. Tujuan Penelitian

Tujuan dari penelitian ini adalah menghasilkan sebuah sistem yang mampu melakukan klasifikasi penyakit pada daun mangga dengan metode *Convolutional Neural Network* (CNN), sehingga dapat mengidentifikasi jenis penyakit pada tanaman mangga.

1.5. Manfaat Penelitian

Manfaat dari Penelitian ini adalah dapat menghasilkan sebuah sistem yang mampu meng-klasifikasikan penyakit pada daun mangga dengan menggunakan metode *Convolutional Neural Network* (CNN), sehingga memudahkan masyarakat dan petani mangga untuk mengidentifikasi jenis penyakit pada daun mangga, dan sebagai bahan referensi bagi peneliti lain yang topiknya berkaitan dengan penelitian ini.

1.6. Metodologi Penelitian

Metode penelitian yang digunakan di penelitian ini adalah, sebagai berikut :

1. Studi Pustaka

Pada tahap ini, penulis mencari referensi dari berbagai sumber terpercaya dan melakukan peninjauan pustaka melalui buku, jurnal, *e-book*, artikel ilmiah, makalah ataupun situs internet yang berhubungan dengan tanaman mangga, penyakit pada tanaman mangga dan *Convolutional Neural Network* (CNN)

2. Analisa dan Perancangan

Pada analisis dan perancangan, penulis menganalisis hal-hal yang diperlukan dalam penelitian dan merancang bentuk sistem yang akan dibuat dalam bentuk sebuah diagram alir (*flowchart*).

3. Implementasi

Pada tahap ini, penulis melakukan pembuatan sistem dengan menggunakan bahasa Python, HTML dan CSS dengan sistem berbasis website, sesuai dengan rancangan diagram alir (*flowchart*).

4. Pengujian

Pada tahap pengujian ini, penulis melakukan uji coba terhadap sistem yang sudah dibuat.

5. Dokumentasi

Pada tahap dokumentasi, membuat dokumentasi terhadap semua kegiatan penelitian yang sudah dilaksanakan, dalam bentuk skripsi.

1.7. Sistematika Penelitian

Berikut merupakan sistematika penulisan skripsi yang terdiri dari beberapa bagian utama seperti yang dijelaskan sebagai berikut.

BAB 1 PENDAHULUAN

Bab ini berisi tentang latar belakang penelitian dari judul “Klasifikasi Penyakit pada Daun Mangga dengan Menggunakan Metode *Convolutional Neural Network* (CNN)”, rumusan

masalah, batasan masalah, tujuan penelitian, manfaat penelitian, metode penelitian dan sistematika penulisan.

BAB 2 LANDASAN TEORI

Bab ini merupakan pembahasan teori yang terkait dengan Tanaman Mangga dan Metode *Convolutional Neural Network* (CNN)".

BAB 3 ANALISIS DAN PERANCANGAN

Bab ini menjelaskan analisis masalah penelitian dan proses pembuatan sistem serta pembahasan.

BAB 4 IMPLEMENTASI DAN PENGUJIAN

Bab ini berisi tentang pembuatan sistem yang sudah dianalisis dan merancang sistem menggunakan bahasa pemograman HTML, dan CSS. Serta pembahasan tentang hal-hal yang dibutuhkan dalam pengujian sistem.

BAB 5 KESIMPULAN DAN SARAN

Bab ini berisikan kesimpulan dari keseluruhan hasil analisis dan pengujian penelitian serta pemberian saran sebagai masukan terhadap permasalahan yang didapat dengan harapan membantu pengembangan penelitian selanjutnya.

BAB 2

LANDASAN TEORI

2.1. Mangga

2.1.1. Deskripsi Mangga

Mangga (*Mangifera Indica L.*) adalah tanaman asli dari negara India.. Penemuan tanaman mangga pertama kali terjadi di Lembah Indus, India pada tahun 327 Sebelum Masehi. Hampir seluruh wilayah India mempunyai pohon manga. Kemudian mangga menyebar ke arah negara lain, seperti Malaysia, Indonesia, Bangladesh, Pakistan, Mesir dan sekitarnya. (Yuniati, 2019)

Tanaman mangga disebarkan oleh pedagang-pedagang India dan orang-orang yang menyebarkan agama Hindu dan Buddha di beberapa Negara pada abad ke-4 M. Di Indonesia, mangga pertama kali ditemukan di daerah timur Indonesia. Lebih tepatnya di daerah Kepulauan Maluku pada tahun 1665 dan mulai tersebar ke daerah-daerah lainnya. (Yuniati, 2019)

Mangga (*Mangifera Indica*) merupakan salah satu tanaman yang sangat populer dan disukai oleh masyarakat Indonesia. Banyak orang menyukai atau memelihara tanaman mangga dikarenakan memiliki buah nya yang enak, manis, kaya vitamin dan mineral yang menyehatkan tubuh, serta buah mangga dapat berubah menjadi sebuah inovasi makanan atau minuman seperti jus mangga, manisan mangga, buah kaleng mangga dll. (Nurmila, 2019)



(a)



(b)

Gambar 2. 1 (a) Daun dan Buah Mangga, (b) Pohon Mangga

2.1.2. Klasifikasi Tanaman Mangga

Berikut ini adalah klasifikasi mangga (*Mangifera Indica*) (Pratama, 2018) :

- Kingdom : *Plantae*
- Sub kingdom : *Tracheobionta*
- Superdivisi : *Spermatophyta*
- Divisi : *Magnoliophyta*
- Kelas : *Magnoliopsida*
- Subkelas : *Rosidae*
- Ordo : *Sapindales*
- Famili : *Anacardiaceae*
- Genus : *Mangifera*
- Spesies : *Mangifera indica*

2.1.3. Penyakit pada Tanaman Mangga

Penyakit tanaman adalah kondisi abnormal tanaman yang mengganggu pertumbuhan, produksi dan berkembangnya suatu tanaman, yang menimbulkan suatu ciri-ciri gejala, yang disebabkan oleh virus, jamur atau bakteri.

Penyakit tanaman mangga merupakan penyakit yang dapat menjadi penghalang besar untuk menghasilkan buah mangga yang cukup untuk memenuhi kebutuhan masyarakat. (Md. Rasel Mia et al.2019)

Beberapa jenis penyakit yang menyerang pada daun mangga antara lain:

a) Kutu Putih

Hama Kutu putih (*Pseudococcus*) merupakan serangga hama pertanian yang paling mengerikan di daerah tropis (Herren, 1981) dan menimbulkan resiko pada buah mangga (Karar et al., 2006). Saat menyedot cairan pada daun mangga, kutu menyuntikkan racun ke dalam jaringan tanaman. Akibatnya daun menjadi kecil dan keriput.

Hama kutu putih sering ditandai dengan adanya gumpalan benang lilin berwarna putih yang banyak di permukaan daun, sehingga membuat permukaan daun menjadi seperti keputihan. Hama kutu putih merupakan penyebab dari tumbuhnya penyakit embun jelaga. (BPPSDMP, 2020)



Gambar 2. 2 Kutu Putih

b) Embun Jelaga

Penyakit Embun Jelaga adalah sebuah penyakit yang disebabkan jamur *Capnodium sp.*. Jamur tersebut tumbuh disebabkan adanya serangan dari serangga atau wereng yang mengisap gula/madu dan sisa-sisa gula/madu tersebut menempel pada permukaan daun, sehingga menjadi lokasi tumbuhnya jamur *Capnodium sp.*. (R. Elqassas, 2018)

Penyakit embun jelaga dapat ditandai dengan adanya lapisan warna hitam dan lengket, yang menutupi permukaan daun mangga akibat adanya Jamur *Capnodium sp.*. Penyakit embun jelaga dapat menghambat kinerja fotosintesis pada tanaman mangga.



Gambar 2. 3 Embun Jelaga

c) Antraknosa

Kerugian pohon mangga di seluruh dunia disebabkan oleh penyakit Antraknosa, dengan nilai persentase 39% di seluruh dunia. Antraknosa dapat membuat penurunan nilai gizi dan kandungan antioksidan yang sehat pada buah mangga. (Sharma et al., 2020).

Antraknosa adalah suatu penyakit pada mangga yang diserang oleh jamur *Colletotrichum Gloeosporioides*. Gejala penyakit antraknosa pada daun mangga:

- Adanya bercak-bercak kecil pada daun yang kemudian akan berkembang menjadi bercak-bercak besar yang berwarna cokelat atau hitam. (Sharma et al., 2020).



Gambar 2. 4 Antraknosa

d) Penyakit *Gloeosporium*

Penyakit *Gloeosporium* berasal dari serangan jamur yang bernama *Gloeosporium Mangiferae*.

Gejala dari penyakit *Gloeosporium* pada tanaman mangga adalah sebagai berikut (Utami, 2017) :

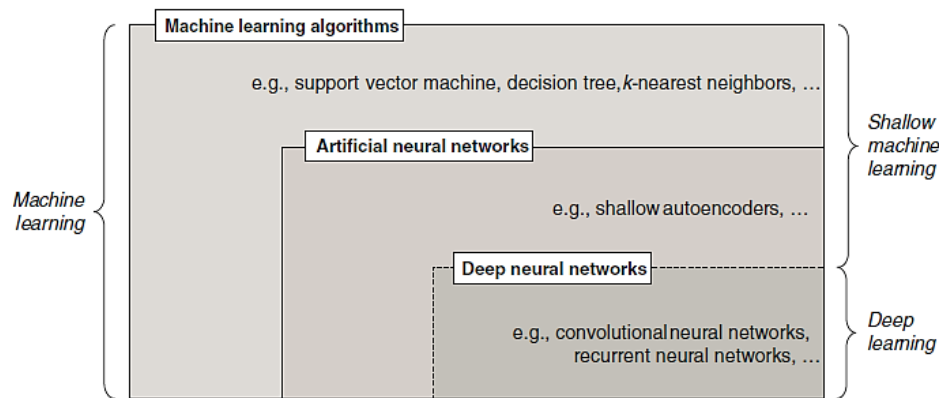
- Beberapa bunga mangga menjadi layu
- Buah mengalami bercak kecil berwarna coklat yang kemudian berkembang menjadi bercak-bercak besar berwarna coklat kehitaman
- Daunnya menghitam & menggulung hingga berlubang.



Gambar 2. 5 Penyakit *Gloeosporium*

2.2. Machine Learning

Machine Learning merupakan bagian dari *Artificial Intelligence*, yang memiliki kemampuan untuk belajar secara mandiri, dengan menggunakan data-data yang diberikan, untuk menemukan solusi dari suatu masalah, seperti masalah di *computer vision*, *face recognition*, pengenalan suara, robotika, *text mining*, *Natural Language Processing*, dll. *Machine Learning* terdiri atas 3 teknik dasar belajar yaitu *Supervised Learning*, *Unsupervised Learning*, dan *Reinforcement Learning* (Roihan, et. al 2020).



Gambar 2. 6 Kelas dari *Machine Learning* (Goodfellow et al. 2016)

2.2.1. Supervised Learning

Supervised learning melakukan pembelajaran menggunakan data telah yang diberi label, sampai mendapatkan kalkulasi baik untuk *output* yang diharapkan, sehingga dapat menemukan solusi yang tepat untuk pengguna.

Beberapa contoh dari *supervised learning* adalah klasifikasi gambar, deteksi suara, deteksi & diagnosa penyakit dan seterusnya. (Alzubaidi, et.al, 2021)

2.2.2. Unsupervised Learning

Unsupervised learning melakukan pembelajaran menggunakan data yang tidak diberi label, sampai menemukan struktur yang tidak diketahui di *input* (masukan). sehingga dapat menemukan solusi yang tepat untuk pengguna.

Salah satu contoh dari *Unsupervised learning* adalah pengelompokan (*clustering*). (Alzubaidi, et.al, 2021)

2.2.3. Reinforcement Learning

Reinforcement Learning melakukan pembelajaran dengan berinteraksi dengan lingkungan dan mengamati responsnya dalam bertindak. Hal tersebut berguna dapat meningkatkan kinerja agen ke depannya dan mencapai tujuan yang diinginkan.

Salah satu contoh dari *reinforcement learning* adalah robotika, *video game*, sistem kemudi otomatis dll. (Alzubaidi, et.al, 2021)

2.3. Deep Learning

Deep Learning adalah bentuk bagian dari *Supervised Learning*, adalah suatu algoritma yang mampu mengekstraksi fitur secara otomatis. *Deep Learning* telah mencapai hasil yang mengesankan dalam melakukan *image recognition* (pengenalan gambar) dan hingga saat ini fitur pengenalan gambar masih digunakan pada sistem kemudi otomatis. (H. Fujiyoshi et al. 2019)

Deep Learning merupakan pilihan terbaik untuk bidang-bidang seperti prediksi gambar, prediksi video, pengenalan suara, *Computer Vision*. dan lainnya (Ghimire et al, 2022).

Forward propagation dan *backward propagation* adalah dua tahap yang penting dalam proses pelatihan *deep learning neural network*.

Forward propagation melakukan pemrosesan data, dari lapisan input ke lapisan output. Sedangkan, *backward propagation* melakukan penyesuaian data, dari lapisan *output* ke lapisan *input*.

Hasil dari proses *forward propagation* digunakan untuk menghitung kesalahan pada *output layer*, dan kemudian digunakan sebagai panduan dalam proses *backward propagation* atau *backpropagation*, untuk menyesuaikan bobot dan bias di setiap layer agar dapat menghasilkan output yang lebih baik dan meminimalisir kesalahan atau *error*.

Terdapat beberapa jenis *neural network*, yaitu *Convolutional Neural Network (CNN)*, *Recurrent Neural Network (RNN)*, *Artificial Neural Network (ANN)*, dan sebagainya.

2.4. Convolutional Neural Network

Convolutional Neural Network (CNN) adalah sebuah metode dari *Deep Learning* yang berfungsi mengenali atau mendeteksi suatu objek di dalam sebuah citra digital dan dapat digunakan untuk fitur Pengenalan Wajah (*Face Recognition*), Klasifikasi Gambar (*Image Classified*), dll. (Nisa, et al. 2021)

CNN melakukan proses klasifikasi gambar dengan cara mengolah gambar yang di-*input*, lalu memecah gambar tersebut menjadi gambar yang lebih kecil namun dengan nilai konvolusi yang sama. Terakhir, menyimpan hasil dari gambar-gambar kecil tersebut menjadi sebuah *array* baru, yang kemudian akan digunakan oleh model untuk prediksi. (Herlambang, 2020)

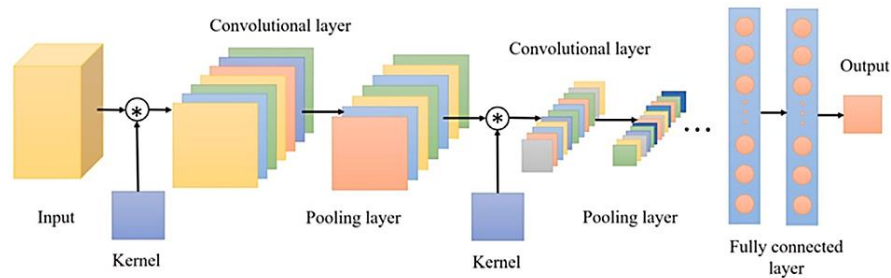
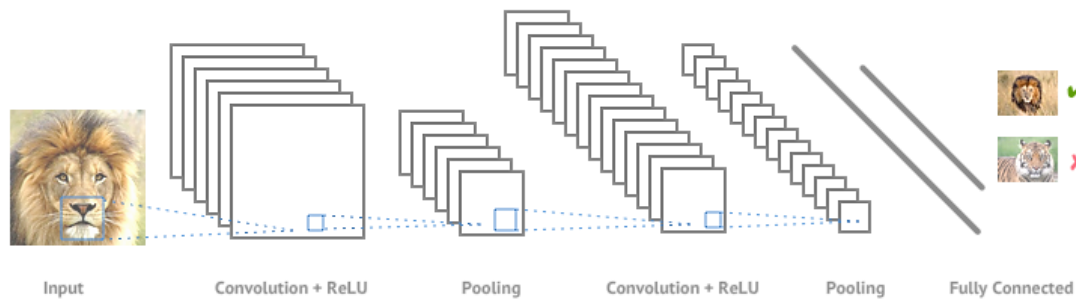


Fig. 3. A basic architecture of a Convolutional Neural Network (CNN) model.

Gambar 2. 7 Arsitektur dari CNN (Ghimire et al., 2022)

Keunggulan dari penggunaan CNN dibandingkan dengan *neural network* lainnya adalah :

- CNN mengkombinasikan proses ekstraksi fitur dan klasifikasi fitur saat pelatihan berlangsung, sehingga mampu mengoptimalkan fitur-fitur pada data.
- CNN mampu mengetahui pola-pola yang kompleks melalui ekstraksi fitur gambar, dikarenakan menggunakan operasi konvolusi.
- CNN mampu memproses gambar yang memiliki ukuran yang beragam.
- CNN mampu memproses *input* yang besar, dengan tingkat komputasi lebih cepat.



Gambar 2. 8 Cara Kerja dari CNN

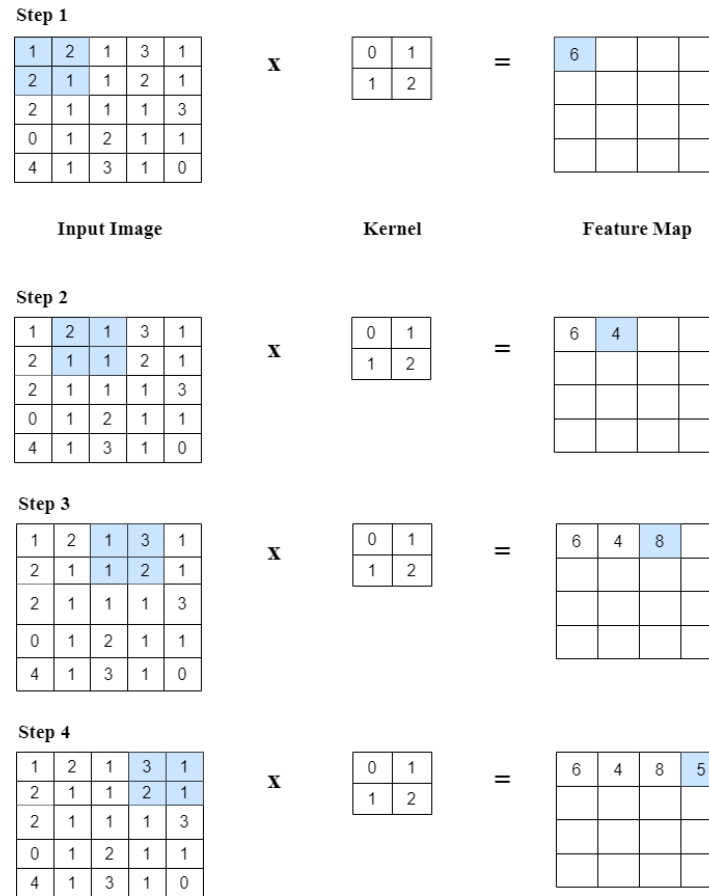
2.4.1. CNN Layer

Arsitektur CNN terdiri dari beberapa lapisan (*layer*) dengan fungsi yang berbeda-beda. Berikut adalah lapisan yang ada di arsitektur CNN:

2.4.1.1. Convolutional Layer

Bagian utama dari CNN adalah *Convolutional Layer*, yang digunakan untuk mendeteksi keberadaan fitur dalam gambar. Fitur dapat berupa apa saja, mulai dari tepi dan kurva sederhana hingga struktur yang lebih kompleks seperti telinga, hidung, mata, dll.

Convolutional Layer terdiri atas beberapa *convolutional filters* (kernel). Kernel memiliki nilai diskrit yang disebut bobot kernel. Kernel berfungsi untuk mengekstrak fitur dari gambar pada saat pelatihan dilaksanakan.



Gambar 2.9 Perhitungan *Convolutional Layer*

Perhitungan manual dari *Convolutional Layer* sebagai berikut:

a. **Step 1**

$$\begin{array}{lcl} \text{Pixel} & = & 1 \ 2 \ 2 \ 1 \\ \text{Kernel} & = & 0 \ 1 \ 1 \ 2 \end{array}$$

$$\begin{array}{l} \text{Feature Map} = (1 \times 0) + (2 \times 1) + (2 \times 1) + (1 \times 2) \\ \text{Feature Map} = 0 + 2 + 2 + 2 \\ \text{Feature Map} = \mathbf{6} \end{array}$$

b. **Step 2**

$$\begin{array}{lcl} \text{Pixel} & = & 2 \ 1 \ 1 \ 1 \\ \text{Kernel} & = & 0 \ 1 \ 1 \ 2 \end{array}$$

$$\begin{array}{l} \text{Feature Map} = (2 \times 0) + (1 \times 1) + (1 \times 1) + (1 \times 2) \\ \text{Feature Map} = 0 + 1 + 1 + 2 \\ \text{Feature Map} = \mathbf{4} \end{array}$$

c. **Step 3**

$$\begin{array}{lcl} \text{Pixel} & = & 1 \ 3 \ 1 \ 2 \\ \text{Kernel} & = & 0 \ 1 \ 1 \ 2 \end{array}$$

$$\begin{array}{l} \text{Feature Map} = (1 \times 0) + (3 \times 1) + (1 \times 1) + (2 \times 2) \\ \text{Feature Map} = 0 + 3 + 1 + 4 \\ \text{Feature Map} = \mathbf{8} \end{array}$$

d. **Step 4**

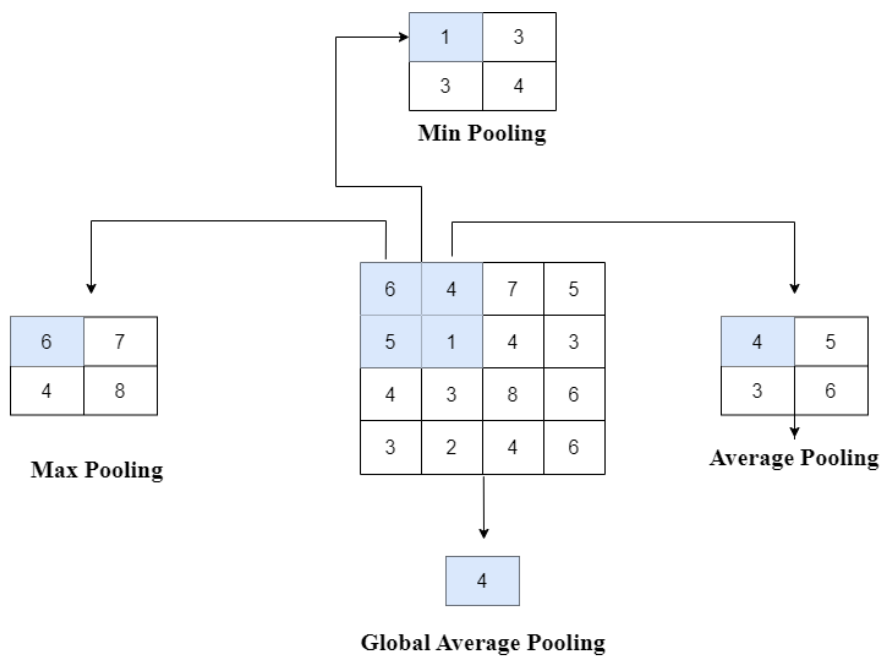
$$\begin{array}{lcl} \text{Pixel} & = & 3 \ 1 \ 2 \ 1 \\ \text{Kernel} & = & 0 \ 1 \ 1 \ 2 \end{array}$$

$$\begin{array}{l} \text{Feature Map} = (3 \times 0) + (1 \times 1) + (2 \times 1) + (1 \times 2) \\ \text{Feature Map} = 0 + 1 + 2 + 2 \\ \text{Feature Map} = \mathbf{5} \end{array}$$

2.4.1.2. Pooling Layer

Pooling layer merupakan salah satu tahapan layer CNN yang umumnya digunakan untuk mengurangi ruang dimensi dari *output* setelah *convolutional layer*. Tujuannya adalah mengurangi jumlah nilai yang diperlukan, sehingga mempercepat proses pelatihan dan mengurangi kejadian *overfitting*.

Beberapa *pooling* yang biasa dipakai dalam *pooling layer* adalah *Min Pooling*, *Max Pooling*, dan *Global Average Pooling* (GAP). (Alzubaidi et al., 2021)



Gambar 2. 10 Jenis-Jenis *Pooling*

Perhitungan manual dari *Pooling Layer* sebagai berikut:

Pixel = 6, 4, 5, 1 ; Filter = 2x2 ; Stride = 2 pixel

a) <i>Min Pooling</i> (Nilai Terendah) Min Pooling = 1	b) <i>Max Pooling</i> (Nilai Tertinggi) Max Pooling = 6	c) <i>Average Pooling</i> (Nilai Rata-Rata) $\text{Average Pooling} = (6+4+5+1)/4$ $\text{Average Pooling} = 16/4$ Average Pooling = 4
--	---	--

2.4.1.3. Activation Function

Activation Function merupakan metode matematis yang menghidupkan atau mematikan *neuron* pada *output* di jaringan syaraf. *Activation Function* juga berfungsi membuat mesin mampu mempelajari pola data dan membuat model, dari kumpulan *dataset* yang kompleks seperti gambar, video, audio, ucapan, teks, dll. (Sharma et. al, 2020)

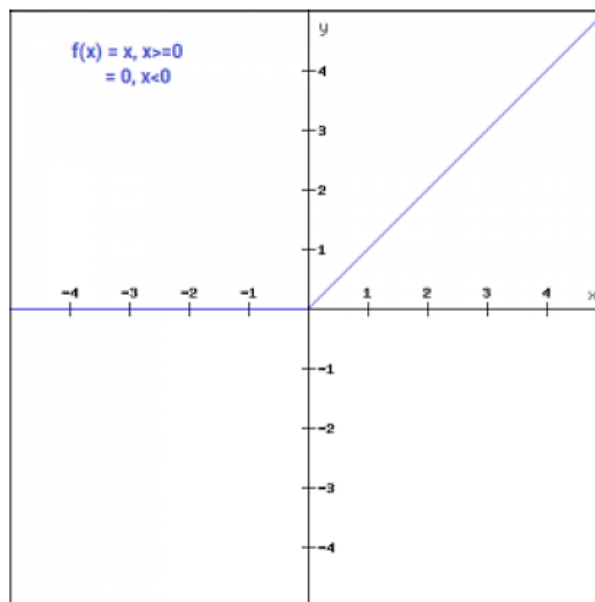
Salah satu jenis *activation function* yang digunakan dalam jaringan saraf tiruan adalah :

2.4.1.3.1. ReLU Function

ReLU (*Rectified Linear Unit*) merupakan fungsi aktivasi non-linear yang banyak digunakan dalam jaringan saraf. Fungsi ReLU lebih efisien daripada fungsi lainnya karena semua neuron tidak diaktifkan pada saat yang sama, melainkan sejumlah neuron diaktifkan pada waktu tertentu. (Sharma et. al, 2020)

Rumus matematika fungsi ReLU adalah sebagai berikut:

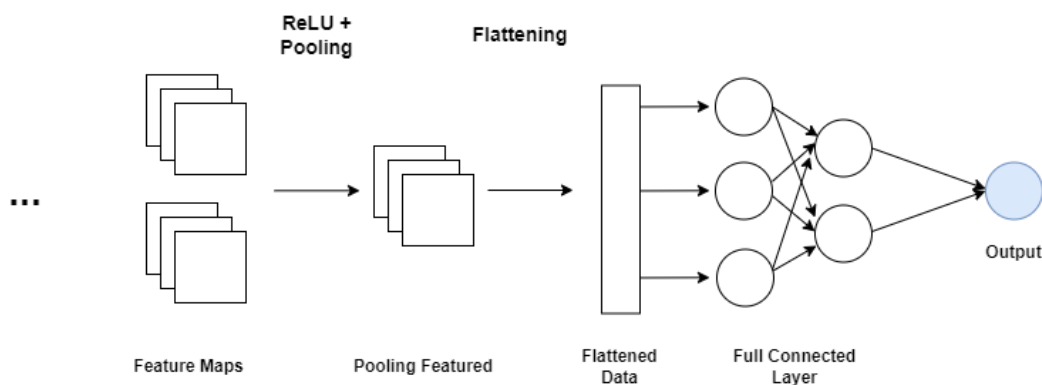
$$F(x) = \text{Max}(0, x) \quad (3)$$



Gambar 2. 11 Bentuk Kurva dari *ReLU Function*

2.4.1.4. Fully Connected Layer

Fully Connected Layer merupakan layer yang berada di urutan terakhir dalam arsitektur CNN, yang berfungsi menghubungkan setiap neuron dalam satu layer dengan setiap neuron dalam layer berikutnya, untuk menentukan hasil *output* yang akan ditampilkan. *Input* dari *Fully Connected Layer* berasal dari layer-layer sebelumnya, yaitu *Pooling Layer* ataupun *Convolutional Layer*, berupa vektor yang terbentuk dari feature maps setelah *flattening*. (Alzubaidi et al., 2021)



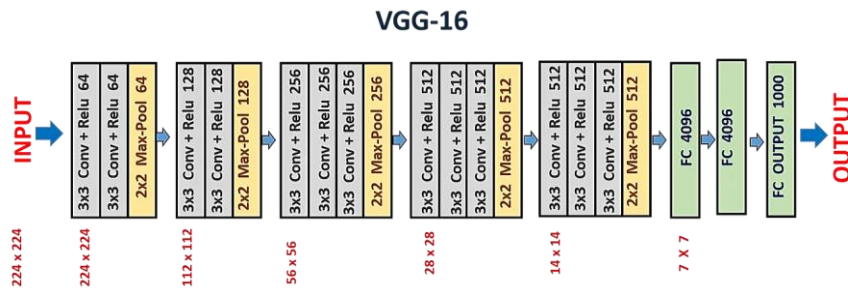
Gambar 2. 12 *Fully Connected Layer*

2.4.1.5. Loss Function

Loss Function berfungsi melakukan perhitungan kesalahan prediksi yang ada di seluruh sampel pelatihan dalam model CNN. Kesalahan ini menunjukkan perbedaan antara hasil prediksi dengan hasil kenyataan. Contoh *Loss Function* adalah *Euclidean Loss Function* dan *Hinge Loss Function*.

2.4.2. Visual Geometry Group (VGG) 16

Visual Geometry Group (VGG) 16 merupakan salah satu arsitektur dari CNN yang cukup terkenal digunakan untuk melakukan klasifikasi gambar. VGG-16 dirancang oleh Karen Simonyan dan Andrew Zisserman, dari *Oxford University* pada tahun 2014.



Gambar 2. 13 Bentuk Lapisan VGG16

Arsitektur VGG-16 memiliki 16 lapisan, yang terdiri atas 13 lapisan konvolusi (*convolution*), 2 lapisan *fully connected* + 1 lapisan keluaran (*output*). Menurut gambar 2.15, pada lapisan *input*, menerima gambar tetap 224x224 *pixel* dengan tiga saluran RGB. Lalu pada lapisan konvolusi, lapisan tersebut memiliki kernel (filter) 3x3 dan diikuti dengan *Max Pooling*. Pada lapisan *output*, terdapat lapisan *softmax* menghasilkan probabilitas sehingga hasil tersebut akan menentukan hasil prediksi kelak. (Albashish, et.al, 2021)

Berikut merupakan salah satu dari hyperparameter utama dari VGG-16 :

- a) *Filter* memiliki ukuran 3x3 di semua lapisan konvolusi.
- b) Lapisan pertama mempunyai 64 filter dan lapisan berikutnya mempunyai 128, 256, 512 dst.
- c) *Strides* dibuat sebanyak 1 pixel.

Arsitektur VGG-16 memiliki keunggulan dibandingkan arsitektur lainnya, dalam dunia klasifikasi gambar, yaitu sebagai berikut:

1. VGG-16 memiliki kemampuan ekstraksi fitur-fitur penting dengan bertahap secara otomatis.
2. VGG-16 menghasilkan akurasi yang tinggi dalam melakukan klasifikasi gambar.
3. VGG-16 memiliki *filter* 3x3 dengan padding yang sama, sehingga dapat melakukan pengulangan dengan mudah
4. VGG-16 menghasilkan model dengan performa yang sangat baik

Adapun arsitektur VGG-16 memiliki beberapa kelemahan, yaitu proses pelatihan yang sangat lama dan memiliki ukuran yang sangat besar.

2.5. Penelitian Relevan

Beberapa penelitian yang relevan dengan penelitian ini antara lain:

1. Menurut penelitian Abdul Kholik (2021), yang berjudul “Klasifikasi Menggunakan Convolutional Neural Network (CNN) pada Tangkapan Layar Halaman Instagram”, Training dilakukan dengan data gambar berukuran 150 x 150 dengan *epochs* sebanyak 25 menghasilkan tingkat nilai *Accuracy* sebesar 91%. Pengujian dilakukan dengan 25 data citra dan model yang dibangun berhasil memprediksi 20 data citra.
2. Penelitian dari Arrofiqoh & Harintaka (2018) yang berjudul “Implementasi Metode Convolutional Neural Network untuk Klasifikasi Tanaman Pada Citra Resolusi Tinggi”. Proses *training* menghasilkan akurasi 100 untuk kategori 5 kelas, dengan 10 gambar per kelasnya. Terakhir, proses validasi sebesar 93% dan proses *testing* sebesar 82%.
3. Berdasarkan dari penelitian Felix et al. (2020) yang berjudul “Implementasi Convolutional Neural Network Untuk Identifikasi Jenis Tanaman Melalui Daun” dengan *dataset* 375 gambar daun (250 data *training* dan 125 data *testing*). Penelitian ini menghasilkan besar akurasi 76%. Percobaan dilakukan dengan total uji coba 37 pengujian, 28 pengujian yang menghasilkan jawaban yang akurat dan 9 pengujian yang tidak akurat.
4. Menurut dari penelitian Salsabila et al. (2021) yaitu “Image Classification of Beef and Pork Using Convolutional Neural Network in Keras Framework”, Hasil klasifikasi terbaik citra daging babi dan daging sapi diperoleh dari model yang menggunakan regularisasi dropout dengan $p = 0,7$ karena memiliki *Accurate* sebesar 97%, data *Loss* sebesar 11% , AUC sebesar 99% dan *F1-Score* sebesar 97,5%. Berdasarkan model tersebut, dari 450 data yang diuji, terdapat 218 citra daging babi diklasifikasikan dengan benar dan 221 citra daging sapi diklasifikasikan dengan benar.

5. Penelitian dari Njayou Youssouf (2022), yang berjudul “Traffic sign classification using CNN and detection using faster-RCNN and YOLOV4”, Dalam penelitian tersebut ini, menggunakan CNN yang dapat mengklasifikasikan 43 rambu lalu lintas yang berbeda dari *dataset benchmark German Traffic Sign Recognition*. *Dataset* terdiri dari 39.186 gambar untuk pelatihan dan 12.630 untuk pengujian. CNN dilakukan untuk klasifikasi ringan dan mencapai akurasi 99,20% dengan parameter hanya 0,8 M.

BAB 3

ANALISIS DAN PERANCANGAN

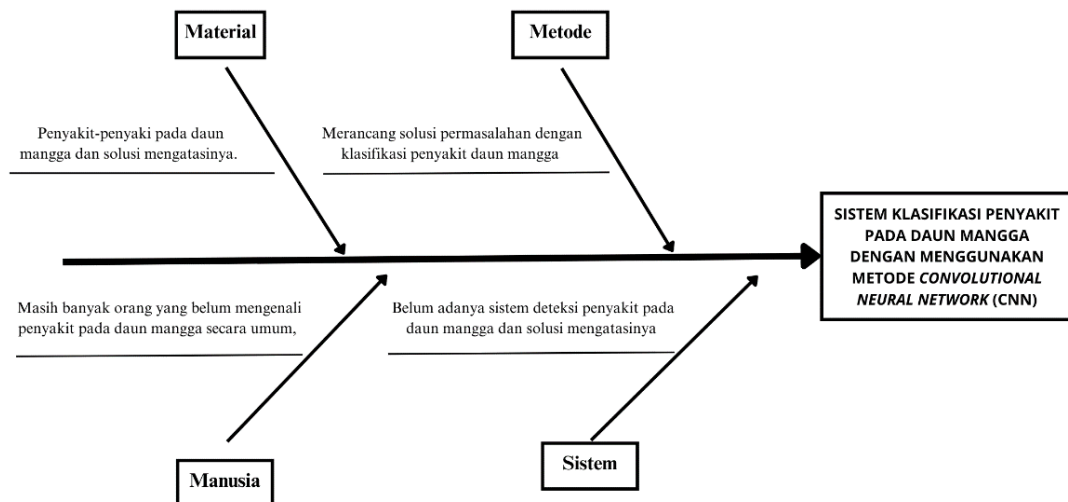
3. 1. Analisis Sistem

Analisis sistem ialah sebuah proses mencari dan menemukan masalah serta kebutuhan secara seksama, yang mana hasil dari analisis tersebut akan digunakan untuk merancang sistem yang efektif dan akurat, sesuai dengan tujuan pembuatan sistem.

3.1.1. Analisis Masalah

Analisis masalah adalah suatu proses analisis identifikasi dan memahami sebuah masalah, serta mencari data-data yang terkait dengan permasalahan tersebut, supaya masalah tersebut dapat diselesaikan. Permasalahan dari penelitian ini adalah masih banyak orang yang belum mengenali penyakit pada daun mangga secara umum, sehingga tanaman mangga menghasilkan buah mangga yang tidak maksimal dan busuk.

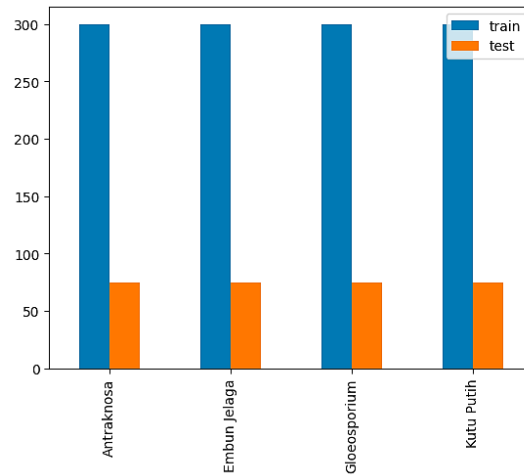
Maka, dari itu, dibutuhkanlah sebuah sistem yang dapat mengenali jenis penyakit yang ada di daun mangga, yang ditampilkan ke dalam diagram Ishikawa (*Fishbone Diagram*) pada gambar 3. 1



Gambar 3. 1 Diagram Ishikawa

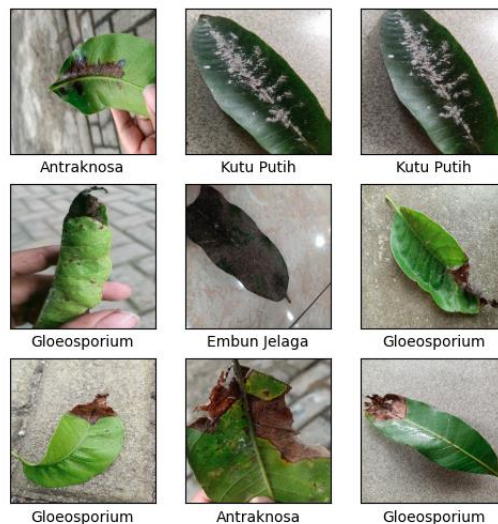
3.1.2. Analisis Data

Penelitian ini menggunakan data sebanyak 1500 data, yang terdiri atas 1200 data setiap kelas untuk fase *training* dan 300 data untuk fase *testing*, yang mana data-data tersebut diambil dari observasi secara langsung ke lapangan dan *dataset* dari *kaggle*, dengan perbandingan antara data *training* dengan data *testing* sebesar 80% : 20%. Detail jumlah data dapat dilihat di gambar 3.2 :



Gambar 3. 2 Jumlah data masing-masing kelas untuk *training* dan *testing*

Di bawah ini merupakan beberapa gambar yang diambil dari *dataset* untuk digunakan sebagai bahan *training* (pelatihan).



Gambar 3. 3 Beberapa Gambar Daun dari *Training Dataset*

3.1.3. Analisis Kebutuhan

Analisis kebutuhan adalah sebuah analisis yang dilakukan untuk menemukan segala hal yang dibutuhkan untuk sistem, sehingga diharapkan sistem mampu memenuhi harapan pengguna, menyelesaikan masalah yang dihadapi dan memenuhi tujuan utama sistem.

Analisis kebutuhan terdiri atas dua bagian, yaitu :

3.1.3.1. Kebutuhan Fungsional

Kebutuhan fungsional adalah seluruh proses yang mesti dikerjakan dan dimiliki oleh sistem, agar dapat berfungsi dengan efektif dan akurat. Kebutuhan fungsional dari rancangan sistem adalah :

- a. Sistem mampu melakukan klasifikasi penyakit daun mangga melalui gambar yang telah diinput dengan menggunakan metode *Convolutional Neural Network* (CNN).
- b. Sistem mampu menampilkan hasil dari klasifikasi tersebut.

3.1.3.2. Kebutuhan Non-Fungsional

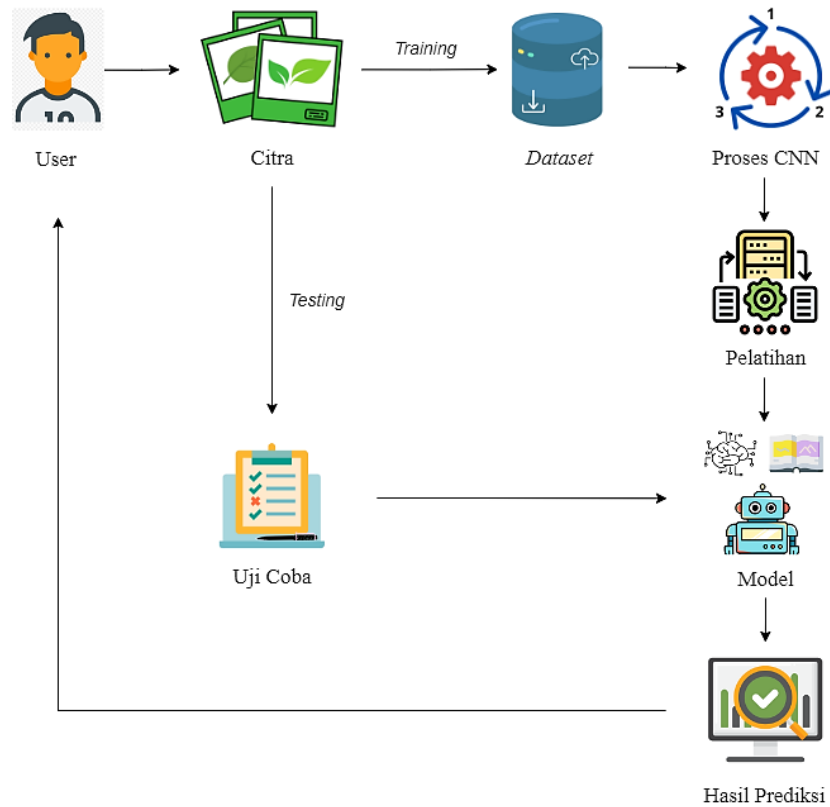
Kebutuhan non-fungsional adalah kebutuhan tambahan dari sistem seperti fitur-fitur dari sebuah sistem, yang tidak berhubungan dengan fungsi utama sistem. Terdapat beberapa kebutuhan non-fungsional dari sistem ini, yakni :

- a. Sistem memiliki fitur bantuan, berupa tata cara penggunaan sistem dengan tujuan memudahkan pengguna menggunakan sistem.
- b. Sistem memiliki tampilan yang sederhana dan ramah (*friendly*).
- c. Sistem bisa digunakan secara bebas dan tanpa adanya perangkat tambahan.

3.2 Arsitektur Umum

Arsitektur umum adalah bentuk susunan, rancangan dan desain dari sistem yang hendak dibuat.

Rancangan arsitektur umum dari sistem, sebagai berikut :



Gambar 3. 4 Arsitektur Umum Sistem

Berikut adalah penjelasan alur tahapan dari rangkaian arsitektur umum :

1. Pertama, pengguna mengumpulkan *dataset* berupa kumpulan citra/gambar daun mangga yang terkena penyakit.
2. Kemudian, *dataset* akan melewati beberapa proses seperti *cleaning*, *labelling*, *preprocessing*, dan *dataset* dibagi menjadi 3 yaitu *train*, *test*, dan *validation*.
3. Selanjutnya, melatih (*training*) model dengan *Convolutional Neural Network* (CNN) menggunakan *dataset* yang sudah diproses.
4. Setelah mendapatkan model dengan akurasi yang tinggi dan memuaskan, maka pengguna dapat melakukan pengujian (*testing*) dengan menggunakan Citra daun mangga yang terkena penyakit.

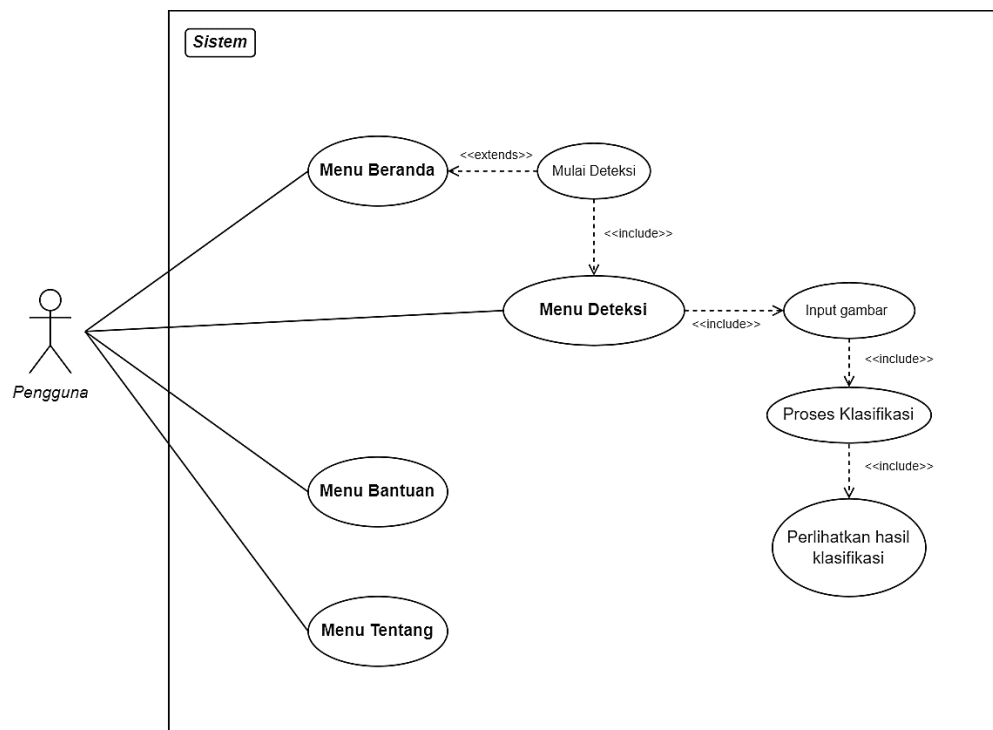
5. Hasil prediksi telah ditemukan.

3.3. Pemodelan Sistem

Pemodelan Sistem adalah bentuk representasi dari sistem yang akan dibuat. Dalam konteks penelitian ini, bentuk pemodelan sistem akan ditampilkan dan dijelaskan menggunakan beberapa diagram, yaitu *use case diagram*, *activity diagram*, dan *sequence diagram*.

3.3.1. Use Case Diagram

Use Case Diagram berfungsi menampilkan gambaran konsep rancangan sistem antara interaksi pengguna (*actor*) dan sistem yang dirancang.



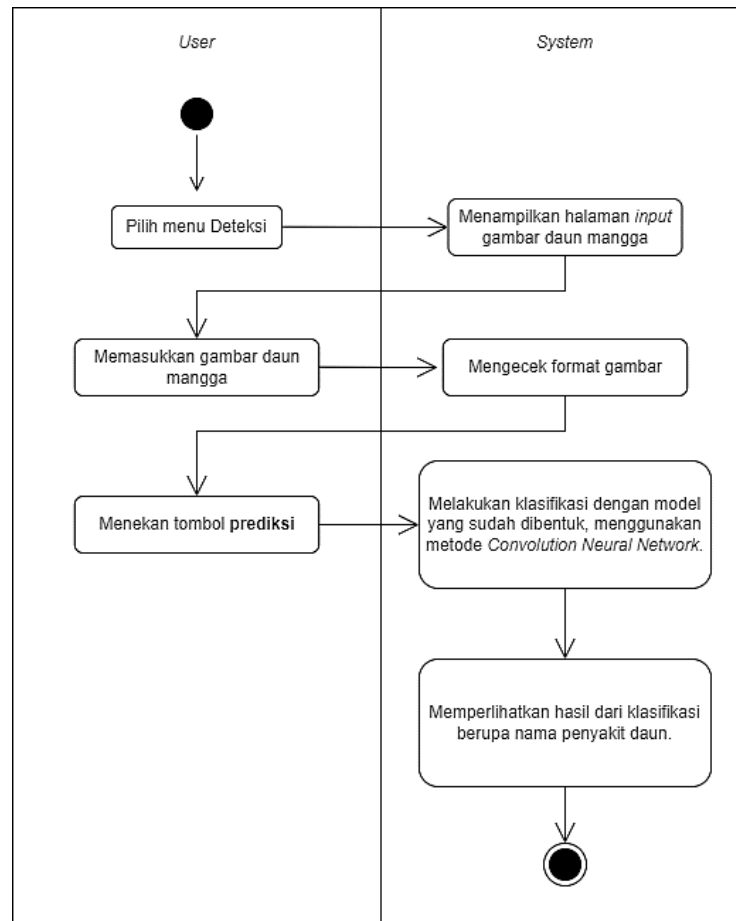
Gambar 3.5 Use Case Diagram Sistem

Menurut gambar 3.5, alur diagram dimulai dengan pengguna membuka *website* nya dahulu. Lalu sistem akan mengarahkan ke menu Beranda. Setelah itu, pengguna memilih fungsi “mulai deteksi” dan sistem akan menuju ke menu deteksi. Di dalam menu deteksi, pengguna memasukkan gambar daun mangga yang hendak diklasifikasi. Setelah itu, sistem akan melakukan proses klasifikasi dengan model yang sudah

dibentuk, menggunakan metode *Convolution Neural Network*. Terakhir, sistem akan memperlihatkan hasil dari klasifikasi berupa nama penyakit daunnya.

3.3.2. Activity Diagram

Activity Diagram berfungsi menampilkan alur kegiatan rancangan sistem, dari awal mulai, proses dan selesai.

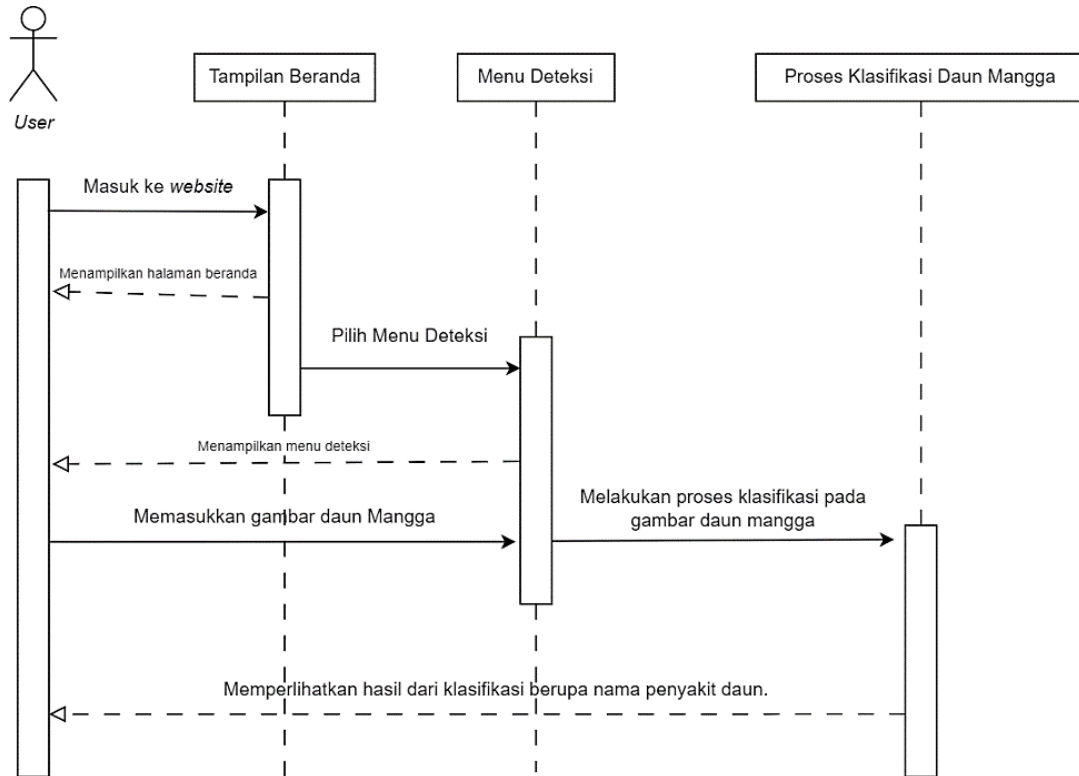


Gambar 3. 6 Activity Diagram Sistem

Menurut gambar 3.6, alur diagram dimulai dengan pengguna memilih menu Deteksi, lalu website akan memperlihatkan tampilan *input* gambar daun mangga. Selanjutnya *pengguna* memasukkan file gambar yang ingin diklasifikasi, lalu menekan tombol “Prediksi” agar sistem berjalan melakukan proses klasifikasi. Terakhir sistem memperlihatkan hasil dari klasifikasi berupa nama penyakit daunnya.

3.3.3. Sequence Diagram

Sequence Diagram berfungsi menampilkan susunan objek-objek, yang saling berinteraksi secara teratur untuk menyelesaikan masalah. *Sequence Diagram* dapat dilihat di bawah ini.



Gambar 3. 7 *Sequence Diagram* Sistem

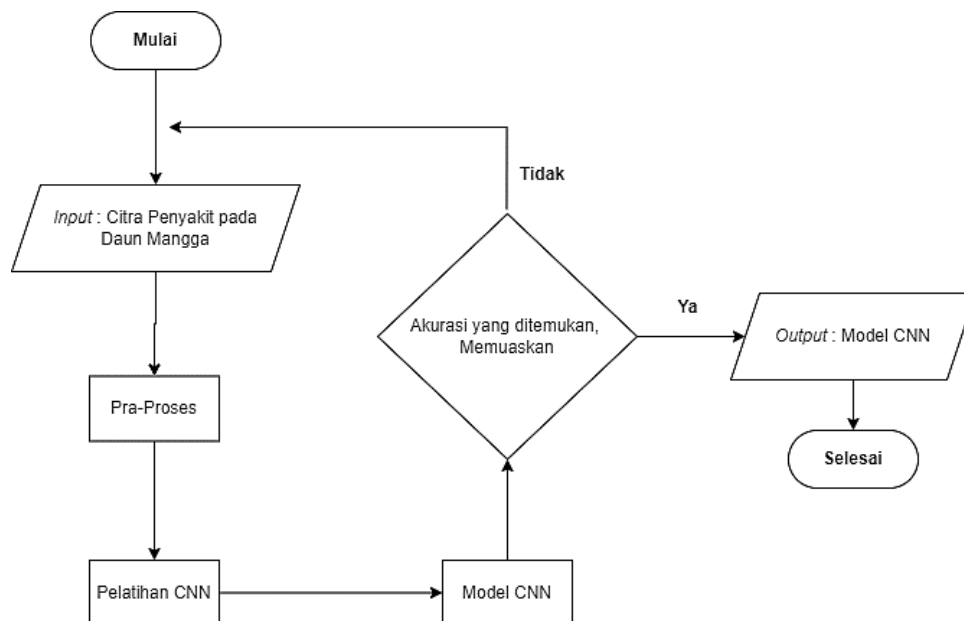
3.3.4. Flowchart (Diagram Alir)

Diagram alir (*flowchart*) adalah representasi dari alur dan hasil proses dari suatu sistem. Tujuan utama perancangan *flowchart* adalah untuk menggambarkan fase solusi dari suatu masalah secara jelas, singkat, mudah dipahami, dan berurut, dengan menggunakan simbol-simbol yang umum.

3.3.4.1. Flowchart Pelatihan Model CNN

Flowchart Training Model CNN dimulai dengan meng-input citra daun mangga. Setelah itu, citra tersebut memasuki tahap pra-proses, untuk *image resizing*, *image labelling*, *image augmentation* dan *data cleaning*. Setelah selesai

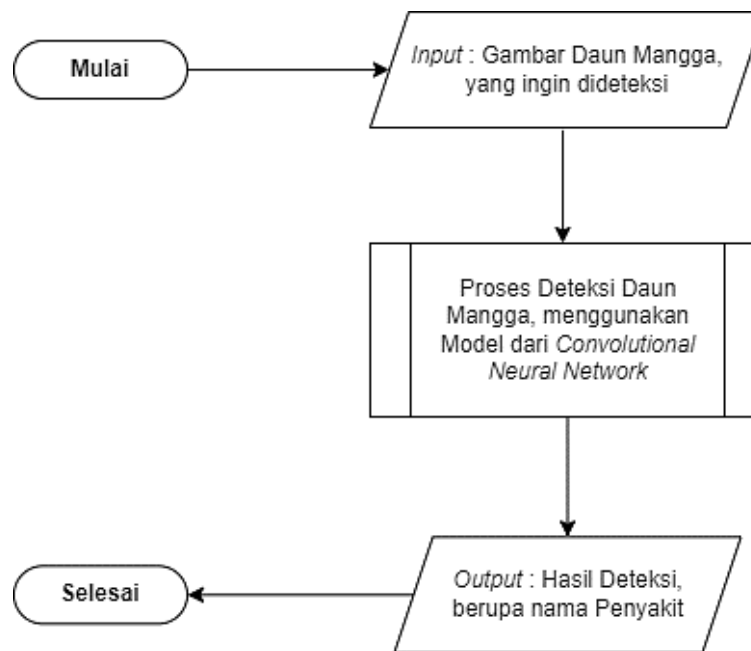
pra-proses, *dataset* dilatih dengan metode *Convolutional Neural Network* (CNN) sampai menghasilkan sebuah model berbasis CNN. Jika pengguna sudah puas terhadap hasil akurasi yang didapat, maka model CNN dapat disimpan dan siap untuk digunakan. Namun jika pengguna belum puas terhadap hasil akurasi yang didapat, maka pengguna dapat mengulangi pelatihan sampai hasil akurasinya memuaskan.



Gambar 3. 8 Flowchart Pelatihan Model CNN

3.3.4.1. Flowchart Sistem

Flowchart Sistem dimulai dengan meng-*input* citra daun mangga. Setelah itu, citra tersebut akan memasuki tahap pra-proses. Setelah selesai pra-proses, sistem memanggil Model CNN yang sudah disimpan dan *Label Map* untuk memprediksi gambar. Terakhir, sistem menampilkan hasil prediksi dari citra tersebut, berupa teks.



Gambar 3. 9 *Flowchart* Sistem

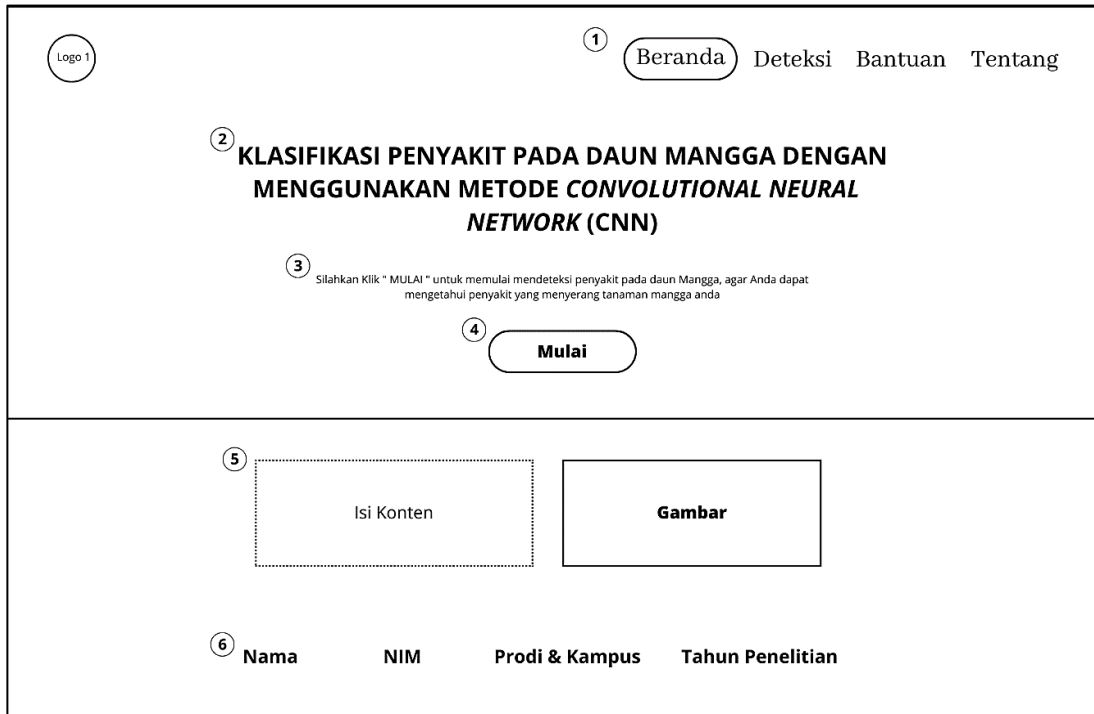
3.4 Perancangan Antarmuka

Antarmuka atau yang lebih dikenal dengan sebutan *Interface* telah menjadi bagian pada sistem yang memiliki fungsi sebagai media interaksi antara sistem dengan pengguna. Dalam melakukan rancangan *interface* diperlukan gambaran struktur pada sistem yang jelas terkait interface atau antarmuka yang akan dibangun sebelum melanjutkan ke tahap pelaksanaan atau implementasi.

Rancangan interface sistem diperlukan tampilan yang sederhana, hal ini berguna agar sistem memberikan kenyamanan serta kemudahan kepada para pengguna

3.4.1. Desain Halaman Beranda

Halaman Beranda merupakan tampilan utama sistem, yang berisikan beberapa informasi seperti judul penelitian, tanaman mangga, nama, NIM, program studi dan tahun penelitian.



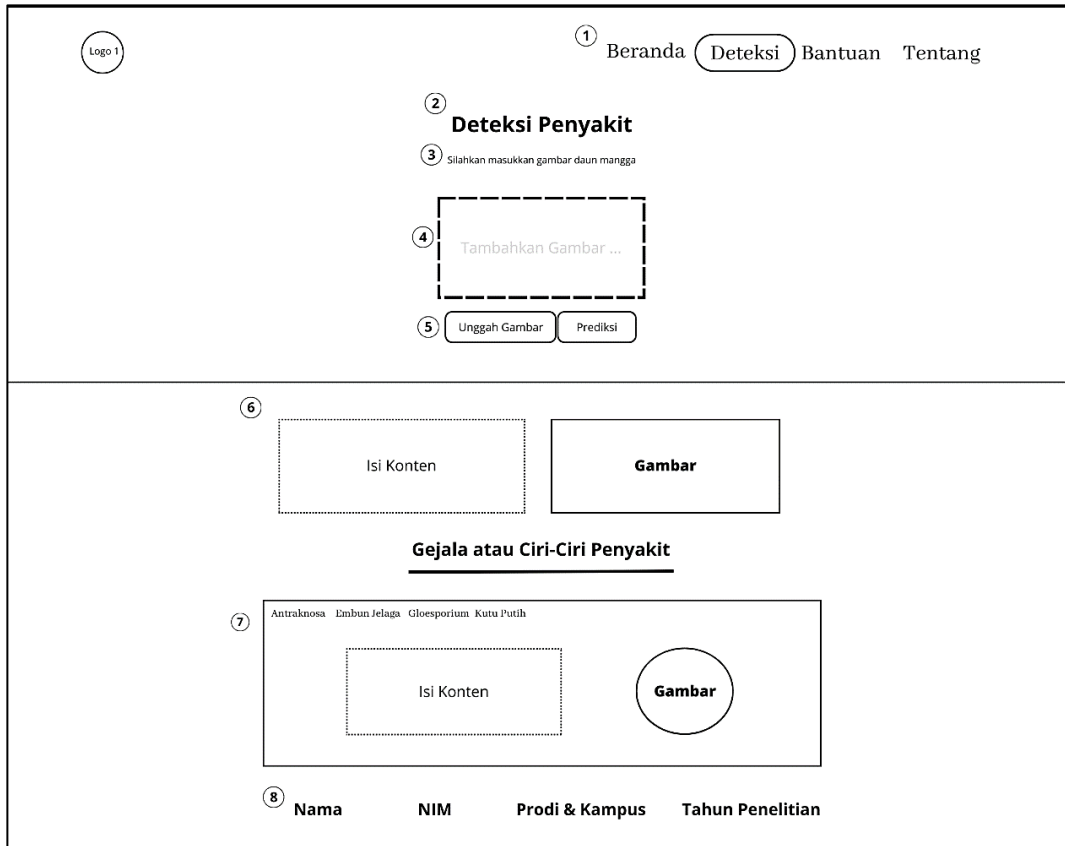
Gambar 3. 10 Rancangan Tampilan Beranda

Keterangan :

1. *Menustrip*, berfungsi menampilkan daftar menu
2. *Label*, berfungsi menampilkan judul menu
3. *Label*, berfungsi menampilkan penjelasan fungsi dari menu
4. *Button*, "Mulai" untuk menuju menu deteksi
5. *Label*, berfungsi menampilkan isi konten.
6. *Label*, berfungsi menampilkan Nama, NIM, Program Studi & Kampus dan Tahun Penelitian

3.4.2. Desain Halaman Deteksi

Halaman Deteksi merupakan halaman untuk melakukan fungsi klasifikasi dari gambar yang hendak dideteksi.

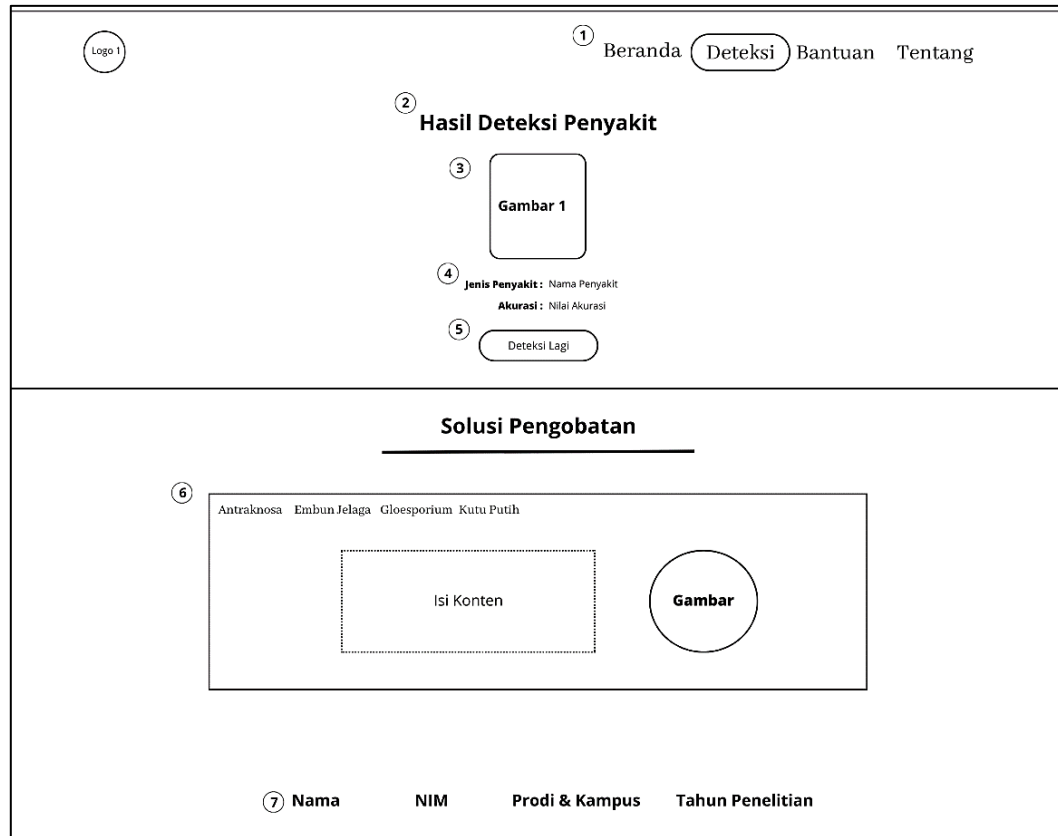


Gambar 3. 11 Rancangan Tampilan Deteksi

Keterangan :

1. *Menustrip*, berfungsi menampilkan daftar menu
2. *Label*, berfungsi menampilkan judul menu deteksi
3. *Label*, berfungsi menampilkan penjelasan fungsi dari menu deteksi
4. Kotak Garis, berfungsi menampilkan lokasi gambar yang dimasukkan
5. *Button* “Unggah Gambar” untuk mengunggah gambar dan *Button* “Prediksi” untuk memulai prediksi
6. *Label*, berfungsi menampilkan isi konten.
7. *Label* dan *Image*, berfungsi menampilkan isi konten dan gambar daun mangga

8. *Label*, berfungsi menampilkan Nama, NIM, Program Studi & Kampus dan Tahun Penelitian



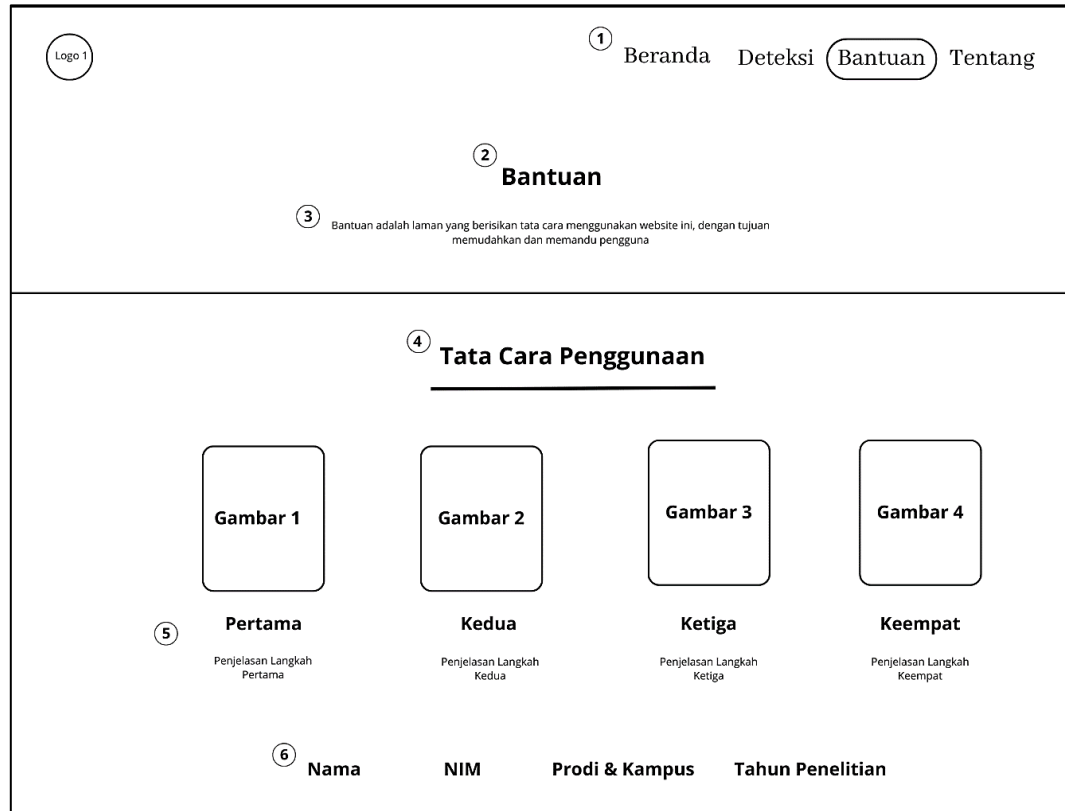
Gambar 3. 12 Rancangan Tampilan Hasil Deteksi

Keterangan :

1. *Menustrip*, berfungsi menampilkan daftar menu
2. *Label*, berfungsi menampilkan judul menu hasil deteksi
3. *Image*, menampilkan kembali gambar yang di-*input*
4. *Label*, berfungsi menampilkan nama penyakit dan nilai akurasi
5. *Button*, “Deteksi Lagi” untuk melakukan deteksi kembali
6. *Label*, berfungsi menampilkan isi konten solusi pengobatan
7. *Label*, berfungsi menampilkan Nama, NIM, Program Studi & Kampus dan Tahun Penelitian.

3.4.3. Desain Halaman Bantuan

Halaman Bantuan merupakan halaman yang memaparkan tata cara menggunakan sistem.



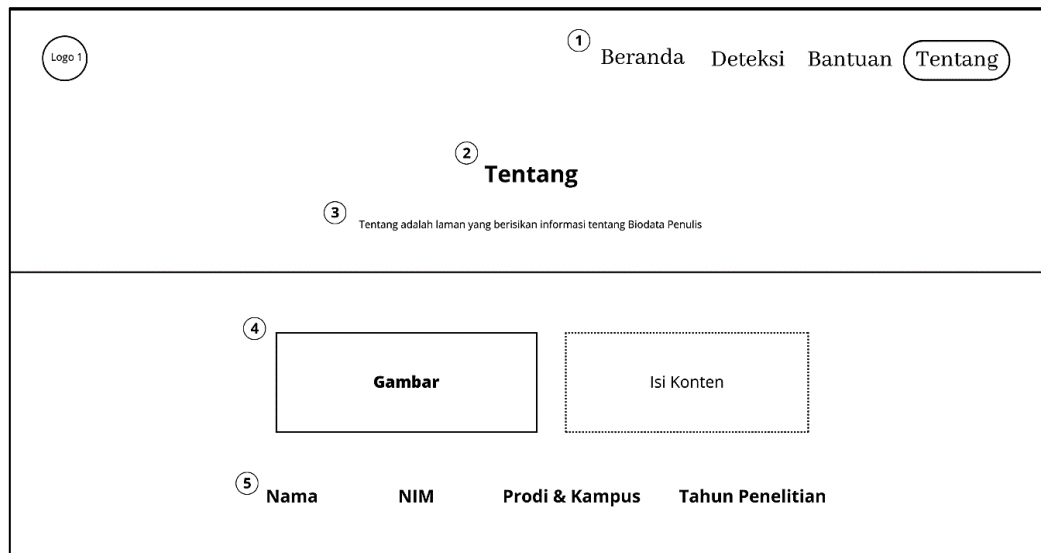
Gambar 3. 13 Rancangan Tampilan Bantuan

Keterangan :

1. *Menustrip*, berfungsi menampilkan daftar menu
2. *Label*, berfungsi menampilkan judul menu bantuan
3. *Label*, berfungsi menampilkan penjelasan fungsi dari menu bantuan
4. *Label*, berfungsi menampilkan sub judul
5. *Label*, berfungsi menampilkan penjelasan tutorial
6. *Label*, berfungsi menampilkan Nama, NIM, Program Studi & Kampus dan Tahun Penelitian

3.4.4. Desain Halaman Tentang

Halaman Tentang merupakan halaman yang memaparkan biodata penulis .



Gambar 3. 14 Rancangan Tampilan Tentang

Keterangan :

1. *Menustrip*, berfungsi menampilkan daftar menu
2. *Label*, berfungsi menampilkan judul menu tentang
3. *Label*, berfungsi menampilkan penjelasan fungsi dari menu tentang
4. *Label*, berfungsi menampilkan penjelasan tutorial
5. *Label*, berfungsi menampilkan Nama, NIM, Program Studi & Kampus dan Tahun Penelitian

BAB 4

IMPLEMENTASI DAN PENGUJIAN

4.1. Implementasi Sistem

Implementasi Sistem merupakan tindakan menerapkan hasil analisis dan rancangan ke dalam sistem yang akan dibangun, dengan melakukan penulisan kode-kode program. Sistem yang dibangun memiliki beberapa halaman, yaitu Beranda, Prediksi, Bantuan dan Tentang.

4.1.1. Halaman Beranda

Halaman Beranda adalah tampilan utama sistem, yang menjadi halaman pertama bagi pengguna apabila membuka sistem. Di halaman beranda, terdapat beberapa hal seperti judul penelitian, tombol mulai prediksi, informasi tentang penyakit mangga, beberapa gambar, nama penulis, NIM penulis, program studi dan tahun penelitian.

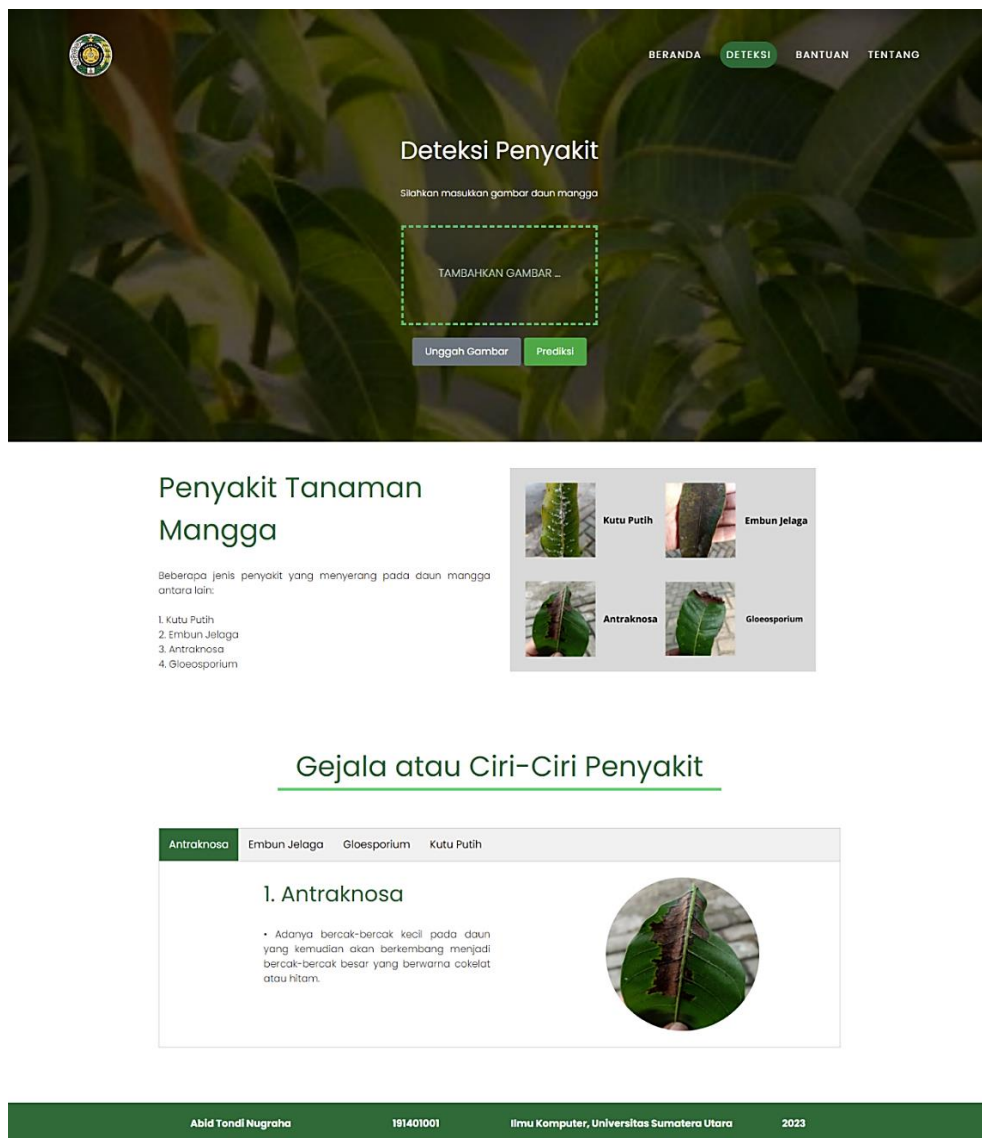


Gambar 4. 1 Halaman Beranda

4.1.2. Halaman Deteksi

4.1.2.1. Halaman Deteksi

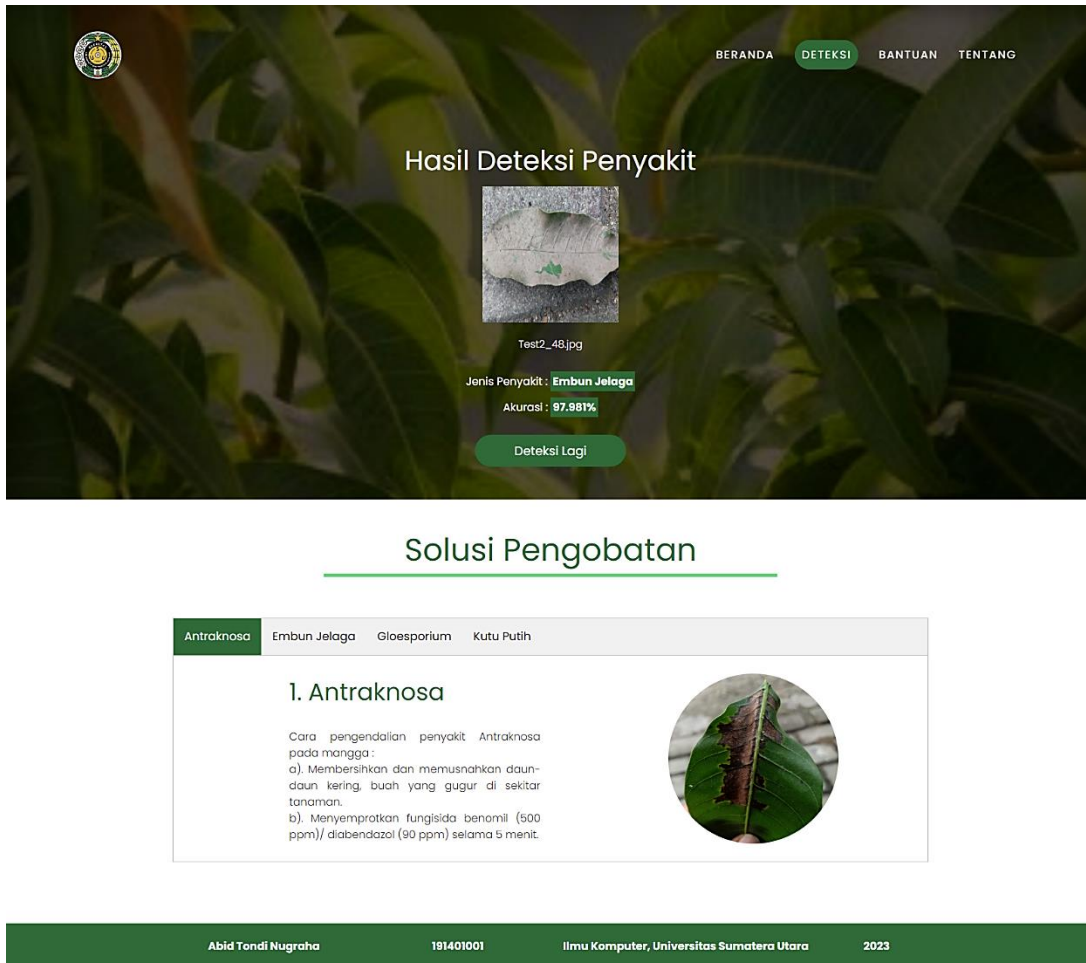
Halaman Deteksi adalah halaman untuk melakukan fungsi klasifikasi dari gambar yang hendak dideteksi. Di halaman prediksi, pengguna mengunggah gambar yang dideteksi dengan cara menekan tombol “Unggah Gambar” dan memulai proses pendeteksian dengan menekan tombol “Prediksi”. Selain itu, pengguna dapat membaca beberapa informasi seperti gejala penyakit pada daun mangga.



Gambar 4. 2 Halaman Deteksi

4.1.2.2. Halaman Hasil Deteksi

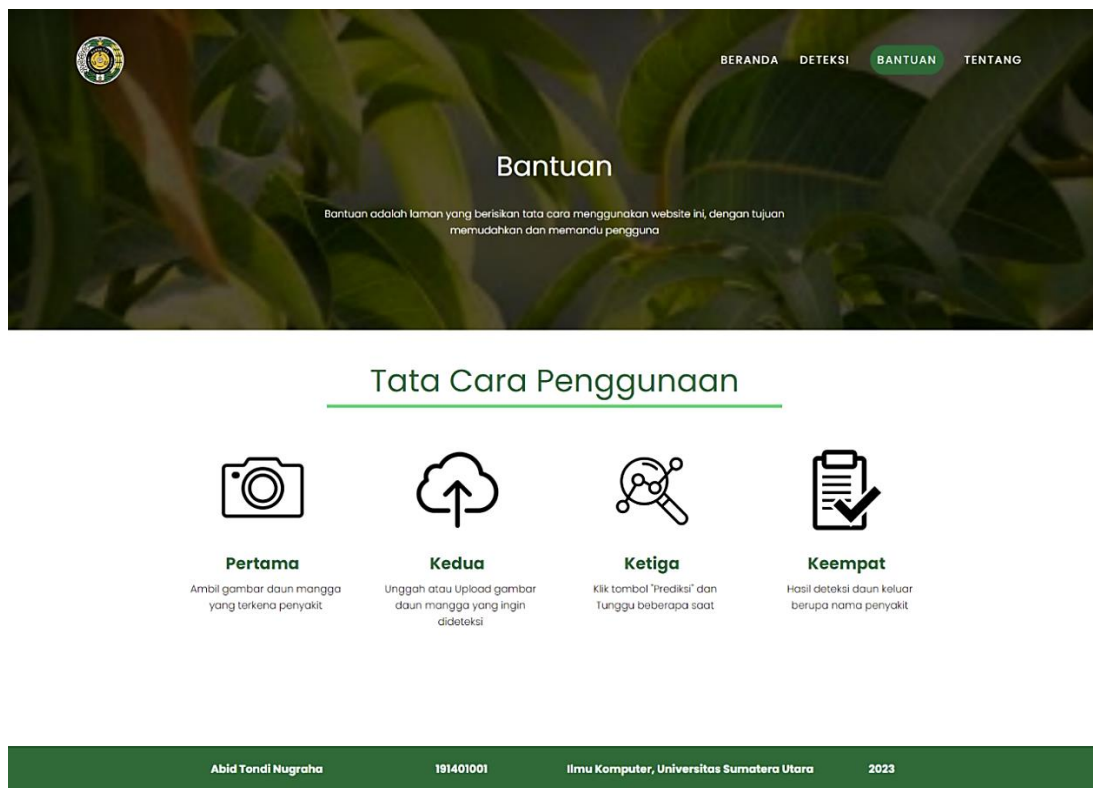
Halaman Hasil Deteksi adalah halaman yang menunjukkan hasil prediksi gambar daun. Di halaman prediksi, pengguna dapat membaca hasil prediksi yang telah selesai diproses, berupa nama penyakit daun mangga tersebut. Selain itu, pengguna dapat membaca beberapa informasi seperti solusi pengobatan penyakit pada daun mangga, sehingga daun tersebut dapat diatasi.



Gambar 4. 3 Halaman Hasil Deteksi

4.1.3. Halaman Bantuan

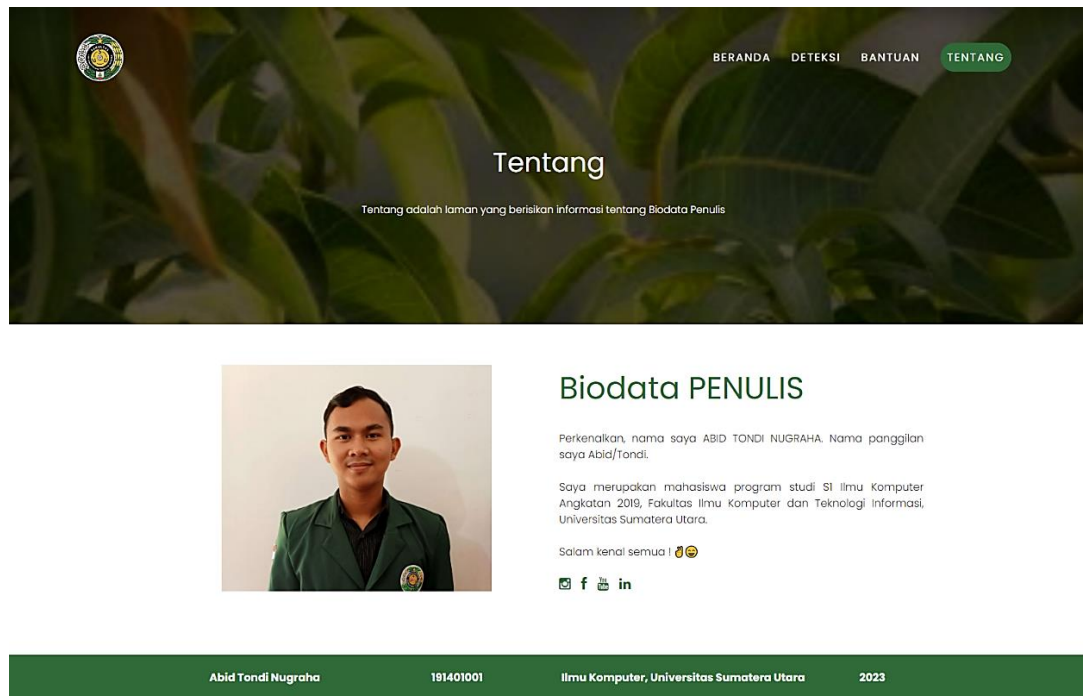
Halaman Bantuan adalah halaman yang memberitahukan kepada pengguna, tata cara menggunakan sistem yang tepat dan benar. Di halaman bantuan, terdapat bagaimana tata cara penggunaan sistem.



Gambar 4. 4 Halaman Bantuan

4.1.4. Halaman Tentang

Halaman Tentang adalah halaman yang berisikan biodata penulis. Di halaman tentang, terdapat beberapa hal seperti judul halaman, biodata penulis dan gambar penulis.



Gambar 4. 5 Halaman Tentang

4.2. Pengujian Sistem

4.2.1. Tahap Pelatihan Model

Pada tahap pelatihan, menggunakan *software Google Colab* dan semua *dataset* disimpan di dalam penyimpanan *Google Drive*.

```
from google.colab import drive
drive.mount('/content/drive')
```

Gambar 4. 6 Kode Program Koneksi Dataset.

Menurut gambar diatas, kode program melakukan *import* fungsi “drive” dari *library google.colab*, agar mengakses *Google Drive* dari *Colab*, sehingga *Colab* dapat mengolah *dataset* yang ada di *Google Drive*.


```
base_dir = '/content/drive/My Drive/Mangga/train'
print("Label Training :",os.listdir(base_dir))
```

```
Label Training : ['Antraknosa', 'Gloeosporium', 'Embun Jelaga', 'Kutu Putih']
```

Gambar 4. 7 Kode Program Akses Data Train

Menurut gambar diatas, kode program mendeklarasikan variabel “base_dir” yang berisikan alamat lokasi folder data latih (*train*) di *Google Drive* dan menampilkan daftar nama file dan folder yang ada di dalam folder latih (*train*), yang mewakili sebagai label latih (*train*).

Model dirancang menggunakan metode Convolutional Neural Network dan arsitektur Visual Geometry Group (VGG) 16.

Model: "vgg16"

Layer (type)	Output Shape	Param #
input_1 (InputLayer)	[(None, 240, 240, 3)]	0
block1_conv1 (Conv2D)	(None, 240, 240, 64)	1792
block1_conv2 (Conv2D)	(None, 240, 240, 64)	36928
block1_pool (MaxPooling2D)	(None, 120, 120, 64)	0
block2_conv1 (Conv2D)	(None, 120, 120, 128)	73856
block2_conv2 (Conv2D)	(None, 120, 120, 128)	147584
block2_pool (MaxPooling2D)	(None, 60, 60, 128)	0
block3_conv1 (Conv2D)	(None, 60, 60, 256)	295168
block3_conv2 (Conv2D)	(None, 60, 60, 256)	590080
block3_conv3 (Conv2D)	(None, 60, 60, 256)	590080
block3_pool (MaxPooling2D)	(None, 30, 30, 256)	0
block4_conv1 (Conv2D)	(None, 30, 30, 512)	1180160
block4_conv2 (Conv2D)	(None, 30, 30, 512)	2359808
block4_conv3 (Conv2D)	(None, 30, 30, 512)	2359808
block4_pool (MaxPooling2D)	(None, 15, 15, 512)	0
block5_conv1 (Conv2D)	(None, 15, 15, 512)	2359808
block5_conv2 (Conv2D)	(None, 15, 15, 512)	2359808
block5_conv3 (Conv2D)	(None, 15, 15, 512)	2359808
block5_pool (MaxPooling2D)	(None, 7, 7, 512)	0

Layer (type)	Output Shape	Param #
vgg16 (Functional)	(None, 7, 7, 512)	14714688
conv2d_3 (Conv2D)	(None, 5, 5, 32)	147488
max_pooling2d_3 (MaxPooling 2D)	(None, 2, 2, 32)	0
flatten_3 (Flatten)	(None, 128)	0
dense_8 (Dense)	(None, 128)	16512
dense_9 (Dense)	(None, 6)	774
=====		
Total params: 14,879,462		
Trainable params: 164,774		
Non-trainable params: 14,714,688		

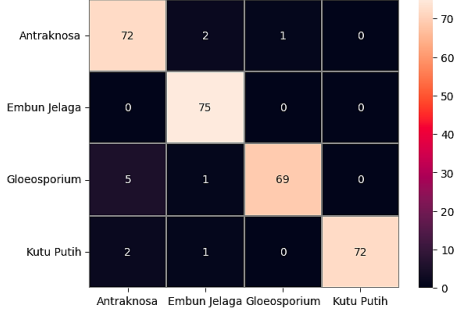
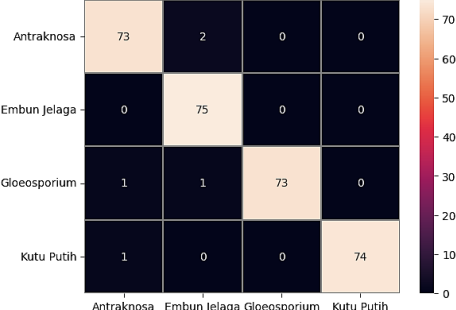
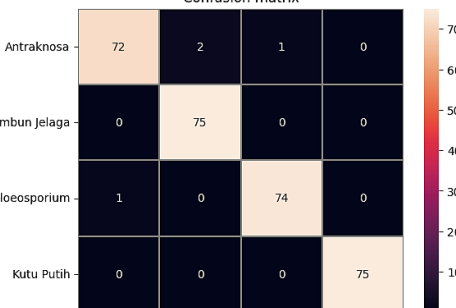
Gambar 4. 8 Lapisan pada Pelatihan Model

Pada penelitian ini, melakukan percobaan pelatihan model dengan menggunakan beberapa epoch yang bervariasi, yaitu epoch sebanyak 5, 10, 15 dan 20. Hal tersebut dilakukan untuk menentukan manakah epoch yang terbaik dalam menghasilkan nilai akurasi yang cukup tinggi, sehingga dapat digunakan dalam pembentukan model untuk sistem yang dirancang.

Beberapa hal yang termasuk ke dalam penilaian dan perbandingan untuk penentuan epoch adalah mencakup nilai *Precision*, *Recall*, *F1-Score*, *Accuracy* dan hasil percobaan prediksi dari masing-masing epoch.

Hasil pelatihan model dengan menggunakan beberapa epoch yang bervariasi, dapat dilihat pada tabel-tabel berikut ini.

1. Hasil *Precision*, *Recall* dan *F1-Score***Tabel 4. 1.** Hasil Precision, Recall dan F1-Score

No.	Hasil Klasifikasi	Confusion Matrix																				
Epoch-5	Laporan Hasil Klasifikasi :	<div>Confusion matrix</div>  <table><tr><td>Antraknosa</td><td>72</td><td>2</td><td>1</td><td>0</td></tr><tr><td>Embun Jelaga</td><td>0</td><td>75</td><td>0</td><td>0</td></tr><tr><td>Gloeosporium</td><td>5</td><td>1</td><td>69</td><td>0</td></tr><tr><td>Kutu Putih</td><td>2</td><td>1</td><td>0</td><td>72</td></tr></table> <div>AntraknosaEmbun JelagaGloeosporiumKutu Putih</div>	Antraknosa	72	2	1	0	Embun Jelaga	0	75	0	0	Gloeosporium	5	1	69	0	Kutu Putih	2	1	0	72
	Antraknosa		72	2	1	0																
	Embun Jelaga		0	75	0	0																
	Gloeosporium		5	1	69	0																
	Kutu Putih		2	1	0	72																
	precisionrecallf1-scoreupport																					
	Antraknosa0.910.960.9475																					
	Embun Jelaga0.951.000.9775																					
	Gloeosporium0.990.920.9575																					
	Kutu Putih1.000.960.9875																					
accuracy0.960.960.96300																						
macro avg0.960.960.96300																						
weighted avg0.960.960.96300																						
Epoch-10	Laporan Hasil Klasifikasi :	<div>Confusion matrix</div>  <table><tr><td>Antraknosa</td><td>73</td><td>2</td><td>0</td><td>0</td></tr><tr><td>Embun Jelaga</td><td>0</td><td>75</td><td>0</td><td>0</td></tr><tr><td>Gloeosporium</td><td>1</td><td>1</td><td>73</td><td>0</td></tr><tr><td>Kutu Putih</td><td>1</td><td>0</td><td>0</td><td>74</td></tr></table> <div>AntraknosaEmbun JelagaGloeosporiumKutu Putih</div>	Antraknosa	73	2	0	0	Embun Jelaga	0	75	0	0	Gloeosporium	1	1	73	0	Kutu Putih	1	0	0	74
	Antraknosa		73	2	0	0																
	Embun Jelaga		0	75	0	0																
	Gloeosporium		1	1	73	0																
	Kutu Putih		1	0	0	74																
	precisionrecallf1-scoreupport																					
	Antraknosa0.970.970.9775																					
	Embun Jelaga0.961.000.9875																					
	Gloeosporium1.000.970.9975																					
	Kutu Putih1.000.990.9975																					
accuracy0.980.980.98300																						
macro avg0.980.980.98300																						
weighted avg0.980.980.98300																						
Epoch-15	Laporan Hasil Klasifikasi :	<div>Confusion matrix</div>  <table><tr><td>Antraknosa</td><td>72</td><td>2</td><td>1</td><td>0</td></tr><tr><td>Embun Jelaga</td><td>0</td><td>75</td><td>0</td><td>0</td></tr><tr><td>Gloeosporium</td><td>1</td><td>0</td><td>74</td><td>0</td></tr><tr><td>Kutu Putih</td><td>0</td><td>0</td><td>0</td><td>75</td></tr></table> <div>AntraknosaEmbun JelagaGloeosporiumKutu Putih</div>	Antraknosa	72	2	1	0	Embun Jelaga	0	75	0	0	Gloeosporium	1	0	74	0	Kutu Putih	0	0	0	75
	Antraknosa		72	2	1	0																
	Embun Jelaga		0	75	0	0																
	Gloeosporium		1	0	74	0																
	Kutu Putih		0	0	0	75																
	precisionrecallf1-scoreupport																					
	Antraknosa0.990.960.9775																					
	Embun Jelaga0.971.000.9975																					
	Gloeosporium0.990.990.9975																					
	Kutu Putih1.001.001.0075																					
accuracy0.990.990.99300																						
macro avg0.990.990.99300																						
weighted avg0.990.990.99300																						

Epoch-20	Laporan Hasil Klasifikasi :					Confusion matrix			
		precision	recall	f1-score	support	Antraknosa	Embun Jelaga	Gloeosporium	Kutu Putih
	Antraknosa	0.99	0.96	0.97	75	72	2	1	0
	Embun Jelaga	0.97	1.00	0.99	75	0	75	0	0
	Gloeosporium	0.99	0.99	0.99	75	1	0	74	0
	Kutu Putih	1.00	1.00	1.00	75	0	0	0	75
	accuracy			0.99	300				
	macro avg	0.99	0.99	0.99	300				
	weighted avg	0.99	0.99	0.99	300				
						Antraknosa	Embun Jelaga	Gloeosporium	Kutu Putih

Menurut tabel 4.1, epoch 15 dan epoch 20 menghasilkan hasil *Precision*, *Recall* dan *F1-Score* yang cukup tinggi, dibandingkan epoch 5 dan epoch 10, yaitu nilai rata-rata *Precision* sebesar 0,98; nilai rata-rata *Recall* sebesar 0,98 dan nilai rata-rata *F1-Score* sebesar 0,98.

2. Hasil Accuracy, Loss & Training Time

Tabel 4. 2. Hasil dari Accuracy, Loss & Training Time






Percobaan	Accuracy	Loss	Training Time	
Epoch-5	96,00	10,27	4.464 s	1 Jam 14 menit 24 Detik
Epoch-10	98,33	05,11	9.045 s	2 Jam 30 Menit 45 Detik
Epoch-15	98,67	03,32	13.255 s	3 Jam 40 Menit 55 Detik
Epoch-20	98,67	02,28	15.281 s	4 Jam 14 Menit 41 Detik




Menurut tabel 4.2., epoch 15 dan epoch 20 menghasilkan hasil *Accuracy* yang tinggi dan sama, namun epoch 20 menghasilkan nilai *Loss* yang paling rendah dibandingkan epoch yang lainnya.

3. Percobaan Prediksi

Pada percobaan ini, menggunakan 8 sampel gambar daun mangga, untuk percobaan awal prediksi. Berikut adalah hasil percobaan prediksi masing-masing epoch.

Tabel 4. 3. Hasil dari Percobaan Prediksi

No.	Gambar	Akurasi			
		Epoch-5	Epoch-10	Epoch-15	Epoch-20
1		99,82	98,97	99,24	99,94
2		97,22	99,74	99,88	99,72
3		96,93	98,26	99,81	99,53
4		98,48	99,79	99,99	99,99
5		88,41	84,51	96,87	95,71

6		99,17	99,91	99,99	99,99
7		79,27	95,95	95,86	88,68
8		99,02	99,97	99,97	99,99
Rata-Rata		94,79	97,13	98,95	97,94

Menurut tabel 4.3, epoch 15 menghasilkan nilai rata-rata akurasi paling tinggi dalam melakukan percobaan prediksi, dibandingkan epoch yang lainnya.

Berdasarkan hasil dari beberapa tabel percobaan diatas, epoch sebanyak 15 menghasilkan nilai yang lebih unggul, yang mencakup nilai *Precision*, *Recall*, *OF1-Score*, *Accuracy* dan percobaan prediksi, dibandingkan epoch lainnya.

Oleh sebab itu, penelitian ini menggunakan epoch sebanyak 15 dalam melakukan pelatihan dan pembentukan model, untuk sistem yang dirancang.

Berikut ini merupakan proses pelatihan model menggunakan data-data yang telah dikumpulkan dan dikelompokkan menjadi beberapa kelas.

```
history = model.fit(train_images, train_labels, batch_size=128, epochs=15, validation_split = 0.2)
print()
print("=====")
print("Proses Training Selesai !")
```

Gambar 4. 9 Kode Program Mulai Pelatihan

Dengan menggunakan fungsi *model.fit*, diharapkan proses pelatihan nya dapat menghasilkan model yang baik dalam melakukan klasifikasi agar mampu menyelesaikan permasalahan .

```
Epoch 1/15
8/8 [=====] - 887s 112s/step - loss: 1.2775 - accuracy: 0.4323 - val_loss: 0.9648 - val_accuracy: 0.5875
Epoch 2/15
8/8 [=====] - 892s 114s/step - loss: 0.7219 - accuracy: 0.7708 - val_loss: 0.5671 - val_accuracy: 0.8417
Epoch 3/15
8/8 [=====] - 901s 115s/step - loss: 0.3961 - accuracy: 0.9083 - val_loss: 0.3602 - val_accuracy: 0.8667
Epoch 4/15
8/8 [=====] - 883s 113s/step - loss: 0.2199 - accuracy: 0.9490 - val_loss: 0.2633 - val_accuracy: 0.9000
Epoch 5/15
8/8 [=====] - 883s 113s/step - loss: 0.1248 - accuracy: 0.9750 - val_loss: 0.1677 - val_accuracy: 0.9458
Epoch 6/15
8/8 [=====] - 889s 114s/step - loss: 0.0709 - accuracy: 0.9875 - val_loss: 0.1327 - val_accuracy: 0.9625
Epoch 7/15
8/8 [=====] - 892s 114s/step - loss: 0.0443 - accuracy: 0.9969 - val_loss: 0.1081 - val_accuracy: 0.9708
Epoch 8/15
8/8 [=====] - 867s 111s/step - loss: 0.0272 - accuracy: 1.0000 - val_loss: 0.0900 - val_accuracy: 0.9833
Epoch 9/15
8/8 [=====] - 895s 115s/step - loss: 0.0187 - accuracy: 1.0000 - val_loss: 0.0907 - val_accuracy: 0.9708
Epoch 10/15
8/8 [=====] - 890s 114s/step - loss: 0.0138 - accuracy: 1.0000 - val_loss: 0.0727 - val_accuracy: 0.9750
Epoch 11/15
8/8 [=====] - 860s 110s/step - loss: 0.0097 - accuracy: 1.0000 - val_loss: 0.0710 - val_accuracy: 0.9750
Epoch 12/15
8/8 [=====] - 887s 114s/step - loss: 0.0077 - accuracy: 1.0000 - val_loss: 0.0681 - val_accuracy: 0.9750
Epoch 13/15
8/8 [=====] - 857s 109s/step - loss: 0.0062 - accuracy: 1.0000 - val_loss: 0.0621 - val_accuracy: 0.9792
Epoch 14/15
8/8 [=====] - 887s 114s/step - loss: 0.0048 - accuracy: 1.0000 - val_loss: 0.0642 - val_accuracy: 0.9833
Epoch 15/15
8/8 [=====] - 885s 113s/step - loss: 0.0041 - accuracy: 1.0000 - val_loss: 0.0585 - val_accuracy: 0.9792
```

Gambar 4. 10 Proses Pelatihan Model

Pada tahapan tersebut, epoch yang ditentukan adalah 15, yang terdiri atas 8 steps yang dilakukan. Selain itu proses training melibatkan 1500 data yang dibagi menjadi data latih dan tes, yang mana data latih sebanyak 300 data per kelas dan data tes sebanyak 75 data per kelas dengan perbandingan 80% : 20%.

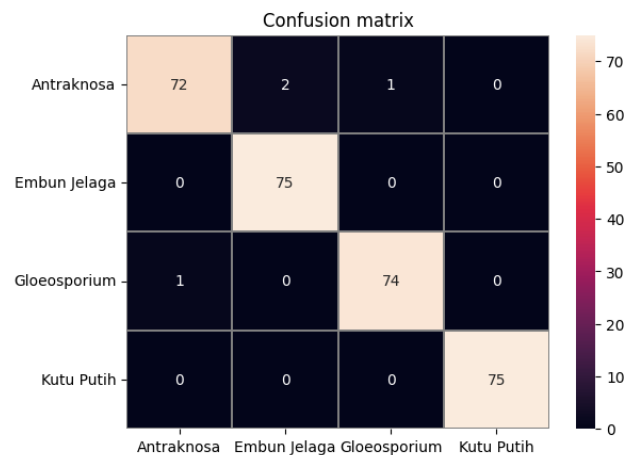
Waktu yang digunakan untuk melakukan proses training, secara menyeluruh adalah sekitar 13.225 detik atau setara dengan 3 jam 40 Menit 55 detik. Hasil akhir dari epoch menghasilkan nilai *accuracy* sebesar 100 dan loss sebesar 0. Berikut ini adalah hasil dari nilai secara keseluruhan.

Laporan Hasil Klasifikasi :

	precision	recall	f1-score	support
Antraknosa	0.99	0.96	0.97	75
Embun Jelaga	0.97	1.00	0.99	75
Gloeosporium	0.99	0.99	0.99	75
Kutu Putih	1.00	1.00	1.00	75
accuracy			0.99	300
macro avg	0.99	0.99	0.99	300
weighted avg	0.99	0.99	0.99	300

Gambar 4. 11 Hasil Akhir Pelatihan Model

Pada gambar di atas, nilai-nilai yang ditampilkan merupakan hasil dari proses *training* yang dilakukan di mana perhitungannya dapat dilihat di bawah ini, dengan menghitung *confusion matrix*.



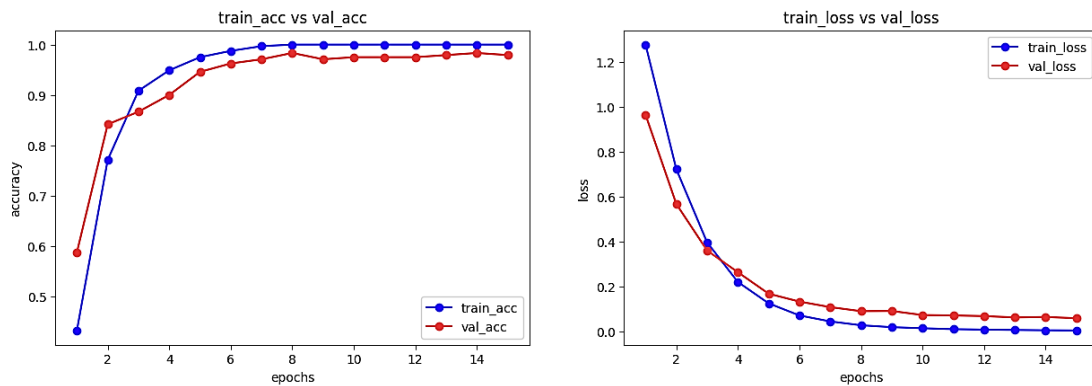
Gambar 4. 12 Hasil *Confusion Matrix*

a. *Accuracy*

Akurasi (*accuracy*) merupakan total persentase hasil prediksi gambar yang benar dari semua data yang telah diuji coba, dengan rumus :

$$\begin{aligned}
 Accuracy &= \frac{\text{True Prediction}}{\text{Total Dataset}} \\
 Accuracy &= \frac{72+75+74+75}{300} \\
 Accuracy &= 0,98
 \end{aligned}$$

Grafik yang menggambarkan akurasi pelatihan dapat ditemukan dalam gambar di bawah ini.



Gambar 4.13 Grafik Accuracy & Loss

b. Precision

Precision merupakan total persentase prediksi positif yang tepat dari keseluruhan prediksi positif yang dilakukan oleh model, dengan rumus :

$$Precision = \frac{True\ Positive}{True\ Positive + False\ Positive}$$

Contoh perhitungan *precision* dari kelas “Antraknosa” :

$$True\ Positive = 72$$

$$False\ Positive = 1$$

$$Precision = \frac{72}{72 + 1}$$

$$Precision = \frac{72}{73}$$

$$Precision = 0,986$$

Maka nilai *precision* dari kelas “Antraknosa” adalah $= 0,986 = 0,99$

Selanjutnya menghitung rata-rata dari keseluruhan nilai *precision* setiap kelas, dengan rumus sebagai berikut

$$\begin{aligned}\text{Rata-Rata Precision} &= \frac{\text{Total jumlah precision}}{\text{Total kelas}} \\ \text{Rata-Rata Precision} &= \frac{0,99+0,97+0,99+1,00}{4} \\ \text{Rata-Rata Precision} &= 0,98\end{aligned}$$

Oleh karena itu, nilai rata-rata *precision* dari hasil pelatihan ini adalah 0,98.

c. *Recall*

Recall merupakan total persentase dari perhitungan keseluruhan nilai positif yang berhasil dikenali oleh model, dengan rumus :

$$\text{Recall} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}}$$

Contoh perhitungan *recall* dari kelas “Antraknosa” :

$$\text{True Positive} = 72$$

$$\text{False Negative} = 3$$

$$\begin{aligned}\text{Recall} &= \frac{72}{72 + 3} \\ \text{Recall} &= \frac{72}{75} \\ \text{Recall} &= 0,96\end{aligned}$$

Maka nilai *recall* dari kelas Antraknosa adalah = 0,96

Selanjutnya menghitung rata-rata dari keseluruhan nilai *recall* setiap kelas, dengan rumus sebagai berikut :

$$\begin{aligned}\text{Rata-Rata Recall} &= \frac{\text{Total jumlah recall}}{\text{Total Kelas}} \\ \text{Rata-Rata Recall} &= \frac{0,96+1,00+0,99+1,00}{4} \\ \text{Rata-Rata Recall} &= 0,98\end{aligned}$$

Oleh karena itu, nilai rata-rata *recall* dari hasil pelatihan ini adalah **0,98**.

d. *F1-Score*

F1-Score merupakan rata-rata *harmonic* dari hasil *recall* dengan *precision*. *F1-Score* berfungsi untuk menghasilkan evaluasi kinerja model secara keseluruhan.

Contoh perhitungan *F1-Score* dari kelas “Antraknosa” :

$$\text{Precision} = 0,99$$

$$\text{Recall} = 0,96$$

$$F1\text{-Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

$$F1\text{-Score} = 2 \times \frac{0,99 \times 0,96}{0,99 + 0,96}$$

$$F1\text{-Score} = 2 \times \frac{0,95}{1,95}$$

$$F1\text{-Score} = 0,97$$

Maka nilai *F1-Score* untuk Antraknosa adalah 0,97

Selanjutnya menghitung rata-rata dari keseluruhan nilai *F1-Score* setiap kelas, dengan rumus sebagai berikut :

$$\text{Rata-Rata } F1\text{-Score} = \frac{\text{Total jumlah } F1\text{-Score}}{\text{Total Kelas}}$$

$$\text{Rata-Rata } F1\text{-Score} = \frac{0,97+0,99+0,99+1,00}{4}$$

$$\text{Rata-Rata } F1\text{-Score} = 0,98$$

Oleh karena itu, nilai rata-rata *F1-Score* hasil pelatihan ini secara keseluruhan adalah 0,98. Hasil tersebut menunjukkan bahwa pelatihan model yang telah dilakukan memiliki performa yang sangat baik.

4.2.2. Tahap Integrasi

Tahap Integrasi adalah sebuah tahap yang menghubungkan model yang sudah disimpan, ke dalam sistem yang telah dirancang, dengan menggunakan suatu *library*. Pada penelitian ini, menggunakan *library* “*Flask*” karena bersifat ringan dan fleksibilitas sehingga mudah untuk digunakan ke sistem *website*.

4.2.2.1. Kode Python

Berikut ini merupakan kode program *flask* untuk mengkoneksikan model ke sistem dan menetakannya sebagai kode utama, dengan memberi nama **app.py**.

```
from flask import Flask, render_template, request, send_from_directory, send_file
from keras.models import load_model
import numpy as np
from keras.applications.vgg16 import preprocess_input
from sklearn.metrics import accuracy_score
import os
import math
from PIL import Image
```

Gambar 4. 14 Memanggil *Library* yang dibutuhkan

Menurut gambar diatas, terdapat beberapa *library* yang digunakan di dalam *app.py*. Salah satunya adalah *library flask* yang menggunakan fungsi *render_template*, *request*, *send_from_directory*, *send_file* untuk mengambil data gambar dari sistem dan mengirim kembali hasilnya.

```
app = Flask(__name__, static_folder='public')
app.config['UPLOAD_FOLDER'] = app.static_folder + '/static/images'
app.config['ALLOWED_EXTENSIONS'] = {'png', 'jpg', 'jpeg'}
target_img = "public/static/images"

model = load_model("model.h5", compile=False)
model.compile(optimizer="Adam")

@app.route('/')
def index_view():
    return send_from_directory(app.static_folder, 'index.html')
```

Gambar 4. 15 Tahap Deklarasi (Insialisasi)

Menurut gambar diatas, program melakukan pendeklarasi terhadap variabel awal seperti penentuan jenis gambar, lokasi folder gambar, mengkoneksikan model dll.

untuk memudahkan menjalankan perintah program selanjutnya dan menentukan halaman utama, apabila program dijalankan.

```
def predict_image(image_path):
    image = Image.open(image_path)
    image = image.resize((240, 240)) # Sesuaikan ukuran dengan model
    image = np.expand_dims(image, axis=0)
    image = image / 255.0 # Normalisasi gambar jika diperlukan

    prediction = model.predict(image)
    predicted_class_index = np.argmax(prediction)

    class_labels = ["Antraknosa", "Embun Jelaga", "Gloeosporium", "Kutu Putih"]
    predicted_class_label = class_labels[predicted_class_index]
    predicted_value = prediction[0][predicted_class_index]

    return predicted_class_label, predicted_value
```

Gambar 4. 16 Fungsi Memanggil Model

```
@app.route('/hasil', methods=['GET', 'POST'])
def hasil():
    if request.method == 'POST':
        file = request.files['file']

        if file.filename == '':
            return "No selected file"

        if file and allowed_file(file.filename):
            filename = file.filename
            filepath = os.path.join(app.config['UPLOAD_FOLDER'], filename)
            file.save(filepath)

            predicted_class_label, predicted_value = predict_image(filepath)
```

Gambar 4. 17 Fungsi Utama Integrasi

Menurut gambar diatas, fungsi “*hasil*” merupakan fungsi utama untuk melakukan integrasi antara *model* dengan *website*. Proses pertama yang dilakukan adalah pengecekan format gambar dan kemudian melakukan klasifikasi terhadap gambar tersebut, melalui kode “*prediction = model.predict(image)*” yang ada di dalam fungsi *predict_image(filepath)*.

```
return render_template('hasil.html', jenis_penyakit=predicted_class_label,
                        nilai_prediksi=str(round(100*np.max(predicted_value),3)),
                        nama_file=filename)
os.remove(filepath)
```

Gambar 4. 18 Fungsi Mengirim Hasil dari Model

Menurut gambar diatas, hasil dari proses klasifikasi dikirimkan ke halaman *hasil.html* dengan menggunakan sintaks “*render_template*”, yang mana hasil yang dikirimkan berupa jenis penyakit, nilai akurasi dan nama gambarnya.

4.2.2.2. Kode HTML

Berikut ini merupakan kode program HTML untuk menampilkan hasil dari proses klasifikasi di halaman **hasil.html** pada sistem.

```
<div style="text-align: center;">
  
  <p class="image-title-wrap2">{{ nama_file }}</p>
  <h4 style="color: white; padding-top: 10px;">Jenis Penyakit : <mark> {{ jenis_penyakit }}</mark></h4>
  <h4 style="color: white; padding-top: 5px;">Akurasi : <mark> {{ nilai_akurasi }}%</mark></h4>
  <div class="bt_main">
    <div class="button2"><a href="../../../public/deteksi.html">Deteksi Lagi</a></div>
  </div>
</div>
```

Gambar 4. 19 Kode HTML Menampilkan Hasil dari Model






Menurut gambar di atas, tag `` menampilkan kembali gambar yang di-*input* sebelumnya dan tag `<h4>` untuk menampilkan teks berupa jenis penyakit dan nilai akurasi dari hasil prediksi yang didapat.






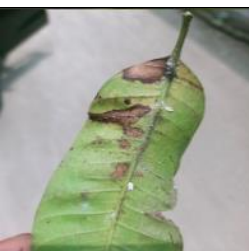
4.2.3 Hasil Pengujian




Setelah melakukan pelatihan model dan integrasi, hal yang selanjutnya dilakukan adalah melakukan pengujian sistem, untuk menguji kinerja dari model yang telah dibuat. Pengujian dilakukan dengan menggunakan 14 sampel gambar dari daun mangga.

Tabel dibawah merupakan hasil pengujian dari sistem yang sudah dirancang, dalam dalam bentuk tabel.

Tabel 4. 4. Hasil Pengujian

No.	Gambar	Label	Prediksi	Akurasi
1		Embun Jelaga	Embun Jelaga	99,24
2		Gloeosporium	Gloeosporium	80,08
3		Gloeosporium	Antraknosa	92,49
4		Embun Jelaga	Embun Jelaga	99,88
5		Gloeosporium	Gloeosporium	99,81

6		Antraknosa	Antraknosa	99,99
7		Kutu Putih	Kutu Putih	96,87
8		Antraknosa	Antraknosa	99,99
9		Embun Jelaga	Embun Jelaga	95,86
10		Embun Jelaga	Embun Jelaga	99,97
11		Antraknosa	Antraknosa	99,98

12		Kutu Putih	Kutu Putih	99.98
13		Kutu Putih	Kutu Putih	99.84
14		Antraknosa	Antraknosa	99.94

Berdasarkan hasil tabel tersebut, perancangan sistem deteksi penyakit pada daun mangga menggunakan metode CNN (*Convolution Neural Network*) tergolong bagus, dengan hasil prediksi cukup akurat untuk masing-masing gambar, dengan nilai rata-rata hasil prediksi sebesar 0,9742 atau 97%.

Menurut Tabel 4.4, terdapat hasil prediksi yang salah pada uji coba prediksi gambar daun mangga, yaitu pada gambar nomor 3.

Sistem menampilkan hasil prediksi dengan nama **Antraknosa**. Sedangkan seharusnya menurut label aktual, gambar tersebut merupakan penyakit **Gloeosporium**. Hal tersebut terjadi dikarenakan adanya bentuk kemiripan citra antara daun Antraknosa dengan *Gloeosporium*, sehingga memungkinkan terjadinya perbedaan antara label akurasi dengan label aktual.

BAB 5

KESIMPULAN DAN SARAN

5.1. Kesimpulan

Berdasarkan hasil penelitian klasifikasi penyakit pada daun mangga dengan menggunakan metode *Convolution Neural Network* (CNN) yang telah dilakukan, maka beberapa kesimpulannya yang dapat diberikan adalah :

1. Metode *Convolution Neural Network* dengan arsitektur VGG-16 mampu mengidentifikasi gambar penyakit pada daun mangga dengan sangat baik, dengan proses *training* model sebanyak 15 *epoch* dan 8 *steps* untuk jumlah data sebanyak 1500 data, dengan durasi waktu 13.225 detik atau setara dengan 3 jam 40 Menit 55 detik, serta hasil akurasi yang diperoleh selama proses pelatihan model secara keseluruhan adalah mencapai 0,986.
2. Performa model dari metode Convolutional Neural Networks (CNN) menggunakan arsitektur VGG-16 bersifat sangat baik, dengan hasil nilai *Accuracy* = 0,98, *Precision* = 0,98 dan *F1-Score* = 0,98.
3. Perancangan sistem klasifikasi penyakit pada daun mangga dengan menggunakan metode CNN (*Convolution Neural Network*) berfungsi dengan baik dan efektif, dengan nilai rata-rata dari keseluruhan hasil prediksi cukup tinggi, yaitu sebesar 0,9742 atau 97 %.
4. Menurut hasil pengujian, terdapat sebuah kesalahan yang mana sistem menampilkan hasil prediksi gambar dengan nama **Antraknosa**. Sedangkan seharusnya menurut label aktual, gambar tersebut merupakan penyakit **Gloeosporium**. Hal tersebut terjadi dikarenakan adanya bentuk kemiripan citra antara daun Antraknosa dengan *Gloeosporium*, sehingga memungkinkan terjadinya perbedaan hasil antara label akurasi dengan label aktual.

5.2. Saran

Berdasarkan hasil penelitian klasifikasi penyakit pada daun mangga dengan menggunakan metode *Convolution Neural Network* (CNN) yang telah dilakukan, maka dapat diberi beberapa sarannya yang dapat diberikan adalah:

1. Menambahkan kelas penyakit yang berbeda dan jumlah gambar *dataset* secara variasi, sehingga dapat meningkatkan nilai akurasi pada saat deteksi gambar daun mangga secara berlangsung.
2. Melanjutkan perancangan sistem dengan sistem berbasis yang berbeda, seperti berbasis aplikasi *Mobile* dan *Desktop*.
3. Sistem hanya mampu mendeteksi satu daun dalam satu gambar. Diharapkan pengembangan selanjutnya, sistem mampu mendeteksi banyak daun dalam satu gambar, untuk mempermudah pengguna.
4. Membuat perbedaan yang jelas antara gambar daun yang terkena **Antraknosa** dengan daun yang terkena **Gloeosporium**, dengan memperhatikan ketajaman gambar, pencahayaan gambar, fokus gambar, sudut pengambilan gambar dan aspek-aspek lainnya, sehingga tidak lagi terjadinya perbedaan hasil antara label akurasi dengan label aktual.

DAFTAR PUSTAKA

- Abdul Kholik. (2021). *Klasifikasi Menggunakan Convolutional Neural Network (CNN) pada Tangkapan Layar Halaman Instagram*. Jurnal Data Mining Sistem Informasi, 2(2): 10-20.
- Albashish, Dheeb. Al-Sayyed, Rizik. Abdullah, Azizi. Ryalat, Mohammad Hashem. Almansour, Needa Ahmad. 2021. *Deep CNN Model based on VGG16 for Breast Cancer Classification*. IEEE International Conference on Information Technology (ICIT), pp. 4, 2021
- Alzubaidi, L., Zhang, J., Humaidi, A. J., Al-Dujaili, A., Duan, Y., Al-Shamma, O., Santamaría, J., Fadhel, M. A., Al-Amidie, M., & Farhan, L. (2021). *Review of deep learning: concepts, CNN architectures, challenges, applications, future directions*. Journal of Big Data, 8(1). <https://doi.org/10.1186/s40537-021-00444-8>
- Arrofiqoh, Erlyna Nour & Harintaka. (2018). *Implementasi Metode Convolutional Neural Network untuk Klasifikasi Tanaman Pada Citra Resolusi Tinggi*. Geomatika, 24(2): 61-68 .
- Badan Penyuluhan dan Pengembangan Sumber Daya Manusia Pertanian (BPPSDMP). (2020). *Kutu Putih (Pseudococcus)*. Diakses pada tanggal 25 januari 2023, dari <http://cybex.pertanian.go.id/mobile/artikel/96297/Kutu-Putih-Pseudococcus/>
- Felix. Wijaya, Steven. Sutra, Stephen Putra. Kosasih, Pyter Wahyu. Sirait, Pahala. (2020). *Implementasi Convolutional Neural Network Untuk Identifikasi Jenis Tanaman Melalui Daun*. Jurnal SIFO Mikroskil, 21(1): 1-10.

- Feriawan, Jimmy. Swanjaya, Daniel. 2020. *Perbandingan Arsitektur Visual Geometry Group dan MobileNet Pada Pengenalan Jenis Kayu*. Seminar Nasional Inovasi Teknologi, UN PGRI Kediri: 186-187
- Ferryansyah, Wahyu. 2023. *Rekacipta Artificial Intelligence (Ai) Dengan Metode Convolution Neural Network (CNN) Untuk Klasifikasi Penyakit Daun Cabai Berbasis Website Dengan Bantuan*. Skripsi : Universitas Sumatera Utara.
- Fujiyoshi, Hironobu. Hirakawa, Tsubasa. Yamashita, Takayoshi. (2019). *Deep learning-based image recognition for autonomous driving*. IATSS Research, 43(4): 244-252. <https://doi.org/10.1016/j.iatssr.2019.11.008>.
- Ghimire, Sujan. Ravinesh C. Deo. David Casillas-Pérez. Sancho Salcedo-Sanz. Ekta Sharma. Mumtaz Ali. (2022). *Deep learning CNN-LSTM-MLP hybrid fusion model for feature optimizations and daily solar radiation prediction*. Measurement, 202 (2022) 111759: 1-2.
- Herdianto, Asep Ardi. Hidayat, Nurul. Dewi, Ratih Kartika. (2019). *Sistem Diagnosis Penyakit Tanaman Mangga Menggunakan Metode Bayesian Network*. Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer, 3(4): 3597-3602.
- Herlambang, Muhammad Fahmi. (2020). *Pengenalan Karakter Huruf Braille Dengan Metode Convolutional Neural Network*. Skripsi thesis, Institut Teknologi Nasional Bandung.
- Mia, Md. Rasel. Roy, Sujit. Subrata Kumar Das, Rahman, Md. Atikur. (2019). *Mango Leaf Diseases Recognition Using Neural Network and Support Vector Machine*. International Journal of Computer Science Engineering (IJCSE). Vol. 8. No.01: 24-25

Njayou Youssouf. (2022). *Traffic sign classification using CNN and detection using faster-RCNN and YOLOV4*. *Heliyon*, 8(2022): 1-8.

Novia Farhan Nissa, Angelia Janiati, Nilam Cahya , Anton , Puji Astuti. (2021). *Application of Deep Learning Using Convutional Neural Network(CNN) Method for Women's Skin Classification*. *Scientific Journal of Informatics*, 8(1): 144-153

Nurmila, Nurhaeni, Ahmad, Ridhay. (2019). EKSTRAKSI DAN KARAKTERISASI PEKTIN DARI KULIT BUAH MANGGA HARUMANIS (*Mangifera Indica L.*) BERDASARKAN VARIASI SUHU DAN WAKTU. *Jurnal Riset Kimia*, 5(1): 58-67.

Plantix.-. Kudis pada Mangga. Diakses pada Tanggal 12 Februari 2023 melalui <https://plantix.net/id/library/plant-diseases/100330/mango-scab>

Pratama, Alif Nandiar Bintang. (2018). STUDI SIFAT FISIK-KIMIA DODOL DENGAN SUBSTITUSI TEPUNG BIJI MANGGA DARI TIGA VARIETAS SELAMA PENYIMPANAN. Skripsi : Universitas Muhammadiyah Malang.

Putri, Dini. Ridha Dwiki. Fahlevi, Muhammad Reza. (2022). *Identifikasi Hama Dan Penyakit Buah Mangga Menggunakan Metode Dempster Shafer*. *InfoSys Journal*, 6 (2).

Rohmatika, Arina .(2022). UPAYA PENGENDALIAN PENYAKIT ANTRAKNOSA DAN EMBUN JELAGA PADA TANAMAN MANGGA (*Mangifera indica L.*) DI PT GALASARI GUNUNG SEJAHTERA GRESIK. Project Report. Universitas Muhammadiyah Gresik.

- Roihan, Ahmad. Sunarya, Po Abas. Rafika, Ageng Setiani. (2020). Pemanfaatan Machine Learning dalam Berbagai Bidang: Review Paper. *Indonesian Journal on Computer and Information Technology*, 5(1), 75-82.
- Saputro, Arief. Mu'min, Syahri. Lutfi, Moch. Putri, Helmanita. (2022). *Deep Transfer Learning Dengan Model Arsitektur Vgg16 Untuk Klasifikasi Jenis Varietas Tanaman Lengkeng Berdasarkan Citra Daun*. *Jurnal Mahasiswa Teknik Informatika*, 6(2):16.
- Sharma, R. R., Singh, D., & Singh, R. (2020). *Integrated management of mango anthracnose*. *International Journal of Current Microbiology and Applied Sciences*, 9(3), 2693-2701.
- Sharma, Siddharth. Sharma, Simone. Athaiya, Anidhya (2020). *ACTIVATION FUNCTIONS IN NEURAL NETWORKS*. *International Journal of Engineering Applied Sciences and Technology*. Vol. 4, Issue 12, ISSN No. 2455-2143, 310-316
- Utami, Muhlisoh Nur. Abadi, Satria. (2017). APLIKASI UNTUK MENDETEKSI PENYAKIT PADA BUAH MANGGA MENGGUNAKAN METODE FUZZY MULTIPLE CRITERIA DECISION MAKING (FMCDM). *Konferensi Mahasiswa Sistem Informasi*, Vol 5, No 1 : 391-392.
- Yuniati, Putri Dwi. (2019). *Pengaruh Pemberiaan Air Rendaman Kismis (Thompson seedless) Terhadap Peningkatan Kadar Hemoglobin Pada Mencit (Mus musculus)*. Diploma thesis, Universitas Muhammadiyah Surabaya.

LAMPIRAN

Curriculum Vitae

ABID TONDI NUGRAHA

IT ENTHUSIAST



a.tondi_



Lubuk Pakam, Deli Serdang,
Sumatera Utara



abidtondinugraha@gmail.com

PROFILE

"A computer expert committed to creating innovative solutions that address complex challenges in the world of technology. Experienced in web-based application development, as well as having in-depth knowledge of programming languages"

EXPERIENCES



PT. CICIL Solusi Mitra Teknologi (Cicil.co.id)

Function: Student Ambassador
Period: Juli 2021 to Juli 2022



PT. TELKOM REGIONAL 1 MEDAN

Function: Unit Managed Service
Operation (Fieldwork Practices)
Period: Juni 2022 to Agustus 2022



SMP SWASTA HARUM SENTOSA PERBAUNGAN

Function: School Operator
Period: January 2023 to Juli 2023

EDUCATION

2023 – Universitas Sumatera Utara
SI Ilmu Komputer

2019 – MAN 2 Deli Serdang

SELF-ABILITY

Programming Languages:
C, C++, PHP, Python, Java Script,

ORGANIZATION EXPERIENCE

UKMI AL-KHUWARIZMI USU

Role: Member of Mentoring Islamic
Teaching Division
Period : 2021/2022

PEMA FASILKOM-TI USU

Function: Member of the Public
Relations Division
Period : 2021/2022

IMILKOM USU

Function: Member of Funds &
Business Division
Period : 2021/2022

SEMINARS, WORKSHOPS AND TRAINING

REVOU (PT. Revolusi Cita Edukasi)

Participants of RevoU Mini Course:
Data Analytics "Intro to Data
Analytics", for 2 weeks and has
passed the RevoU Mini Course
Certification Test in 2023

KEDAIREKA AND PHAROS

Participants of Series-1 Workshop on
Python Introduction "Kedaireka
Matching Fund 2022: SMAPHA -
Smart Mapping for Public Health
Applications" in 2022

PT. TELKOM INDONESIA

Participants of the Digister Connect
Goes to North Sumatra Webinar
"Learning, Growing and Contributing
to Indonesia: The right place for
digital innovation champions" in
2022

IMILKOM USU

Participants of Basic Organization
Training (PDO) in 2021