

**IMPLEMENTASI MODEL *HYBRID CONVOLUTIONAL NEURAL NETWORK (CNN)–EXTREME GRADIENT BOOSTING (XGBOOST)*
DALAM KLASIFIKASI PENYAKIT KAKI GAJAH**

(Skripsi)

Oleh

**FATHIYYA JASMINE
NPM 2217051026**



**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS LAMPUNG
BANDAR LAMPUNG
2026**

**IMPLEMENTASI MODEL *HYBRID CONVOLUTIONAL NEURAL NETWORK (CNN)–EXTREME GRADIENT BOOSTING (XGBOOST)*
DALAM KLASIFIKASI PENYAKIT KAKI GAJAH**

Oleh

FATHIYYA JASMINE

Skripsi

**Sebagai Salah Satu Syarat untuk Mencapai Gelar
SARJANA KOMPUTER**

Pada

**Jurusan Ilmu Komputer
Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung**



**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS LAMPUNG
BANDAR LAMPUNG
2026**

ABSTRAK

IMPLEMENTASI MODEL *HYBRID CONVOLUTIONAL NEURAL NETWORK (CNN)–EXTREME GRADIENT BOOSTING (XGBOOST)* DALAM KLASIFIKASI PENYAKIT KAKI GAJAH

Oleh

FATHIYYA JASMINE

Penyakit kaki gajah atau *elephantiasis* merupakan kondisi pembengkakan ekstrem pada salah satu kaki yang disebabkan faktor internal maupun faktor eksternal. Penyakit kaki gajah (*elephantiasis*) memiliki gejala yang mirip antara filariasis limfatik (parasit) dan limfedema (non-parasit), sehingga terdapat kesulitan dalam membedakan kondisi fisik penyakit tersebut. Diagnosis kedua penyakit sebagian besar masih bergantung pada metode konvensional dan penelitian menggunakan pendekatan algoritma untuk penyakit kaki gajah (*elephantiasis*) masih sangat jarang dilakukan. Penelitian ini bertujuan mengembangkan model *hybrid* menggunakan CNN dan XGBoost, dengan menggabungkan kemampuan CNN untuk ekstraksi fitur dan XGBoost sebagai pengklasifikasi guna mengoptimalkan hasil. Metode *hybrid* yang digunakan juga mampu memberikan waktu pelatihan yang lebih efisien daripada model CNN murni. Evaluasi dilakukan menggunakan metode *k-fold cross validation* ($k = 10$) dengan metrik seperti *accuracy*, *precision*, *recall*, dan *f1-score*. Hasil menunjukkan bahwa model CNN murni memperoleh *mean training accuracy* sebesar 87.11% dan *mean validation accuracy* 89.83%, dengan performa terbaik pada *fold* ke-5 yang menghasilkan *accuracy* 94.87%. Model *hybrid* CNN-XGBoost menunjukkan kinerja yang lebih baik dan konsisten, dengan *mean training accuracy* 93.07% dan *mean validation accuracy* sebesar 91.83%, serta hasil terbaik pada *fold* ke-1 dengan *accuracy* 96.15%. Selain itu, model *hybrid* juga menghasilkan metrik evaluasi yang lebih tinggi dan waktu pelatihan yang lebih cepat sekitar 44–53 detik dibandingkan CNN murni sekitar 9–24 menit yang menunjukkan efektivitas kombinasi dari model CNN dan XGBoost.

Kata Kunci: *Machine Learning*, *Deep Learning*, *Hybrid*, CNN, XGBoost, Klasifikasi, Penyakit Kaki Gajah.

ABSTRACT

IMPLEMENTATION OF A HYBRID CONVOLUTIONAL NEURAL NETWORK (CNN)–EXTREME GRADIENT BOOSTING (XGBOOST) MODEL FOR THE CLASSIFICATION OF ELEPHANTIASIS

By

FATHIYYA JASMINE

Elephantiasis is a condition of extreme swelling in one leg caused by internal and external factors. Elephantiasis has symptoms similar to those of lymphatic filariasis (parasitic) and lymphedema (non-parasitic), making it difficult to distinguish between the two conditions. Diagnosis of both diseases still largely relies on conventional methods, and research using algorithmic approaches for elephantiasis is still very rare. This study aims to develop a hybrid model using CNN and XGBoost, combining the capabilities of CNN for feature extraction and XGBoost as a classifier to optimize results. The hybrid method used is also capable of providing more efficient training time than pure CNN models. Evaluation was performed using the k-fold cross validation method ($k = 10$) with metrics such as accuracy, precision, recall, and f1-score. The results show that the pure CNN model obtained a mean training accuracy of 87.11% and a mean validation accuracy of 89.83%, with the best performance on the 5th fold, which produced an accuracy of 94.87%. The hybrid CNN-XGBoost model showed better and more consistent performance, with a mean training accuracy of 93.07% and a mean validation accuracy of 91.83%, as well as the best results on the 1st fold with an accuracy of 96.15%. In addition, the hybrid model also produced higher evaluation metrics and faster training times of around 44–53 seconds compared to pure CNNs of around 9–24 minutes, demonstrating the effectiveness of the combination of CNN and XGBoost models.

Keywords: Machine Learning, Deep Learning, Hybrid, CNN, XGBoost, Classification, Elephant Foot Disease.

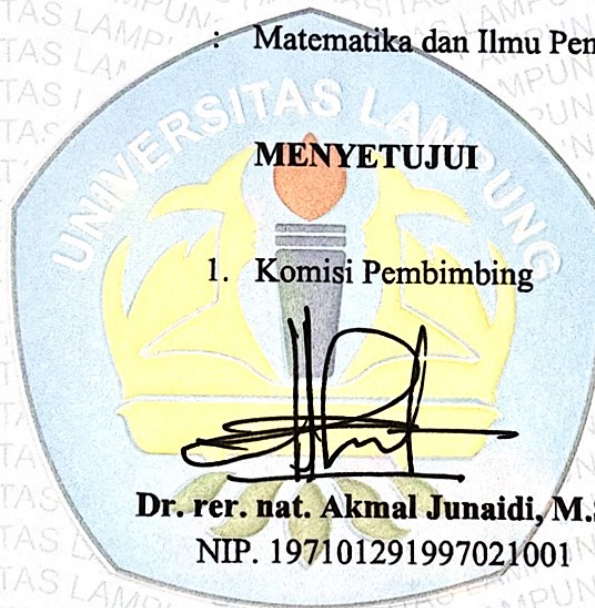
Judul Skripsi : **IMPLEMENTASI MODEL *HYBRID CONVOLUTIONAL NEURAL NETWORK (CNN)-EXTREME GRADIENT BOOSTING (XGBOOST)* DALAM KLASIFIKASI PENYAKIT KAKI GAJAH**

Nama Mahasiswa : **Fathiyya Jasmine**

Nomor Pokok Mahasiswa : **2217051026**

Program Studi : **S1-Ilmu Komputer**

Fakultas : **Matematika dan Ilmu Pengetahuan Alam**



2. **Ketua Jurusan Ilmu Komputer**

3. **Ketua Program Studi Ilmu Komputer**

[Signature]
Dwi Sakethi, S.Si., M.Kom.
NIP. 196806111998021001

[Signature]
Tristiyanto, S.Kom., M.I.S., Ph.D.
NIP. 198104142005011001

MENGESAHKAN

1. Tim Penguji

Ketua : Dr. rer. nat. Akmal Junaidi, M.Sc.

Penguji

Penguji : Dewi Asiah Shofiana, S.Komp., M.Kom.

Pembahas I

Penguji : Rahman Taufik, S.Pd., M.Kom.

Pembahas II



2. Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam



Dr. Eng. Heri Satria, S.Si., M.Si.

NIP. 197110012005011002

Tanggal Lulus Ujian Skripsi: 12 Maret 2026

PERNYATAAN

Saya bertanda tangan di bawah ini, menyatakan bahwa skripsi saya yang berjudul **“Implementasi Model *Hybrid Convolutional Neural Network (CNN)*–*Extreme Gradient Boosting (XGBoost)* dalam Klasifikasi Penyakit Kaki Gajah”** merupakan hasil karya saya sendiri dan bukan karya orang lain. Seluruh penulisan yang terdapat dalam skripsi ini telah mematuhi aturan penulisan karya akademik dari Universitas Lampung. Apabila dikemudian hari ditemukan bahwa skripsi saya adalah hasil jiplakan atau dibuat orang lain, maka saya siap menerima segala konsekuensi berupa pencabutan gelar yang saya terima.

Bandar Lampung, 07 April 2026



Fathiyya Jasmine

NPM. 2217051026

RIWAYAT HIDUP



Penulis lahir di Bandar Lampung pada tanggal 13 September 2004 sebagai anak ketiga dari pasangan Bapak Saleh dan Ibu Fajarita. Penulis menempuh pendidikan formal dimulai dari Taman Kanak-Kanak di Dharma Wanita Kota Agung dan lulus pada Tahun 2010. Penulis melanjutkan pendidikan Sekolah Dasar (SD) di MIN 1 Kota Agung dan lulus pada Tahun 2016. Kemudian, pendidikan Sekolah Menengah Pertama (SMP) di MTs Negeri 1 Kota Agung hingga Tahun 2019, lalu melanjutkan pendidikan Sekolah Menengah Atas (SMA) di SMA Negeri 1 Kota Agung yang diselesaikan pada Tahun 2022.

Pada tahun 2022 penulis terdaftar menjadi mahasiswa Jurusan Ilmu Komputer, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Lampung melalui jalur SNMPTN. Selama menjalani masa perkuliahan, penulis melakukan berbagai kegiatan akademik maupun nonakademik sebagai berikut.

1. Menjadi Anggota Biro Kesekretariatan Himpunan Mahasiswa Jurusan Ilmu Komputer FMIPA Unila pada periode 2023/2024.
2. Menjadi Sekretaris Pelaksana pada program kerja Pelatihan SPIA dan SPIK Himakom pada Tahun 2023.
3. Menjadi Pengurus sebagai Sekretaris Biro Kesekretariatan Himpunan Mahasiswa Jurusan Ilmu Komputer Tahun 2024.
4. Menjadi Asisten Dosen mata kuliah Matematika Diskrit di Jurusan Ilmu Komputer Tahun 2024.
5. Menjadi Asisten Dosen mata kuliah Komunikasi Data dan Jaringan Komputer Tahun 2024.

6. Mengikuti Program MSIB Kampus Merdeka di Bangkit *Academy* led by Google, Tokopedia, Gojek, & Traveloka Batch 2 periode 2024/2025 pada jalur pembelajaran (*learning path*) *Cloud Computing*.
7. Menjadi Asisten Dosen mata kuliah Internet Of Things pada Tahun 2025.
8. Mengikuti Kuliah Kerja Nyata (KKN) di Desa Palembang, Kecamatan Kalianda, Kabupaten Sumatera Selatan.
9. Melaksanakan Kerja Praktik (KP) di Bank Lampung Pusat dengan fokus pada pengembangan manajemen *website* surat pada Divisi Konsumer Tahun 2025.

MOTTO

لَا يُكَلِّفُ اللَّهُ نَفْسًا إِلَّا وُسْعَهَا

“Allah tidak membebani seseorang melainkan sesuai dengan kesanggupannya”

(QS. Al-Baqarah: 286)

“Orang yang paling bahagia adalah yang hidup sederhana, namun penuh syukur.”

- Ali bin Abi Thalib

“Berproses dengan sabar, bertumbuh dengan ilmu, dan berserah kepada Allah.”

PERSEMBAHAN

Alhamdulillahirobbilalamin

Puji syukur kehadiran Allah Subhanahu Wa Ta'ala atas segala limpahan rahmat dan karunia-Nya, sehingga skripsi ini berhasil diselesaikan dengan baik. Shalawat serta salam selalu tercurahkan kepada Nabi Muhamad Shallahu 'Alaihi Wasallam sebagai suri teladan bagi seluruh umat manusia.

Kupersembahkan karya ini kepada:

Kedua Orang Tua Tercinta

Atas semua doa, dukungan, kasih, serta pengorbanan yang tidak pernah berhenti.

Terima kasih atas setiap usaha, bimbingan, dan kepercayaan yang menjadi landasan dalam setiap langkah penulis.

Seluruh Keluarga Besar Ilmu Komputer 2022

Atas semua semangat dan dukungan selama proses perkuliahan berlangsung.

Almamater Tercinta, Jurusan Ilmu Komputer FMIPA Universitas Lampung

Sebagai tempat belajar, bertumbuh, dan memperoleh pengetahuan yang menjadi bekal untuk masa depan.

SANWANCANA

Puji syukur kehadiran Allah Subhanahu Wa Ta'ala, yang telah memberikan nikmat kesehatan, rahmat, dan hidayah-Nya, sehingga penulis dapat menyelesaikan skripsi yang berjudul “Implementasi Model *Hybrid Convolutional Neural Network* (CNN) dan *Extreme Gradient Boosting* (XGBoost) Dalam Klasifikasi Penyakit Kaki Gajah” dengan lancar. Terima kasih penulis ucapkan kepada semua pihak yang telah memberikan dukungan, bimbingan, dan arahan dalam penyusunan Skripsi ini. Oleh karena itu, penulis mengucapkan terima kasih yang sebesar-besarnya kepada:

1. Allah SWT atas segala rahmat, pertolongan, dan kemudahan yang diberikan, sehingga setiap proses dapat berjalan dengan baik dan lancar.
2. Kedua orang tua tercinta, Ayah Saleh dan Umi Fajarita yang selalu memberikan kasih sayang, doa, dan dukungan, sehingga penulis dapat menyelesaikan perkuliahan dan penyusunan skripsi ini. Terima kasih selalu ada di saat penulis membutuhkan kepercayaan dan dukungan, serta senantiasa percaya pada setiap langkah penulis lakukan.
3. Bapak Dwi Sakethi, S.Si., M.Kom. selaku Ketua Jurusan Ilmu Komputer FMIPA Universitas Lampung.
4. Ibu Yunda Heningtyas, M.Kom. selaku Sekretaris Jurusan Ilmu Komputer FMIPA Universitas Lampung.
5. Bapak Tristiyanto, S.Kom., M.I.S., Ph.D. selaku Kepala Program Studi S1 Ilmu Komputer FMIPA Universitas Lampung.
6. Bapak Dr. rer. Nat. Akmal Junaidi, M.Sc selaku Pembimbing Utama yang telah meluangkan waktu dalam memberikan bimbingan, arahan, kritik, serta berbagai masukan yang membantu proses penyelesaian penelitian skripsi ini.
7. Ibu Dewi Asiah Shofiana, S.Komp., M.Kom. selaku Pembahas Pertama yang telah memberikan masukan dan saran yang bermanfaat, sehingga penulis dapat memperbaiki penelitian menjadi lebih baik.

8. Bapak Rahman Taufik S.Pd., M.Kom. sebagai Pembahas Kedua yang telah memberikan kritik dan saran, sehingga penelitian ini dapat disempurnakan menjadi lebih baik.
9. Bapak dan Ibu Dosen Jurusan Ilmu Komputer FMIPA Universitas Lampung yang telah memberikan banyak ilmu dan motivasi selama perkuliahan.
10. Seluruh staf dan karyawan Jurusan Ilmu Komputer FMIPA Universitas Lampung, Ibu Ade Nora Maela, Pak Zainuddin, Mas Nofal, Mas Sam, dan Pak Dahud yang telah membantu perihal administrasi, sarana, dan prasarana.
11. Teman seperjuangan dan terbaik selama perkuliahan yaitu Citra, Elen, Atikah, Yayas, Nazwa, dan Pia yang selalu menemani, membantu, mendukung, mendengar keluh kesah, menghibur, dan berbagi cerita dari awal semester.
12. Himpunan Mahasiswa Jurusan Ilmu Komputer Periode 2023 & 2024, atas pengalaman dan kebersamaan selama berorganisasi dan masa perkuliahan.
13. Abang Rayhan dan Kakak Fira selaku saudara kandung, serta Keluarga Tetra Isnaini yang telah memberikan doa, semangat, dan dukungan untuk mengerjakan dan menyelesaikan skripsi dengan baik.
14. Untuk diri sendiri, yaitu Fathiyya Jasmine yang selalu berusaha dan tidak menyerah selama perkuliahan dan penyusunan skripsi berlangsung.

Penulis menyadari bahwa masih terdapat kekurangan dalam penyusunan skripsi ini. Namun, penulis berharap penelitian ini dapat memberikan manfaat dan menjadi referensi bagi para pembaca. Semoga penelitian ini dapat menjadi langkah awal untuk penelitian yang lebih baik di masa mendatang.

Bandar Lampung, 07 April 2026



Fathiyya Jasmine

NPM. 2217051026

DAFTAR ISI

DAFTAR ISI	xiii
DAFTAR TABEL	xvi
DAFTAR GAMBAR	xviii
I. PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	5
1.3 Batasan Masalah.....	5
1.4 Tujuan Penelitian.....	5
1.5 Manfaat Penelitian	6
II. TINJAUAN PUSTAKA	7
2.1 Penelitian Terdahulu.....	7
2.2 <i>Artificial Intelligence</i> (AI).....	10
2.3 <i>Machine learning</i> (ML).....	11
2.4 <i>Deep Learning</i>	12
2.5 <i>Computer Vision</i>	13
2.6 Pengolahan Citra Digital	13
2.7 <i>Convolutional Neural Network</i> (CNN)	15
2.8 <i>Residual Network</i> (ResNet).....	19
2.9 <i>Extreme Gradient Boosting</i> (XGBoost)	20
2.10 <i>Hybrid Convolutional Neural Network</i> dan <i>Extreme Gradient Boosting</i> (CNN-XGBoost)	23
2.11 Penyakit Kaki Gajah (<i>Elephantiasis</i>)	24
2.11.1 Filariasis Limfatik (Parasit).....	24
2.11.2 Limfedema (Non-Parasit).....	25
2.12 <i>Bayesian Optimization</i> (Optuna).....	26
2.13 <i>Cross Validation</i>	27

2.14 <i>Confusion Matrix</i>	28
III. METODOLOGI PENELITIAN	30
3.1 Tempat dan Waktu Penelitian.....	30
3.1.1 Tempat Penelitian.....	30
3.1.2 Waktu Penelitian	30
3.2 Alat Pendukung.....	31
3.2.1 Perangkat Keras (<i>Hardware</i>).....	31
3.2.2 Perangkat Lunak (<i>Software</i>).....	31
3.3 Tahapan Penelitian	33
3.4 Pengumpulan Data	34
3.5 <i>Preprocessing</i>	34
3.5.1 Hapus Latar (<i>Remove Background</i>)	34
3.5.2 <i>Crop & Resize</i>	35
3.5.3 <i>Split Data</i>	35
3.5.4 Augmentasi Data	36
3.5.5 Normalisasi Citra	36
3.6 <i>Convolution Neural Network</i> (CNN).....	37
3.7 <i>Extreme Gradient Boosting</i> (XGBoost).....	37
3.8 Model	37
3.8.1 <i>Hyperparameter</i> Model CNN dan <i>Hybrid</i> CNN-XGBoost	38
3.9 <i>Evaluation</i> Model.....	39
IV. HASIL DAN PEMBAHASAN	41
4.1 <i>Dataset</i> Kaki Gajah (<i>Elephantiasis</i>).....	41
4.2 <i>Preprocessing</i>	42
4.2.1 Hapus Latar (<i>Remove Background</i>)	42
4.2.2 <i>Crop & Resize</i>	43
4.2.3 <i>Split Data</i>	44
4.2.4 Augmentasi Data	45
4.2.5 Normalisasi Citra	46
4.3 Skenario pada Model CNN dan <i>Hybrid</i> CNN-XGBoost	48
4.4 <i>Training Convolutional Neural Network</i> (CNN)	48
4.4.1 <i>Hyperparameter</i> CNN.....	51

4.4.2 Hasil <i>Training</i> CNN.....	53
4.5 <i>Training Hybrid Convolutional Neural Network dan Extreme Gradient Boosting</i> (CNN-XGBoost).....	61
4.5.1 <i>Hyperparameter XGBoost</i>	64
4.5.2 Hasil <i>Training Hybrid Convolutional Neural Network dan Extreme Gradient Boosting</i> (CNN-XGBoost).....	66
4.6 <i>Testing Model</i>	74
4.7 <i>Evaluasi Model</i>	76
V. SIMPULAN DAN SARAN	87
5.1 <i>Simpulan</i>	87
5.2 <i>Saran</i>	88
DAFTAR PUSTAKA	89

DAFTAR TABEL

Tabel	Halaman
1. Penelitian Terdahulu	7
2. <i>Confusion Matrix</i>	29
3. Jadwal Pelaksanaan Penelitian	30
4. Hyperparameter yang Digunakan pada Kedua Model	38
5. Rincian Pembagian Dataset	44
6. Contoh Perhitungan Normalisasi	47
7. Skenario pada Model CNN dan <i>Hybrid</i> CNN-XGBoost	48
8. Lapisan Arsitektur Model CNN	50
9. <i>Hyperparameter Manual Tuning</i> pada Model CNN	51
10. <i>Search Space</i> Optuna pada Model CNN	52
11. Hasil <i>Hyperparameter Tuning</i> Optuna pada model CNN	53
12. Hasil <i>Training</i> CNN Non-Augmentasi Skenario Pertama	53
13. Hasil <i>Training Fold</i> CNN Non-Augmentasi Skenario Pertama	55
14. Hasil <i>Training</i> CNN Non-Augmentasi Skenario Kedua	55
15. Hasil <i>Training Fold</i> CNN Non-Augmentasi Skenario Kedua	57
16. Hasil <i>Training</i> CNN Augmentasi Skenario Pertama	57
17. Hasil <i>Training Fold</i> CNN Augmentasi Skenario Pertama	59
18. Hasil <i>Training Fold</i> CNN Augmentasi Skenario Kedua	59
19. Hasil <i>Training Fold</i> CNN Augmentasi Skenario Kedua	60
20. Hyperparameter pada Model XGBoost	64
21. <i>Search Space</i> Optuna pada Model Hybrid CNN-XGBoost	65
22. Hasil <i>Hyperparameter Tuning</i> Optuna pada Model <i>Hybrid</i> CNN-XGBoost ..	66
23. Hasil <i>Training Hybrid</i> CNN-XGBoost Non-Augmentasi Skenario Pertama ..	67
24. Hasil <i>Training Fold Hybrid</i> CNN-XGBoost Augmentasi Skenario Pertama ..	68
25. Hasil <i>Training Hybrid</i> CNN-XGBoost Non-Augmentasi Skenario Kedua	69

26. Hasil <i>Training Fold Hybrid</i> CNN-XGBoost Non-Augmentasi Skenario Pertama.....	70
27. Hasil <i>Training Hybrid</i> CNN-XGBoost Augmentasi Skenario Pertama	71
28. Hasil <i>Training Fold Hybrid</i> CNN-XGBoost Augmentasi Skenario Pertama..	72
29. Hasil <i>Training Hybrid</i> CNN-XGBoost Augmentasi Skenario Kedua	73
30. Hasil Evaluasi Model CNN	77
31. Performa Model CNN Per Kelas	78
32. Hasil Evaluasi Model <i>Hybrid</i> CNN-XGBoost	79
33. Performa Model <i>Hybrid</i> CNN-XGBoost Per Kelas	81

DAFTAR GAMBAR

Gambar	Halaman
1. Penyakit <i>Elephantiasis</i> : (a) Filariasis Limfatik (Parasit) dan (b) Limfedema (Non-Parasit).....	3
2. Hubungan AI, ML, dan DL	11
3. Arsitektur <i>Deep Neural Network</i>	12
4. Representasi Warna Citra Digital	14
5. Arsitektur Algoritma <i>Convolutional Neural Network</i> (CNN).....	15
6. Proses <i>Convolutional Layer</i> : (a) Input Matriks, (b) <i>Kernel</i> , (c) <i>Feature Map</i>	16
7. Operasi <i>Max</i> dan <i>Average</i> pada <i>Pooling Layer</i>	17
8. <i>Flatten</i> dan <i>Fully Connected Layer</i>	18
9. Arsitektur ResNet50.....	20
10. Cara Kerja Algoritma XGBoost	20
11. Arsitektur CNN-XGBoost.....	24
12. Cara Kerja <i>Cross Validation</i>	28
13. Tahapan Penelitian	33
14. Jumlah <i>Dataset</i> Perkelas	41
15. Hasil Tahap Hapus Latar (<i>Remove Background</i>)	42
16. Hasil Tahap <i>Crop</i>	43
17. Hasil Tahap <i>Resize</i>	44
18. Contoh Hasil Augmentasi Data	46
19. Model dasar dan Lapisan <i>Classifier</i> CNN.....	49
20. Grafik <i>Training</i> CNN Non-Augmentasi pada Skenario Pertama.....	54
21. Grafik <i>Training</i> CNN Non-Augmentasi Skenario Kedua	56
22. Grafik <i>Training</i> CNN Augmentasi Skenario Pertama.....	58
23. Grafik <i>Training</i> CNN Augmentasi Skenario Kedua	60
24. Alur <i>Hybrid</i> CNN-XGBoost	61

25. Model Dasar <i>Feature Extractor</i> \.....	62
26. <i>Training</i> Pada Model XGBoost.....	63
27. Grafik <i>Training Hybrid CNN-XGBoost</i> Augmentasi Skenario Pertama.....	67
28. Grafik <i>Training Hybrid CNN-XGBoost</i> Non-Augmentasi Skenario Kedua ..	69
29. Grafik <i>Training Hybrid CNN-XGBoost</i> Augmentasi Skenario Pertama.....	71
30. Grafik <i>Training Hybrid CNN-XGBoost</i> Augmentasi Skenario Kedua.....	73
31. Testing Model pada Penyakit Filariasis Limfatik (Parasit).....	75
32. Testing Model pada Penyakit Limfedema (Non-Parasit).....	76
33. <i>Confusion Matrix</i> Model CNN.....	78
34. <i>Confusion Matrix</i> Model <i>Hybrid CNN-XGBoost</i>	80
35. Analisis Hasil Klasifikasi Salah.....	82
36. Filariasis Limfatik dan Limfedema yang Mempengaruhi Hasil Klasifikasi Model.	83
37. Grafik Perbandingan Performa Model CNN dan <i>Hybrid CNN-XGBoost</i>	84

I. PENDAHULUAN

1.1 Latar Belakang

Penyakit kaki gajah atau yang disebut sebagai *elephantiasis* merupakan istilah medis untuk menggambarkan pembengkakan ekstrem pada tungkai atau anggota tubuh lainnya yang terjadi karena masalah pada sistem limfatik (Singh *et al.*, 2024). Sistem limfatik berperan penting dalam mempertahankan keseimbangan cairan tubuh serta mendukung fungsi imun dalam melindungi tubuh dari patogen. Gangguan pada sistem ini dapat menyebabkan penumpukan cairan limfatik yang tidak terdistribusi dengan baik. Akibatnya, terjadi kelainan pada pembuluh dan kelenjar limfatik karena pembengkakan jaringan yang semakin membesar seiring waktu (Lee *et al.*, 2022). Pembengkakan yang menetap dapat menimbulkan ketidaknyamanan, keterbatasan dalam bergerak, dan perubahan fisik yang berpengaruh pada kualitas hidup penderitanya. Kondisi ini pada dasarnya juga dapat terjadi karena dua faktor utama, yaitu infeksi parasit berupa cacing filaria dan juga faktor non-parasit yang disebabkan seperti operasi kanker, radiasi terapi kanker, atau kelainan bawaan yang memicu terjadinya limfedema (Mackenzie *et al.*, 2024).

Elephantiasis pada sebagian besar kasus disebabkan oleh penyakit tropis terabaikan (*Neglected Tropical Diseases/NTDs*) yang merupakan kelompok penyakit menular dan melemahkan yang mempengaruhi jutaan orang di daerah tropis dan subtropis, terutama di negara-negara berkembang dengan sanitasi buruk dan infrastruktur kesehatan terbatas (Branda *et al.*, 2025).

Filariasi limfatik, adalah salah satu NTD yang paling umum dan berdampak besar di berbagai negara berkembang. Penyakit ini terjadi karena infeksi cacing filaria, yaitu *Wuchereria bancrofti*, *Brugia malayi*, dan *Brugia timori* (Rahmi *et al.*, 2022). Cacing filaria, terutama *Wuchereria bancrofti*, yang ditularkan melalui gigitan nyamuk dan menyerang sistem limfatik, menyebabkan pembengkakan ekstrem pada anggota tubuh, terutama kaki yang berujung pada kelainan bentuk dan kecacatan. Menurut WHO per tahun 2014, sekitar 120 juta orang di seluruh dunia sudah terinfeksi parasit ini.

Selain filariasis limfatik yang disebabkan oleh infeksi parasit, kondisi *elephantiasis* juga dapat terjadi akibat non-parasit yaitu limfedema. Limfedema (non-parasit) merupakan kondisi kronis dan progresif akibat penumpukan cairan limfatik di ruang interstitial, sehingga menyebabkan pembengkakan lokal (Primasari, 2020). Kondisi ini dibedakan menjadi dua jenis, yaitu limfedema primer yang terjadi karena kelainan bawaan atau genetik sejak lahir dan limfedema sekunder yang terjadi akibat faktor eksternal seperti trauma, operasi kanker, terapi radiasi, infeksi, obesitas atau penyebab lainnya (Duhon *et al.*, 2022). Berdasarkan *International Society of Lymphology* (ISL), limfedema pada tahap 3 (mulai tahap 0 hingga 3) dalam klasifikasi klinis termasuk sebagai *elephantiasis*. Tahap ini ditandai dengan perubahan pada kulit, seperti akantosis, penumpukan lemak, fibrosis, serta pertumbuhan kulit yang mirip dengan kutil dengan atau tanpa adanya pembengkakan (Damstra *et al.*, 2020).

Penelitian ini menjadi penting, karena penelitian terkait analisis algoritma dengan topik penyakit kaki gajah (*elephantiasis*) masih sangat jarang dilakukan. Sebagian besar penelitian terdahulu lebih banyak berfokus pada penyakit umum, sehingga topik ini masih kurang mendapat perhatian. Diagnosis filariasis limfatik hingga saat ini masih dilakukan secara tradisional melalui pemeriksaan mikroskopis sampel darah (Osei-Poku *et al.*, 2025). Metode tersebut memiliki sejumlah keterbatasan, seperti ketergantungan pada waktu pengambilan sampel, pemeriksaan yang

memakan waktu, dan kualitas peralatan yang digunakan. Sementara itu, diagnosis limfedema (non-parasit) juga masih bergantung pada pemeriksaan klinis dan fisik yang komprehensif (Borman, 2018).

Pada limfedema (non-parasit) tahap lanjut sering kali sulit dibedakan dengan filariasis limfatik, karena keduanya sama-sama menimbulkan pembesaran ekstrem, perubahan tekstur kulit, dan muncul lipatan dalam (seperti kulit kaki gajah) seperti yang ditunjukkan pada Gambar 1.



Gambar 1. Penyakit *Elephantiasis*: (a) Filariasis Limfatik (Parasit) dan (b) Limfedema (Non-Parasit). Sumber Citra: (a.1) Narahari, CC BY-NC-SA 4.0; (a.2) Kapoor; (b.1 & b.2) Chanwimalueang *et al.*, CC BY. Gambar dipotong dari gambar aslinya.

Kondisi ini menjadikan keduanya sulit dibedakan hanya melalui pemeriksaan fisik, terutama di daerah dengan keterbatasan fasilitas laboratorium. Pengembangan metode diagnostik berbasis citra digital dan kecerdasan buatan dapat membantu mengatasi tantangan tersebut (Kawashima *et al.*, 2025). Oleh karena itu, model klasifikasi dalam penelitian ini dikembangkan sebagai alat pendukung *screening* berdasarkan citra visual, untuk membedakan apakah kondisi kaki menunjukkan tanda-tanda klinis khas filariasis limfatik (parasit) atau limfedema (non-parasit). Efektivitas pendekatan ini juga diperkuat oleh berbagai penelitian sebelumnya yang mengindikasikan bahwa model *deep learning* dapat mencapai akurasi tinggi dalam mendeteksi berbagai penyakit dari citra medis (Damstra *et al.*, 2020).

Convolutional Neural Network (CNN), sebagai salah satu metode *deep learning* banyak digunakan dalam analisis citra medis untuk diagnosis berbagai penyakit (Izza *et al.*, 2024). CNN dapat mengekstrak karakteristik

gambar secara otomatis tanpa perlu proses ekstraksi manual, sehingga klasifikasi menjadi lebih efisien dan akurat karena kemampuan model belajar dari data (Hidayat & Jemakmun, 2025). Namun, CNN memiliki kelemahan, seperti kebutuhan jumlah data besar, proses *training* yang cukup memakan waktu, dan risiko *overfitting* (Akbar *et al.*, 2024). Untuk mengatasi kelemahan ini, pendekatan *hybrid* yang menggabungkan algoritma CNN dengan *Extreme Gradient Boosting* (XGBoost) semakin banyak digunakan. XGBoost dikenal sebagai algoritma *boosting* modern yang efektif dalam memodelkan sistem yang kompleks (Ali *et al.*, 2023). XGBoost melakukan *boosting* terhadap fitur visual yang diekstraksi oleh CNN, sehingga mengkombinasi kekuatan CNN dalam mengenali pola visual dengan kemampuan XGBoost dalam menghasilkan klasifikasi yang lebih akurat dan tahan terhadap *overfitting* (Jebastine, 2024). Beberapa penelitian juga menunjukkan bahwa model *hybrid* ini mampu meningkatkan akurasi klasifikasi dalam berbagai aplikasi medis.

Pendekatan model *hybrid* CNN-XGBoost untuk klasifikasi penyakit *elephantiasis* pada citra kaki memberikan potensi besar untuk meningkatkan akurasi dan efektivitas dalam membedakan antara kasus kaki akibat filariasis limfatik (parasit) dan limfedema (non-parasit). Dengan menggunakan citra kaki sebagai *input*, CNN dapat mengekstraksi karakteristik fitur visual yang terkait, seperti pola tekstur kulit, tingkat pembengkakan, dan warna kulit. Fitur-fitur ini kemudian digunakan oleh XGBoost untuk mengklasifikasikan citra sebagai positif atau negatif terhadap infeksi filariasis. Selain itu, model *hybrid* ini lebih tahan terhadap *overfitting* dan dapat bekerja dengan baik pada *dataset* yang relatif kecil (Ajiboye, 2024). Dengan demikian, pendekatan *hybrid* CNN-XGBoost menjadi pilihan yang menjanjikan untuk mengklasifikasikan kaki akibat filariasis limfatik (parasit) dan limfedema (non-parasit) berdasarkan citra kaki. Model *hybrid* CNN-XGBoost dapat menjadi alat diagnostik yang efektif dan terjangkau untuk mendukung program eliminasi filariasis.

1.2 Rumusan Masalah

Berdasarkan penjelasan latar belakang di atas, maka rumusan masalah dalam penelitian ini adalah:

1. Bagaimana model CNN dan XGBoost dikombinasikan dalam mengklasifikasi citra kaki antara kasus akibat filariasis limfatik (parasit) dan limfedema (non-parasit)?
2. Bagaimana akurasi dan efektivitas model *hybrid* CNN-XGBoost dalam mengklasifikasikan penyakit *elephantiasis*?
3. Bagaimana pengaruh penerapan augmentasi data pada model terhadap peningkatan performa klasifikasi?

1.3 Batasan Masalah

Untuk menjaga penelitian tetap terfokus, maka penelitian ini dibatasi dengan sebagai berikut:

1. Penelitian menggunakan bahasa pemrograman python dengan mengimplementasikan model CNN murni dan *hybrid* CNN-XGBoost.
2. *Dataset* yang digunakan terdiri dari dua kelas, yaitu citra kaki akibat filariasis limfatik (parasit) dan limfedema (non-parasit).
3. Input yang digunakan hanya berupa citra kaki, tanpa informasi tambahan seperti data klinis atau riwayat medis.

1.4 Tujuan Penelitian

Tujuan yang ingin dicapai dari penelitian ini adalah sebagai berikut:

1. Mengembangkan model klasifikasi penyakit *elephantiasis* berbasis *hybrid* CNN-XGBoost.
2. Melakukan evaluasi kinerja model CNN murni dan *hybrid* CNN-XGBoost menggunakan *k-fold cross validation*, *confusion matrix*, serta metrik *accuracy*, *precision*, *recall*, dan *f1-score*.
3. Membandingkan kinerja model *hybrid* CNN-XGBoost dengan model CNN murni dalam klasifikasi citra.

1.5 Manfaat Penelitian

Manfaat dari penelitian ini adalah sebagai berikut:

1. Memberikan kontribusi ilmiah karena penelitian terkait penerapan algoritma pada penyakit *elephantiasis* masih sangat sedikit, sehingga penggunaan algoritma *hybrid* CNN-XGBoost dapat memperluas cakupan riset di bidang klasifikasi citra medis.
2. Membantu membedakan kasus kaki akibat filariasis limfatik (parasit) dengan limfedema (non-parasit) yang hampir serupa secara visual.
3. Membuktikan efektivitas kombinasi CNN dan XGBoost untuk klasifikasi penyakit berbasis citra.

II. TINJAUAN PUSTAKA

2.1 Penelitian Terdahulu

Berikut merupakan ringkasan penelitian terdahulu yang relevan sebagai referensi dalam penelitian ini, seperti yang terlihat pada Tabel 1.

Tabel 1. Penelitian Terdahulu

Penelitian	Metode	Hasil
<i>Deep Learning-Based Classification of Lymphedema and Other Lower Limb Edema Diseases Using Clinical Images</i> (Lewsirat <i>et al.</i> , 2025)	CNN	Penelitian ini menunjukkan bahwa model CNN dengan arsitektur EfficientNetV2 mencapai akurasi sebesar 78.6%, lebih tinggi dari akurasi tenaga medis sebesar 62,7%.
Implementasi Algoritma YOLOv3 Menggunakan Fitur Ekstraktor ResNeXt Untuk Deteksi Filariasis (Aswir & Wasilah, 2024)	YOLOv3	Penelitian ini menghasilkan algoritma YOLOv3 dengan fitur ekstraksi ResNeXt mampu memberikan performa tinggi, dengan capaian akurasi rata-rata 96.77% dalam proses deteksi objek.
Metode <i>Convolutional Neural Network</i> dan <i>Extreme Gradient XGBoost</i> untuk Mengklasifikasi Penyakit Pneumonia (Putra <i>et al.</i> , 2024)	<i>Hybrid</i> CNN-XGBoost	Penelitian ini menghasilkan temuan bahwa pada skenario 70:30, model CNN mencapai akurasi sebesar 88.73%, sedangkan model <i>hybrid</i> dengan XGBoost sebesar 91.12%.

Penelitian	Metode	Hasil
<i>Hybrid Skin Lesion Detection Integrating CNN and XGBoost for Accurate Diagnosis</i> (Ajiboye, 2024)	Hybrid CNN-XGBoost	Penelitian ini menghasilkan bahwa model <i>hybrid</i> CNN-XGBoost mencapai akurasi sebesar 86.46%, lebih besar daripada model CNN murni yang memperoleh akurasi sebesar 84.29%.
<i>An Implementation of Hybrid CNN-XGBoost Method for Leukemia Detection</i> (Hidayat et al., 2023)	Hybrid CNN-XGBoost	Penelitian ini menunjukkan bahwa pendekatan <i>hybrid</i> CNN-XGBoost menghasilkan akurasi sebesar 85.43% dari 85.32%, dibandingkan dengan klasifikasi model CNN murni.

Penelitian oleh Lewsirirat et al. (2025) yang berjudul “*Deep Learning-Based Classification Of Lymphedema And Other Lower Limb Edema Diseases Using Clinical Images*”, berfokus pada pengembangan model kecerdasan buatan untuk mengklasifikasikan kondisi pembengkakan kaki, termasuk kaki normal, limfedema, CVI & DVT, dan penyakit sistemik. Penelitian ini menggunakan 1.622 citra dan melatih 16 model pembelajaran mendalam, termasuk CNN (AlexNet, GoogLeNet, ResNet50, VGG16, MobileNetV3, DenseNet169, SqueezeNet, dan EfficientNetV2) serta transformer. Di antara semua model yang diuji, EfficientNetV2 menunjukkan performa terbaik dengan akurasi 78,6%, melampaui tingkat akurasi dokter yang hanya mampu meraih 62,7% dalam evaluasi klinis.

Pada penelitian “Implementasi Algoritma YOLOv3 Menggunakan Fitur Ekstraktor ResNeXt Untuk Deteksi Filariasis” oleh Aswir & Wasilah (2024), mengembangkan model deteksi kaki filariasis (*elephantiasis*) dengan mengintegrasikan YOLOv3 dengan fitur ekstaktor ResNeXt. *Dataset* pada penelitian tersebut berjumlah 150 citra yang dibagi ke dalam data latih (70%), data validasi (10%), dan data uji (20%). Hasil penelitian menunjukkan bahwa

metode YOLOv3 dengan fitur ekstraktor ResNeXt berhasil mencapai akurasi rata-rata 96.77%, menjadikannya cepat dan akurat dalam mendiagnosis dini penderita filariasis (*elephantiasis*). Oleh karena itu, *deep learning* sangat efektif dalam menganalisis citra medis penyakit filariasis.

Kemudian, penelitian oleh Putra *et al.* (2024) yang berjudul "Metode *Convolutional Neural Network* dan *Extreme Gradient XGBoost* untuk Mengklasifikasi Penyakit Pneumonia" bertujuan untuk mengklasifikasikan pasien penderita pneumonia dan pasien normal dengan memanfaatkan CNN untuk ekstraksi fitur dan XGBoost untuk klasifikasi. *Dataset* yang digunakan berasal dari Kaggle sebanyak 5.856 citra, dengan pembagian data data latih (70%) dan data uji (30%). Hasil menunjukkan bahwa model CNN mencapai akurasi sebesar 88,73%, sedangkan pendekatan *hybrid* CNN dengan XGBoost berhasil meningkat menjadi 91.12%. Peningkatan ini membuktikan besarnya potensi pendekatan *hybrid* CNN-XGBoost.

Selanjutnya, penelitian yang dilakukan oleh Ajiboye (2024) yang berjudul "Hybrid Skin Lesion Detection Integrating CNN and XGBoost for Accurate Diagnosis" berfokus pada klasifikasi lesi kulit dengan mengombinasikan CNN dan XGBoost pada *dataset* HAM10000 yang terdiri dari 10.015 citra dermatoskopik dari tujuh kelas lesi kulit, di mana beberapa di antaranya termasuk jenis kanker kulit. Hasil penelitian ini menunjukkan bahwa model *hybrid* mampu mencapai akurasi sebesar 86.46% dan ROC-AUC rata-rata 0.98, melampaui akurasi model CNN murni sebesar 84.29%. Penelitian ini juga menyimpulkan bahwa pendekatan *hybrid* dapat membantu dokter kulit dalam mendiagnosis dari tujuh kelas lesi kulit.

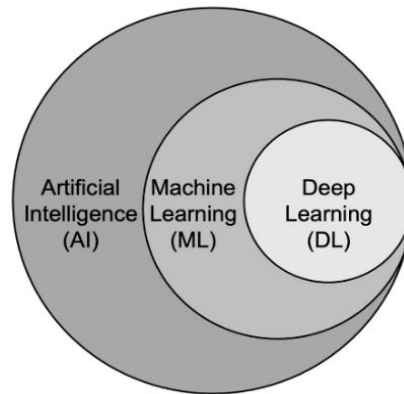
Lalu, penelitian terakhir oleh Hidayat *et al.* (2023) yang berjudul "An Implementation of Hybrid CNN-XGBoost Method for Leukemia Detection" bertujuan untuk mendeteksi leukemia dari citra sel darah. *Dataset* yang digunakan terdiri dari 12.528 citra sel darah, yang dibagi menjadi data latih

(70%), data validasi (15%), dan data uji (15%). Hasil penelitian menunjukkan bahwa integrasi CNN-XGBoost meningkatkan akurasi deteksi leukemia dari 85.32% (CNN murni) menjadi 85.43%. Berdasarkan penelitian yang telah dipaparkan sebelumnya, dengan penggunaan metode dan data yang berbeda-beda dapat disimpulkan bahwa penerapan model *hybrid* CNN dan XGBoost mampu meningkatkan akurasi dalam klasifikasi citra medis. Oleh karena itu, penelitian ini difokuskan pada implementasi model *hybrid* CNN-XGBoost dalam klasifikasi penyakit kaki gajah (*elephantiasis*), serta melakukan perbandingan tingkat akurasi dengan metode CNN murni.

2.2 *Artificial Intelligence (AI)*

Artificial intelligence atau kecerdasan buatan merupakan cabang dari ilmu komputer yang berfokus pada pengembangan sistem cerdas yang dapat meniru kemampuan manusia (Solehudin & Al-Nur, 2025). Kemampuan ini mencakup berpikir, belajar, memahami bahasa, menyelesaikan masalah, serta membuat keputusan. Dalam perkembangannya, *artificial intelligence* dirancang supaya sistem tidak hanya mengikut perintah yang statis, melainkan juga mampu beradaptasi dengan situasi yang berubah (Lisaldy *et al.*, 2024). Kemampuan tersebut membuat *artificial intelligence* mampu memberikan solusi inovatif dalam berbagai tantangan yang kompleks dan dinamis (Rozali *et al.*, 2024).

Secara umum, perkembangan dari *artificial intelligence* mencakup berbagai pendekatan untuk menciptakan sistem yang cerdas. Pendekatan utama pada *artificial intelligence* sebagian besar memanfaatkan *Machine learning* (ML) dan *Deep Learning* (DL), sebagaimana ditunjukkan pada Gambar 2. Kedua pendekatan tersebut memungkinkan sistem untuk memperoleh pengetahuan atau informasi dari data dan secara otomatis mengidentifikasi pola yang rumit. Oleh karena itu, teknologi ini memiliki potensi yang luas untuk meningkatkan efisiensi dan inovasi di beragam bidang kehidupan.



Gambar 2. Hubungan AI, ML, dan DL (Ong *et al.*, 2024)

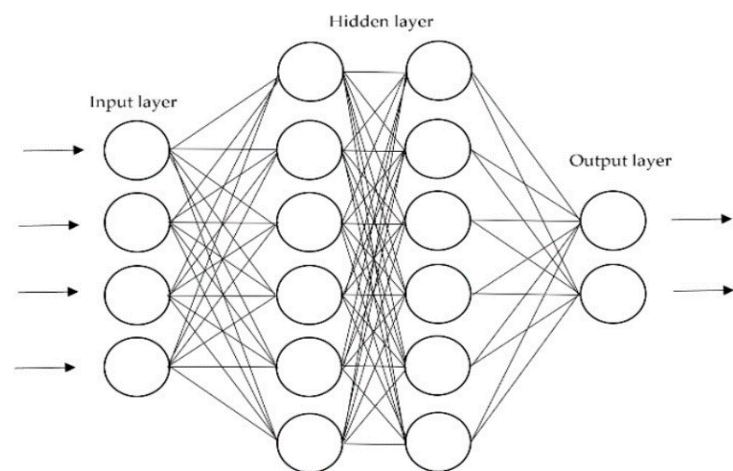
2.3 *Machine learning* (ML)

Machine learning atau pembelajaran mesin merupakan bagian dari kecerdasan buatan yang memungkinkan sistem belajar dari data dan membuat keputusan untuk mengatasi berbagai permasalahan (Fathurohman, 2021; Janiesch *et al.*, 2021). Proses pembelajaran dilakukan untuk mencapai kecerdasan pada sistem dengan melibatkan dua tahap, yaitu pelatihan (*training*) dan pengujian (*testing*). Melalui pendekatan ini, sistem dilatih untuk menganalisis data, mengidentifikasi pola, dan mengembangkan model prediksi yang digunakan pada berbagai sektor, mulai dari bidang kesehatan, ekonomi, pertanian, dan masih banyak lagi (Madjid & Kurniawan, 2022).

Secara fundamental, *machine learning* berfokus pada pengembangan model algoritma yang mampu beradaptasi dengan data baru, sehingga dapat membuat keputusan atau prediksi yang lebih kuat (Pasupuleti *et al.*, 2024). Konsep ini membuat *machine learning* semakin penting di era *big data*, di mana volume informasi yang besar memerlukan metode analisis yang cepat dan efisien. Terdapat beragam algoritma *machine learning* yang dikembangkan untuk menjawab semua kebutuhan pada berbagai bidang, mulai dari tugas klasifikasi, regresi, klustering, maupun analisis prediktif (A. Kapoor, 2024). Oleh karena itu, *machine learning* termasuk salah satu alat bantu dalam membuat prediksi dan keputusan berbasis data dengan lebih efektif (Farmin *et al.*, 2025).

2.4 Deep Learning

Deep learning atau pembelajaran mendalam merupakan bagian dari *machine learning* yang menggunakan jaringan saraf tiruan untuk memproses data dalam jumlah besar (Raup *et al.*, 2022). Setiap lapisan dalam jaringan dirancang secara bertahap untuk mengidentifikasi karakteristik dari data, mulai dari pola dasar hingga pola yang lebih kompleks (Abdullah & Yahya, 2025). Kemampuannya dalam mengolah data tidak terstruktur seperti, gambar, audio, dan teks, menjadikan metode ini banyak digunakan dalam berbagai aplikasi (Jumrah & Pulkadang, 2025). Struktur jaringan dengan lapisan-lapisan ini yang menjadi elemen penting dari efektivitas *deep learning*, karena setiap lapisan memiliki peran khusus dalam mengolah data dan mengekstraksi fitur (Kurniadi *et al.*, 2025). Gambar 3 berikut merupakan lapisan arsitektur pada *deep learning*.



Gambar 3. Arsitektur *Deep Neural Network* (Hanifi *et al.*, 2020)

Lapisan jaringan saraf tiruan terdiri dari tiga bagian utama, yaitu *input layer*, *hidden layer*, dan *output layer*. Data awal masuk melalui *layer*, selanjutnya diproses secara bertahap melalui beberapa *hidden layer* yang terhubung dengan neuron (ditunjukkan sebagai lingkaran pada gambar) untuk mengidentifikasi pola penting. Setelah data diproses dari *hidden layer*, *output layer* akan menghasilkan prediksi atau keputusan akhir berdasarkan proses dari seluruh *layer* sebelumnya. *Neuron* pada setiap lapisan tersebut saling

terhubung mulai dari *input layer* hingga *output layer* (Simatupang, 2023). Proses ini memungkinkan model dapat menganalisis pola dari data yang kompleks seperti gambar, audio, dan teks (Nurhakiki *et al.*, 2024).

2.5 *Computer Vision*

Computer vision merupakan teknologi yang memiliki kemampuan untuk memungkinkan komputer dapat melihat, mengenali, dan menganalisis objek layaknya manusia (Setiawan *et al.*, 2021). Teknologi ini memanfaatkan *artificial intelligence* untuk menghasilkan informasi penting dari *input* data visual citra dan video (Sutisna *et al.*, 2024). Pada data citra dan video, berbagai objek, pola, maupun pergerakan dapat dianalisis untuk mendapatkan informasi yang relevan. Pendekatan ini membuat *computer vision* secara otomatis dapat mengekstraksi fitur penting dari data visual untuk mendukung analisis atau pengambilan keputusan (Liu *et al.*, 2021).

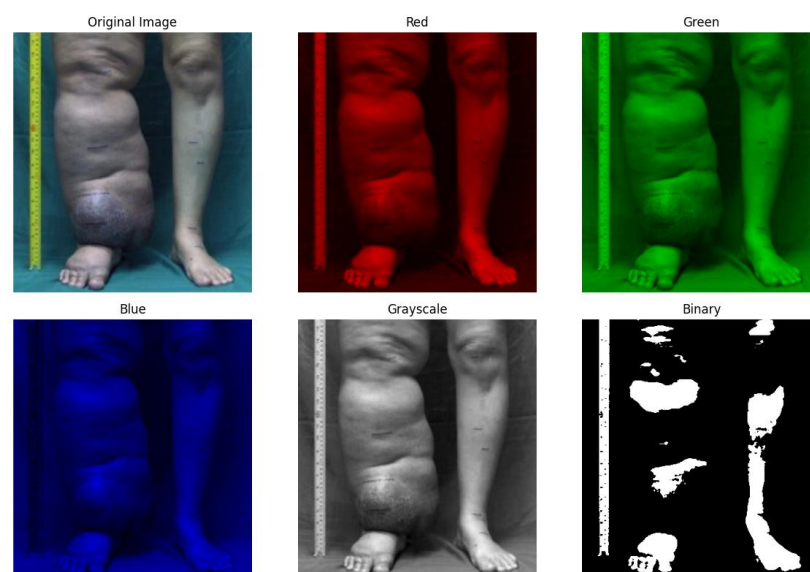
Berbagai sektor telah menerapkan *computer vision* untuk mengatasi beragam masalah. Misalnya, penelitian pada sektor kesehatan yang dilakukan oleh Supiyandi *et al.* (2024) dengan judul "Penggunaan *Computer Vision* untuk Diagnosis Pneumonia Pediatri", penelitian tersebut menunjukkan bahwa *computer vision* memiliki potensi besar dalam meningkatkan akurasi dan efisiensi diagnosis penyakit pneumonia. Teknologi ini mendukung analisis citra medis secara otomatis dan mendeteksi pola-pola yang sulit terlihat oleh mata manusia. Kemajuan ini membuktikan bahwa *computer vision* memiliki peran penting dalam meningkatkan efisiensi dan ketepatan di bidang kesehatan, terutama dalam proses deteksi dan diagnosis (Kabir *et al.*, 2025).

2.6 **Pengolahan Citra Digital**

Citra digital merupakan representasi visual dari dunia nyata yang dikonversi ke dalam bentuk digital sehingga dapat diproses dan dipahami oleh komputer. Citra ini tersusun oleh elemen-elemen terkecil berupa piksel yang tersusun

dalam baris dan kolom. Setiap piksel yang ada memiliki nilai numerik tertentu yang merepresentasikan tingkat kecerahan atau warna pada lokasi spesifik dalam citra. Untuk memaksimalkan pemanfaatannya, digunakan pengolahan citra digital yang bertujuan untuk memperbaiki kualitas citra agar mudah dimengerti oleh manusia atau diproses oleh sistem komputer (Jumadi & Sartika, 2021).

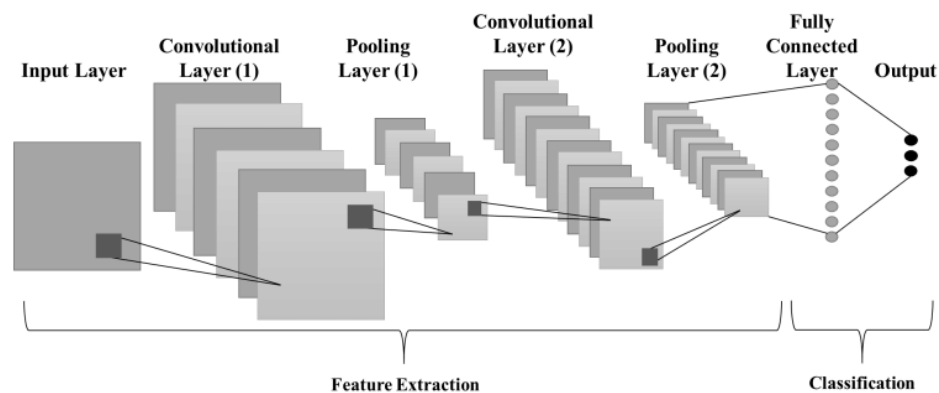
1. Dalam pengolahan citra digital, terdapat tiga jenis citra yang umum digunakan (Ashari *et al.*, 2019) seperti pada Gambar 4, dengan sumber citra kaki limfedema dari Chanwimalueang *et al.* (2015). Ketiga jenis citra tersebut terdiri dari:
2. Citra berwarna (RGB), jenis citra yang menggunakan tiga elemen warna yaitu merah (*Red*), hijau (*Green*), dan biru (*Blue*), dengan setiap warna memiliki rentang nilai dari 0 hingga 255.
3. Citra berskala keabuan (*grayscale*), jenis citra yang menampilkan gradasi dari hitam dan putih hingga menghasilkan warna abu, dengan rentang nilai 0 mewakili hitam pekat dan 255 mewakili putih bersih.
4. Citra biner (*binary*), jenis citra yang hanya memiliki dua nilai piksel, yaitu 0 mewakili hitam dan 1 mewakili putih.



Gambar 4. Representasi Warna Citra Digital. Sumber Citra: Chanwimalueang *et al.*, CC BY. Gambar dipotong dan dimodifikasi warna dari gambar asli.

2.7 Convolutional Neural Network (CNN)

Convolutional Neural Network (CNN) merupakan salah satu algoritma dalam *deep learning* yang dirancang untuk mengenali dan mengidentifikasi pola pada suatu objek (Anhar & Putra, 2023). Pendekatan ini meniru cara kerja sistem pengenalan citra pada konteks visual, sehingga dapat memahami dan mengolah informasi visual dengan efektif. CNN telah terbukti memberikan hasil yang signifikan dalam berbagai tugas pengenalan citra, mulai dari proses klasifikasi, prediksi, dan deteksi. Selain itu, CNN bekerja menggunakan lapisan *convolutional* dan *pooling* yang dilakukan bertahap untuk mengekstraksi fitur (*feature maps*) dari data secara otomatis (Kurniadi *et al.*, 2025). Arsitektur umum dari CNN terdiri dari beberapa lapisan yang terlihat seperti pada Gambar 5.



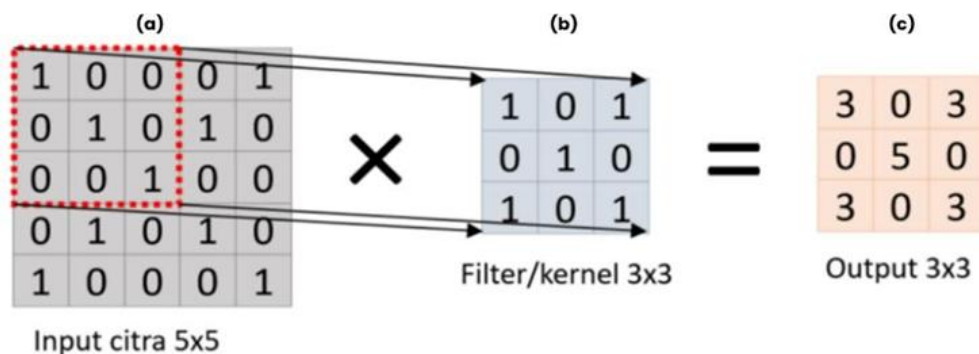
Gambar 5. Arsitektur Algoritma *Convolutional Neural Network* (CNN) (Samat *et al.*, 2020)

Arsitektur CNN umumnya terdiri dari tiga jenis lapisan, yaitu *convolutional layer*, *pooling layer*, dan *fully connected*. Lapisan-lapisan ini terbagi menjadi dua bagian, yaitu *feature extraction* dan *classification*. Pada bagian *feature extraction*, terdapat dua lapisan yang digunakan untuk menghasilkan fitur (*feature maps*), yaitu *convolutional layer* dan *pooling layer*. *Feature maps* adalah hasil dari penerapan filter konvolusi atau yang disebut sebagai *kernel*. Hasil ini berupa angka-angka yang merepresentasikan pola dari sebuah citra yang selanjutnya diproses pada bagian *classification* (Azmi *et al.*, 2023). Pada bagian *classification*, subsampling *feature maps* terakhir akan melalui

proses *flatten* menjadi vektor satu dimensi sebelum diteruskan ke *fully connected layer*, lalu menghasilkan akurasi klasifikasi untuk setiap kelas.

a. *Convolutional Layer* (Lapisan Konvolusi)

Convolutional layer adalah lapisan utama yang bertujuan untuk melakukan ekstraksi fitur dari data *input*, seperti citra (Ibrahim *et al.*, 2022). Lapisan ini melakukan proses operasi *convolution* menggunakan filter kernel. Setiap kernel digeser ke seluruh area *input* dengan operasi *convolution*, sehingga menghasilkan *feature maps* mendeteksi fitur tertentu, seperti garis, tepi, tekstur, atau pola (Pakiding *et al.*, 2025). Pada lapisan ini, fungsi aktivasi ReLU diterapkan untuk mengubah nilai negatif menjadi nol, sehingga model dapat mempelajari pola yang kompleks dan abstrak. Selain itu, proses ini juga menerapkan *stride* untuk mengatur seberapa jauh filter kernel bergeser pada setiap langkah dan *padding* menambahkan nol di sekeliling *input* citra untuk menjaga ukuran output dan mencegah hilangnya informasi pada citra untuk mengatur pergeseran kernel sekaligus mempertahankan ukuran output. Berikut merupakan proses dari *convolutional layer* yang ditunjukkan seperti pada Gambar 6:



Gambar 6. Proses *Convolutional Layer*: (a) Input Matriks, (b) *Kernel*, (c) *Feature Map* (Handoko *et al.*, 2024)

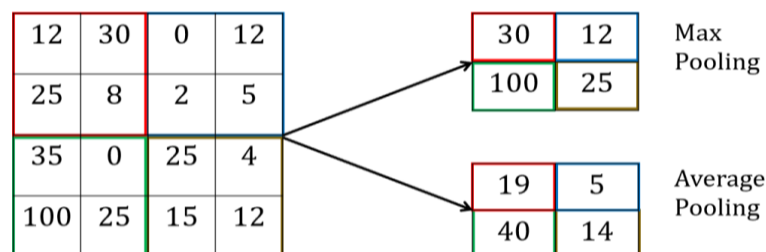
Perhitungan operasi *convolution* pada Gambar 6 di atas, dimulai dengan mengambil sub-matriks dari citra berukuran 5×5 yang seukuran dengan filter *kernel* 3×3 . Setiap elemen dari sub-matriks tersebut dikalikan dengan elemen *kernel*, sesuai dengan persamaan (1) berikut:

$$g(x, y) = h(x, y) \cdot f(x, y) \quad (1)$$

Operasi *convolution* $g(x, y)$ sebagaimana didefinisikan pada persamaan (1) merupakan produk dot dari dua fungsi. Di mana, $h(x, y)$ merupakan matriks *input* dan $f(x, y)$ merupakan matriks *kernel*. kemudian hasil perkalian tersebut dijumlahkan untuk memperoleh satu nilai pada *feature map*. Proses ini digeser ke kanan satu *pixel* perlangkah (*stride* = 1), menghasilkan *feature map* berukuran 3×3 .

b. *Pooling Layer* (Lapisan *Pooling*)

Pooling layer adalah lapisan yang berfungsi untuk mengurangi ukuran matriks dari *feature map*, namun tetap mempertahankan informasi penting yang telah diekstrak pada *convolution layer* (Dani & Handayani, 2024). Konsep dasar dari proses *pooling* adalah melakukan *down-sampling*, yaitu mengurangi kompleksitas perhitungan pada *layer* berikutnya dan mengurangi resolusi dari citra. Proses ini menggunakan *kernel* dengan ukuran tertentu yang digeser ke kanan pada area *feature map*. Hasil dari lapisan *pooling* akan menghasilkan matriks baru dengan ukuran yang lebih kecil, yang selanjutnya menjadi representasi fitur penting untuk tahap ekstraksi lanjutan atau proses klasifikasi (Qadriah & Cut, 2024).



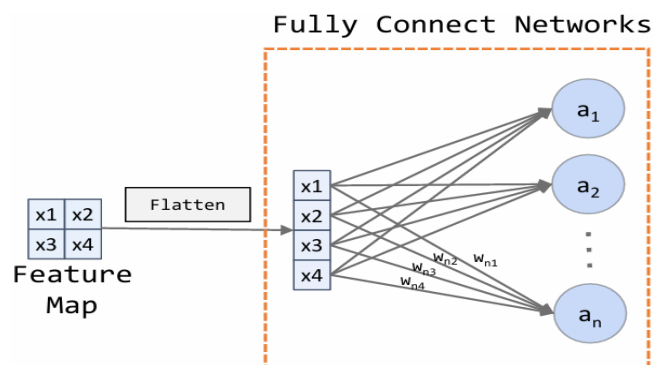
Gambar 7. Operasi *Max* dan *Average* pada *Pooling Layer* (Alsaleh & Sainte, 2021)

Terdapat dua jenis operasi *pooling layer* yang ditunjukkan pada Gambar 7, yaitu *max pooling* dan *average pooling* (Briliantio *et al.*, 2020). Kedua jenis operasi dimulai dengan menerapkan *stride* = 2, yaitu matriks *kernel*

bergeser sejauh dua piksel dari ukuran matriks aslinya, sehingga membagi matriks *input* berukuran 4×4 menjadi berukuran 2×2 . Pada operasi *max pooling*, nilai tertinggi dari tiap submatriks *input* 2×2 dipilih, sehingga menghasilkan *output* dengan nilai 30, 12, 100, dan 25. Berbeda dengan operasi *average pooling* yang menghitung rata-rata setiap submatriks *input* 2×2 , dengan cara menjumlahkan seluruh nilai lalu membaginya dengan banyak nilai *input*, sehingga menghasilkan nilai output 19, 5, 40, dan 14. Proses ini efektif mengurangi ukuran *feature map* secara signifikan, namun tetap mempertahankan informasi penting untuk langkah pengolahan berikutnya.

c. **Fully Connected Layer (Lapisan Terhubung Penuh)**

Fully connected layer merupakan lapisan terakhir pada arsitektur CNN yang berfungsi untuk mengklasifikasikan citra berdasarkan informasi yang diperoleh dari lapisan-lapisan sebelumnya (Fuadah *et al.*, 2022). Setelah melewati lapisan *convolution* dan *pooling*, *feature map* yang dihasilkan masih berbentuk matriks, sehingga harus melalui proses *flatten* terlebih dahulu. *Flatten* adalah sebuah lapisan yang mengubah fitur (*feature map*) menjadi vektor dengan satu dimensi sebagai *input* untuk lapisan *fully connected layer*. Vektor ini kemudian dilanjutkan ke *fully connected layer*, di mana setiap neuron saling terhubung dengan seluruh *input* untuk proses klasifikasi (Nurdin *et al.*, 2020). Proses ini dapat dilihat seperti pada Gambar 8 berikut.



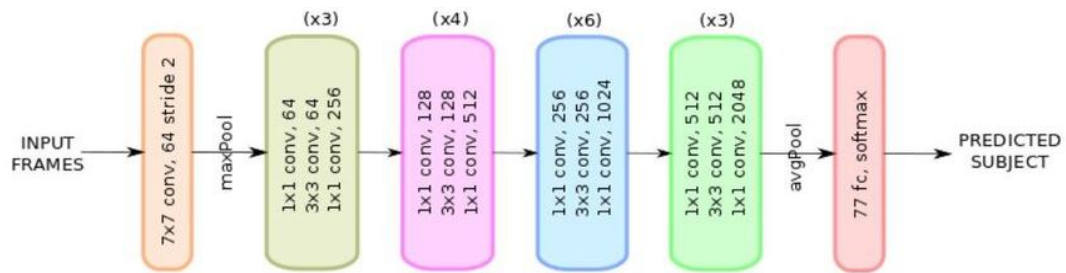
Gambar 8. *Flatten* dan *Fully Connected Layer* (R. Wang *et al.*, 2024)

Pada proses pelatihan, bobot dan bias pada *fully connected* disesuaikan dengan algoritma *backpropagation*, dengan tujuan mengurangi kesalahan prediksi (Nurhadi & Hendrik, 2024). Proses ini dilakukan berulang-ulang supaya hasil prediksi menjadi akurat. Umumnya, lapisan akhir menggunakan fungsi aktivasi seperti *softmax* untuk klasifikasi *multi-class* atau sigmoid untuk klasifikasi biner. Karena memiliki banyak parameter, lapisan ini sering digabungkan dengan teknik regulasi seperti *dropout* agar model tidak mengalami *overfitting* dan dapat melakukan generalisasi dengan efektif pada data baru (Salehin & Kang, 2023).

2.8 Residual Network (ResNet)

Residual Network atau ResNet merupakan jenis arsitektur dari *Convolution Neural Network* (CNN) yang banyak dimanfaatkan sebagai *pretrained model* atau model yang telah dilatih sebelumnya (Ridhovan & Suharso, 2022). ResNet memiliki kemampuan untuk mengembangkan arsitektur dengan lebih banyak lapisan tanpa menambah kompleksitas secara signifikan, sehingga pelatihan tetap berjalan dengan efisien dan stabil tanpa mengalami penurunan akurasi saat proses pelatihan. Selain itu, ResNet memiliki beberapa variasi arsitektur berdasarkan banyaknya lapisan, seperti ResNet18, ResNet34, ResNet50, ResNet101, dan ResNet152, di mana masing-masing jumlah lapisan pada arsitektur ini mampu meningkatkan kompleksitas kemampuan dalam merepresentasikan fitur dan mengklasifikasi citra dengan tepat (Ramadhan *et al.*, 2024).

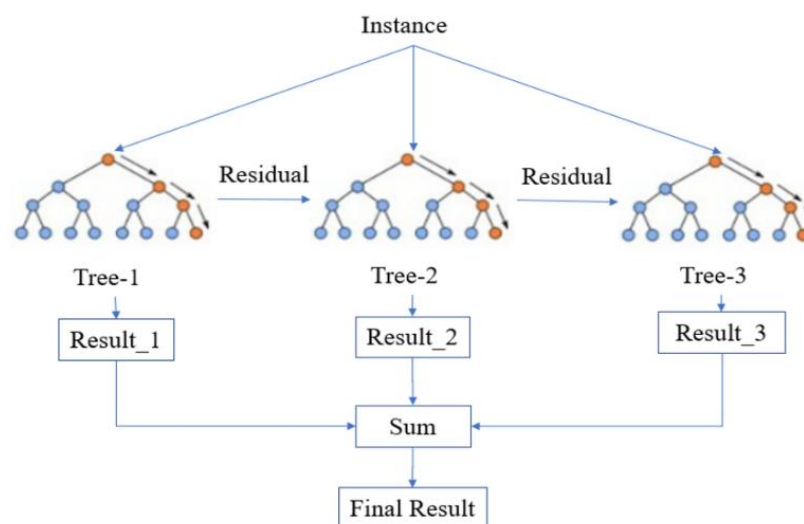
Penelitian ini menggunakan arsitektur ResNet50 yang terdiri dari lima puluh lapisan yang menggunakan susunan *bottleneck block* dengan tiga lapisan *convolution* (1x1, 3x3, 1x1). Arsitektur ini terbagi menjadi menjadi empat tahap utama yang masing-masing memiliki 3, 4, 6, dan 3 blok residual. Setiap blok bertujuan untuk memperdalam representasi fitur tanpa menambah beban komputasi secara signifikan. Arsitektur dari ResNet50 dapat dilihat seperti pada Gambar 9:



Gambar 9. Arsitektur ResNet50 (X. Wang & Li, 2023)

2.9 Extreme Gradient Boosting (XGBoost)

XGBoost (*Extreme Gradient Boosting*) merupakan algoritma *machine learning* yang menggunakan pendekatan *ensemble learning*, dengan cara menggabungkan pohon keputusan (*decision tree*) secara bertahap untuk membangun model prediksi yang kuat (Muradiansyah, 2024). XGBoost bekerja dengan cara membangun pohon secara bertahap, di mana setiap pohon bertujuan untuk memperbaiki residual atau kesalahan prediksi dari pohon-pohon yang mendahuluinya. Dengan menggunakan metode ini, memungkinkan pengembangan model klasifikasi berbasis *ensemble* dan pohon keputusan dapat meningkatkan efektivitas klasifikasi serta mengurangi jumlah fitur, sehingga waktu yang diperlukan untuk memproses data menjadi lebih singkat (Manju *et al.*, 2019).



Gambar 10. Cara Kerja Algoritma XGBoost (W. Wang *et al.*, 2021)

Pada Gambar 10 menunjukkan alur dari cara kerja algoritma XGBoost, di mana pada data awal digunakan untuk melatih *Tree* 1, kemudian sisa residual atau kesalahan dari *Tree* 1 digunakan untuk melatih *Tree* 2. Proses ini berlanjut hingga *Tree* ke- n , sehingga setiap pohon memperbaiki kesalahan prediksi dari pohon sebelumnya. Semua *output* dari pohon tersebut kemudian dijumlahkan menjadi hasil prediksi akhir dari XGBoost. Secara matematis, hasil prediksi akhir dapat direpresentasikan sebagai berikut:

$$\hat{y}_i = \sum_{k=1}^n f_k(x_i) \quad (2)$$

Pada persamaan (2), f_k merujuk pada pohon keputusan ke- k dengan n menyatakan jumlah total pohon keputusan, sehingga $f_k(x_i)$ mewakili hasil prediksi dari pohon k terhadap data ke- i dan \hat{y}_i menunjukkan nilai *output* yang diprediksi oleh model. Proses pembelajaran fungsi f_k pada pohon dilakukan dengan cara meminimalkan fungsi objektif yang terdiri dari fungsi loss dan regularisasi kompleksitas model. Fungsi objektif ini dapat dinyatakan sebagai berikut:

$$L^t = \sum_{i=1}^n \left[l(y_i, \hat{y}_i^{(t-1)} + f_t(x_i)) \right] + \Omega(f_t) \quad (3)$$

Pada persamaan (3) tersebut, l merupakan fungsi *loss* yang berfungsi untuk menilai perbedaan antara nilai yang diprediksi pada iterasi sebelumnya $\hat{y}_i^{(t-1)}$ dengan nilai aktual (y_i). Sementara itu, fungsi regularisasi ($\Omega(f_t)$) berperan untuk mengatur tingkat kompleksitas model pohon agar dapat mengurangi kemungkinan terjadinya *overfitting*. Fungsi regularisasi dapat dinyatakan pada Persamaan (4) berikut:

$$\Omega(f_t) = \gamma T_t + \frac{1}{2} \lambda \sum_{j=1}^T w_j^2 \quad (4)$$

XGBoost menggunakan pendekatan ekspansi Taylor orde kedua untuk mendekati fungsi *loss* seperti yang direpresentasikan pada persamaan ke (5). Pendekatan ini menerapkan dua komponen utama, yaitu gradien orde

pertama (*gradien*) dan gradien orde kedua (*hessian*) dari fungsi *loss*. Pada persamaan tersebut, komponen $g_i f_t(x_i)$ dan $\frac{1}{2} h_i f_t^2(x_i)$ merupakan hasil aproksimasi Taylor orde kedua terhadap fungsi *loss*, di mana g_i menyatakan gradien, h_i menyatakan hessian, dan $f_t(x_i)$ merupakan prediksi dari pohon keputusan pada iterasi ke- t terhadap data ke- i .

$$L^t \cong \sum_{i=1}^n \left[l \left(y_i, \hat{y}_i^{(t-1)} + g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i) \right) \right] + \Omega(f_t) \quad (5)$$

Nilai *gradien* (g_i) dan *hessian* (h_i) diperoleh dengan menggunakan turunan pertama dan kedua dari fungsi *loss* terhadap hasil prediksi pada iterasi sebelumnya. Pada persamaan (6), *gradien* menunjukkan besar dan arah kesalahan prediksi terhadap nilai prediksi sebelumnya, sedangkan *hessian* menunjukkan tingkat perubahan gradien. Keduanya dapat dihitung melalui persamaan sebagai berikut:

$$g_i = \frac{\partial L(y_i, F_{k-1}(x_i))}{\partial F_{k-1}(x_i)} = p_i - y_i \quad (6)$$

$$h_i = \frac{\partial^2 L(y_i, F_{k-1}(x_i))}{\partial F_{k-1}(x_i)} = p_i (1 - p_i) \quad (7)$$

Selanjutnya, XGBoost mengukur kualitas pemisahan (*split*) dengan menggunakan metrik *gain*, yang menunjukkan seberapa besar penurunan kesalahan (*loss*) atau peningkatan kualitas model yang diperoleh suatu fitur setelah node dibagi. Secara matematis, I_L dan I_R adalah himpunan data pada node kiri ($G_L = \sum_{i \in I_L} g_i$, $H_L = \sum_{i \in I_L} h_i$) dan kanan ($G_R = \sum_{i \in I_R} g_i$, $H_R = \sum_{i \in I_R} h_i$) setelah dilakukannya pemisahan. Berdasarkan hal tersebut, nilai *gain* dapat dihitung menggunakan persamaan (8) sebagai berikut:

$$Gain = \frac{1}{2} \left[\frac{G_L^2}{H_L + \lambda} + \frac{G_R^2}{H_R + \lambda} - \frac{(G_L + G_R)^2}{H_L + H_R + \lambda} \right] - \gamma \quad (8)$$

Pada persamaan tersebut, G_L dan G_R merupakan masing-masing total *gradien* di sisi kiri dan kanan, serta H_L dan H_R adalah total *hessian* pada sisi kiri dan kanan. Nilai $(G_L + G_R)$ dan $(H_L + H_R)$ merupakan akumulasi *gradien* dan *hessian* dari seluruh data sebelum proses *split* dilakukan.

Setiap daun pohon XGBoost memiliki bobot (*leaf weight*) yang menentukan kontribusi daun terhadap hasil prediksi. Untuk menghitung nilai bobot, dihitung menggunakan persamaan (6):

$$\omega_j = -\frac{\sum_{i \in I_j} g_i}{\sum_{i \in I_j} h_i + \lambda} = -\frac{G_j}{H_j + \lambda} \quad (6)$$

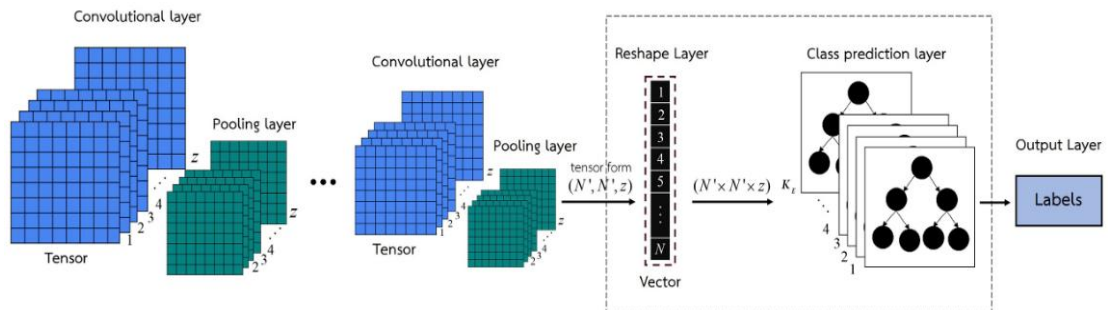
Dengan, G_j dan H_j merupakan total *gradien* dan *hessian* pada daun ke- j . Nilai bobot ini, kemudian digunakan untuk memperbaiki hasil prediksi pada iterasi berikut. Dengan berbagai kombinasi perhitungan, XGBoost mampu memberikan hasil prediksi yang mendekati nilai sebenarnya atau aktual (Mubarok & Septian, 2025).

2.10 *Hybrid Convolutional Neural Network dan Extreme Gradient Boosting (CNN-XGBoost)*

Hybrid CNN-XGBoost merupakan sebuah pendekatan yang menggabungkan kemampuan dari *Convolutional Neural Network* (CNN) dalam mengambil fitur citra dengan keunggulan dari *Extreme Gradient Boosting* (XGBoost) sebagai algoritma klasifikasi (Jiao *et al.*, 2021). Dalam hal ini, CNN berfungsi sebagai pengambil fitur yang secara otomatis mengenali pola visual penting dalam gambar, seperti tekstur, tepi, dan warna. Melalui serangkaian lapisan *convolutional* dan *pooling*, citra ditransformasikan menjadi representasi numerik yang memiliki dimensi tinggi. Selain itu, model dapat membuat prediksi dan klasifikasi lebih akurat, karena dapat menangkap fitur yang lebih relevan dan informatif (Nagy & Kapusta, 2023).

Selanjutnya, vektor fitur yang dihasilkan dari CNN tidak langsung diproses oleh lapisan *fully connected*, melainkan digunakan sebagai *input* bagi XGBoost. Algoritma ini bekerja dengan membangun pohon keputusan secara bertahap untuk meningkatkan akurasi prediksi. Dengan menerapkan XGBoost, model menjadi lebih tangguh, serta mampu menangani data yang

tidak seimbang. Oleh karena itu, melalui metode kombinasi atau *hybrid* ini, fitur yang diekstraksi oleh CNN mampu meningkatkan kinerja model XGBoost (Putra *et al.*, 2024). Berikut merupakan arsitektur dari *hybrid* algoritma CNN-XGBoost seperti yang ditunjukkan pada Gambar 11:



Gambar 11. Arsitektur CNN-XGBoost (Thongsuwan *et al.*, 2019)

2.11 Penyakit Kaki Gajah (*Elephantiasis*)

Dalam penelitian ini, menyoroti penyakit kaki gajah (*elephantiasis*) dengan fokus pada dua penyebab utama, yaitu:

2.11.1 Filariasis Limfatik (Parasit)

Filariasis limfatik adalah penyakit parasit menular akibat infeksi cacing filaria, yaitu *Wuchereria bancrofti*, *Brugia malayi*, dan *Brugia timori* yang ditularkan melalui gigitan nyamuk. Nyamuk tersebut biasanya dari genus *Aedes*, *Anopheles*, *Armigeres*, *Culex*, dan *Mansonia* (Kermelita *et al.*, 2023). Penularan terjadi ketika nyamuk yang mengandung larva cacing filaria menggigit orang sehat, kemudian masuk ke dalam pembuluh darah. Infeksi tersebut menyerang sistem limfatik dan menyebabkan gangguan berupa penyumbatan cairan getah bening (Rajamanickam & Babu, 2025). Jika tidak ada penanganan tepat, penyakit filariasis menjadi masalah jangka panjang dan bersifat menahun dengan gejala pembengkakan ekstrem (*elephantiasis*) pada tungkai atau kaki. Selain pada kaki yang merupakan lokasi pembengkakan paling

umum, pembengkakan ini juga dapat terjadi pada lengan, payudara, serta area genital yang dapat menimbulkan kecacatan permanen dan berdampak besar pada kualitas hidup bagi penderitanya.

Hingga saat ini, filariasis limfatik masih menjadi masalah kesehatan masyarakat di negara-negara tropis, termasuk Indonesia. Penyakit ini termasuk sebagai penyakit tropis yang sering terabaikan (*Neglected Tropical Disease/NTD*), karena sering terjadi pada wilayah terpencil dengan kondisi perekonomian yang lemah (Ardanari *et al.*, 2020). Penyakit ini tidak hanya memberikan dampak pada kesehatan saja, tetapi juga memengaruhi sisi sosial dan ekonomi karena mengalami pembatasan dalam aktivitas harian. Selain itu, kurangnya sosialisasi dari tenaga kesehatan dan rendahnya tingkat kesadaran masyarakat juga memperburuk situasi ini (Dheo *et al.*, 2019; Paide *et al.*, 2023). Oleh karena itu, untuk mengurangi peningkatan penderita filariasis dibutuhkan pendekatan menyeluruh seperti pemberian obat pencegahan secara masal, perawatan, dan edukasi masyarakat untuk menghentikan siklus penularan penyakit filariasis (Yuziani *et al.*, 2021).

2.11.2 Limfedema (Non-Parasit)

Limfedema merupakan pembengkakan kronis yang terjadi ketika terdapat masalah pada sistem limfatik karena faktor non-parasit, seperti operasi, terapi radiasi (radioterapi), infeksi, atau kelainan bawaan sejak lahir. Limfedema dibagi menjadi dua jenis, yaitu primer dan sekunder. Limfedema primer terjadi karena faktor internal seperti kelainan limfatik bawaan sejak lahir dan kasusnya jarang terjadi (Paudel *et al.*, 2025). Sementara limfedema sekunder, lebih sering terjadi karena faktor eksternal seperti, operasi kanker, kemoterapi terapi radiasi, infeksi, trauma, dan penyakit vena (Garza *et al.*, 2017). Cacing filaria sebagai penyebab limfatik filariasis

termasuk ke dalam limfedema sekunder, karena infeksi yang ditimbulkan selalu berujung pada terjadinya limfedema (Marchica *et al.*, 2021). Namun, tidak semua kasus limfedema disebabkan oleh cacing filaria, karena ada faktor non-parasit lain juga yang dapat menyebabkan limfedema (Varghese, 2021).

Penderita limfedema diperkirakan 140 hingga 250 juta orang di seluruh dunia, baik limfedema primer maupun sekunder (Duhon *et al.*, 2022). Jumlah tersebut, menunjukkan masalah serius, terutama pada stadium 3 (0-3) atau yang dikenal sebagai *lymphostatic elephantiasis*. Limfedema pada stadium 3 atau tahap yang paling parah, ditandai dengan pembengkakan besar, pengerasan kulit, dan perubahan kulit yang menyerupai kaki gajah (McGinity *et al.*, 2024). Namun, terdapat pendekatan untuk memperlambat perkembangan penyakit, mencegah komplikasi, dan mengurangi pembengkakan melalui terapi kompresi, fisioterapi, perawatan kulit, atau tindakan pembedahan tertentu (de Sire *et al.*, 2021).

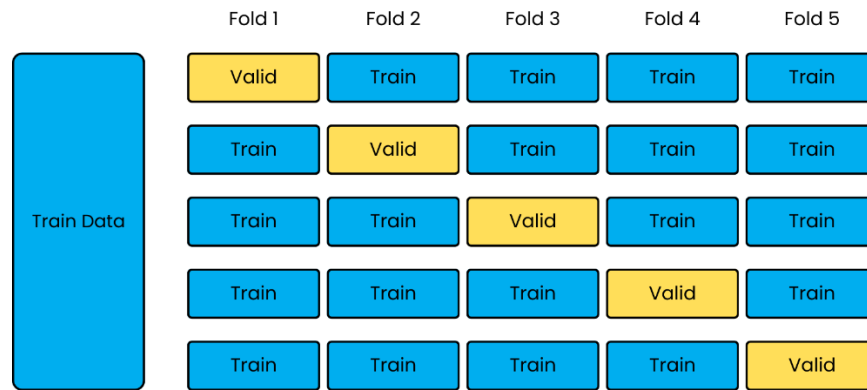
2.12 *Bayesian Optimization (Optuna)*

Bayesian optimization adalah teknik optimasi berbasis probabilitas yang digunakan untuk meningkatkan kinerja model melalui *hyperparameter tuning* atau *hypertuning*. *Hypertuning* merupakan proses mencari nilai optimal yang dapat memengaruhi performa model. *Bayesian optimization* ini bekerja dengan membuat model prediksi yang dapat memperkirakan kinerja model berdasarkan kombinasi *hyperparameter* yang telah diuji sebelumnya, sehingga pemilihan kombinasi berikutnya menjadi lebih efisien dibandingkan mencoba semua kemungkinan kombinasi satu per satu hingga mendapatkan *hyperparameter* terbaik. Selain itu, dengan penerapan metode ini dapat mempersingkat waktu komputasi, karena proses pencarian difokuskan pada parameter yang paling optimal.

Proses ini dilakukan menggunakan Optuna, yaitu *library* Python yang mempermudah pencarian *hyperparameter* melalui metode *bayesian optimization* dengan pendekatan algoritma *Tree-Structured Parzen Estimator* (TPE) (Watanabe, 2025). Optuna memiliki mekanisme *prunning*, yang memungkinkan penghentian percobaan *hyperparameter* lebih awal apabila tidak menunjukkan hasil yang ideal. Berbagai penelitian menunjukkan bahwa Optuna cocok digunakan pada algoritma XGBoost dan CNN, di mana algoritma tersebut memiliki kombinasi *hyperparameter* yang kompleks. Dengan demikian, proses pencarian *hypertuning* tidak hanya menghemat waktu dan sumberdaya, melainkan dapat meningkatkan peluang dalam memberikan performa model terbaik.

2.13 *Cross Validation*

Cross validation adalah metode yang digunakan untuk mengevaluasi performa dan kemampuan generalisasi suatu model terhadap data yang belum pernah dilihat sebelumnya. Salah satu pendekatan yang paling umum digunakan adalah *k-fold cross-validation*, di mana data latih (*train*) dibagi menjadi k bagian (*fold*) dengan ukuran yang relatif sama. Pada setiap iterasi, satu *fold* digunakan sebagai data validasi (*test fold*), sementara sisa *fold* ($k-1$) digunakan sebagai data latih. Proses ini dilakukan secara bergantian dan berulang hingga seluruh *fold* telah dijadikan data validasi satu kali, lalu kinerja model dari semua iterasi dirata-ratakan untuk mendapatkan kinerja yang lebih konsisten dan tidak bergantung pada satu pembagian data saja. Metode ini digunakan ketika jumlah *dataset* relatif kecil, karena terdapat proses pembagian data dan pengujian secara optimal. Gambar 12 berikut merupakan cara kerja dari teknik *cross validation*.



Gambar 12. Cara Kerja *Cross Validation*

Selain *k-fold cross validation*, terdapat berbagai variasi metode *cross-validation*. Salah satunya adalah *stratified k-fold*, yang membagi data menjadi k bagian (*fold*) secara bergantian seperti *k-fold* biasa, tetapi dengan pendekatan tertentu agar proporsi kelas pada setiap *fold* tetap seimbang dengan proporsi kelas asli pada *dataset*. Selain itu, terdapat *leave-one-out cross-validation* (LOOCV) dan *leave-p-out cross-validation* (LPOCV) yang juga dapat digunakan untuk melakukan pembagian data pelatihan menjadi beberapa subset dan validasi pada proses evaluasi. Pemilihan metode *cross-validation* yang tepat berperan penting dalam mendapatkan evaluasi model yang akurat dan mengurangi potensi bias akibat pembagian data tertentu.

2.14 *Confusion Matrix*

Confusion Matrix merupakan metode evaluasi yang digunakan untuk menilai kinerja algoritma klasifikasi dalam *machine learning*. Matriks ini menampilkan perbandingan antara hasil prediksi model dengan label sebenarnya pada data uji (Tjahjadi & Santoso, 2023). Melalui *confusion matrix*, metrik evaluasi seperti *accuracy*, *precision*, *recall*, dan *f1-score* dapat dihitung berdasarkan hasil klasifikasi tiap label atau kelas. Tabel 2 berikut merupakan tabel rincian *confusion matrix*.

Tabel 2. *Confusion Matrix*

Kelas Aktual	Kelas Prediksi	
	<i>Positive</i>	<i>Negative</i>
<i>Positive</i>	TP	FN
<i>Negative</i>	FP	TN

Di mana, *True Positive* (TP) menggambarkan kondisi ketika prediksi dan nilai aktual sama-sama positif. Sebaliknya, *False Positive* (FP) terjadi saat model memberikan prediksi positif, namun nilai aktual bernilai negatif, di mana situasi ini sering disebut *Type I Error*. Sementara itu, *True Negative* (TN) menunjukkan situasi ketika nilai prediksi dan nilai aktual sama-sama negatif. Adapun *False Negative* (FN), terjadi ketika model memprediksi negatif, namun nilai aktual bernilai positif, yang disebut sebagai *Type II Error* (Sathyanarayanan, 2024).

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \times 100\% \quad (9)$$

$$Precision = \frac{TP}{TP+FP} \quad (10)$$

$$Recall = \frac{TP}{TP+FN} \quad (11)$$

$$F1 - Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (12)$$

Accuracy adalah metrik yang menunjukkan persentase prediksi model yang benar dari seluruh data yang diuji. Nilai ini didapat dari perbandingan jumlah data yang diklasifikasikan dengan benar terhadap total keseluruhan data. *Precision* digunakan untuk menilai seberapa tepat prediksi positif yang dihasilkan oleh model sesuai dengan kondisi sebenarnya. Sementara itu, *Recall* digunakan untuk menilai seberapa besar data positif yang berhasil dideteksi dengan tepat oleh model. Untuk mencapai keseimbangan antara *precision* dan *recall*, digunakan *f1-score*, yaitu ukuran yang menghitung rata-rata harmonis dari kedua nilai tersebut sehingga memberikan gambaran yang lebih seimbang mengenai kinerja model (Kurniawan *et al.*, 2024).

3.2 Alat Pendukung

3.2.1 Perangkat Keras (*Hardware*)

Perangkat keras atau hardware yang digunakan dalam penelitian ini adalah laptop yang menjadi perangkat utama untuk pengolahan data dan pengembangan model, dengan spesifikasi sebagai berikut:

- a. Perangkat: Lenovo V14 G2 ITL
- b. Prosesor: 11th Gen Intel® Core™ i3-1115G4 @ 3.00GHz
- c. RAM: 12 GB
- d. GPU: NVIDIA GeForce MX350

3.2.2 Perangkat Lunak (*Software*)

Dalam pelaksanaan penelitian ini, digunakan berbagai perangkat lunak atau *software* yang mendukung proses pengolahan data dan pengembangan model. Berikut merupakan daftar perangkat lunak yang digunakan selama penelitian berlangsung:

a. Sistem Operasi

Perangkat laptop yang digunakan pada penelitian ini dilengkapi dengan sistem operasi *Windows 11 Home Single Language 64-bit* dengan arsitektur *x64-based processor*. Sistem operasi ini menyediakan keamanan, stabilitas, dan kompatibilitas penuh dalam menjalankan berbagai perangkat lunak. Dengan demikian, perangkat ini dapat menjalankan proses analisis data hingga pengembangan dan pengujian model secara optimal.

b. Google Chrome

Google Chrome berfungsi sebagai *browser* utama yang cepat dan aman, sehingga memudahkan akses ke berbagai layanan *online*

berbasis web, termasuk Google Colab, Google Drive, serta layanan *online* lainnya. Selain itu, *browser* ini menyediakan antarmuka yang mudah digunakan dan memiliki tingkat kompatibilitas yang baik dengan berbagai situs.

c. Google Colab dan Kaggle Notebook

Google Colaboratory (Colab) dan Kaggle Notebook adalah layanan *cloud computing* milik Google, yang menyediakan *notebook* interaktif berbasis Jupyter yang dapat dijalankan langsung melalui browser tanpa instalasi lokal. Dalam penelitian ini, keduanya digunakan untuk menulis dan menjalankan kode Python dengan dukungan *runtime* GPU secara gratis, sehingga proses pelatihan model *deep learning* bisa menjadi lebih efisien. Selain itu, terdapat berbagai pustaka *machine learning* yang digunakan untuk pengolahan data seperti seperti NumPy, Pandas, Matplotlib, Scikit-learn, OpenCV, Seaborn, dan masih banyak lagi.

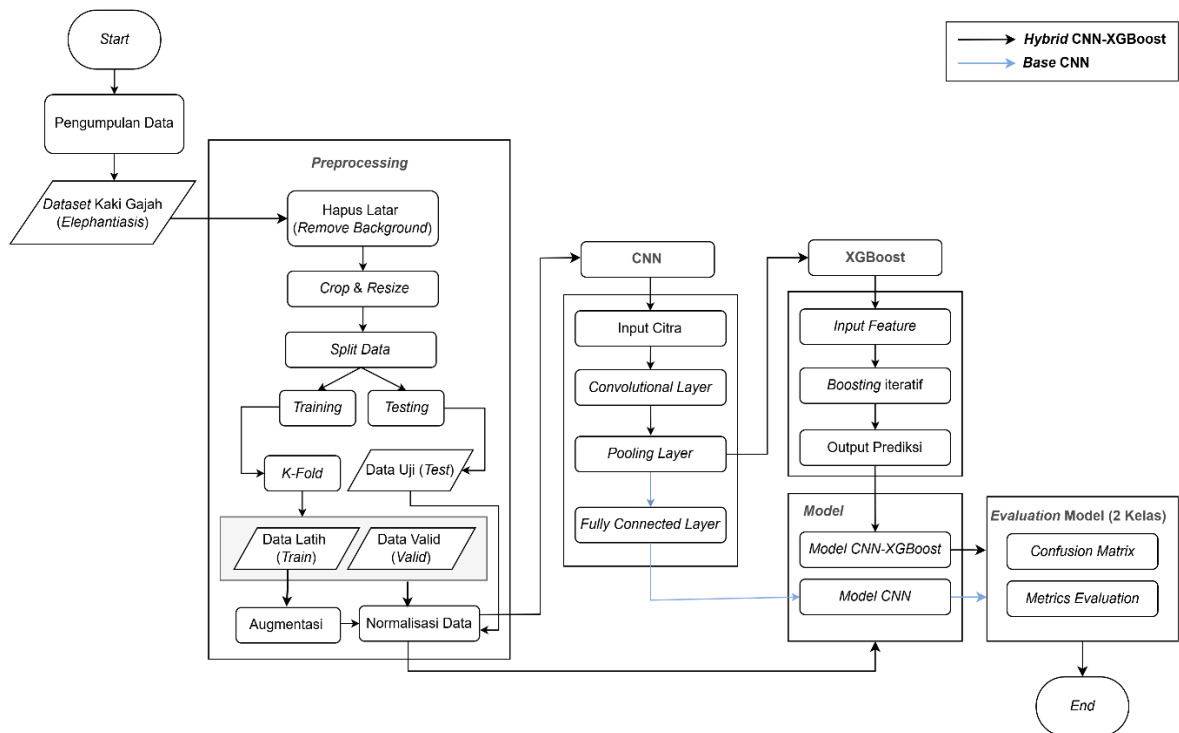
d. Google Drive

Google drive pada penelitian ini digunakan sebagai layanan penyimpanan *dataset* yang aman dan dapat diakses secara *real-time*. Fitur sinkronisasi otomatis memastikan bahwa semua perubahan data tersimpan dan dapat diakses dengan mudah. Layanan ini juga terintegrasi dengan Google Colab, sehingga mendukung pengelolaan dan akses data selama proses penelitian secara keseluruhan.

3.3 Tahapan Penelitian

Terdapat beberapa tahap yang dilakukan pada penelitian, mulai dari pengumpulan data, *pre-processing*, *training*, *testing*, hingga evaluasi model.

Tahap penelitian ini dapat dilihat seperti pada Gambar 13 berikut:



Gambar 13. Tahapan Penelitian

Flowchart tahapan penelitian ini menunjukkan dua arsitektur yang dianalisis. Panah hitam merepresentasikan alur dari model *hybrid* CNN-XGBoost, yang mencakup semua tahap mulai dari *preprocessing*, ekstraksi fitur oleh CNN, hingga klasifikasi yang dilakukan oleh XGBoost. Sebagai perbandingan, panah biru merepresentasikan model dari CNN murni yang melalui tahapan awal seperti model *hybrid* CNN-XGBoost, seperti *preprocessing* dan *training* CNN. Namun, hasil ekstraksi fiturnya langsung digunakan untuk pelatihan dan pengklasifikasian tanpa melibatkan tahapan dari XGBoost. Selanjutnya pada tahap akhir, data uji akan dievaluasi menggunakan kedua model untuk membandingkan kinerja dari masing-masing model tersebut.

3.4 Pengumpulan Data

Pengumpulan *dataset* citra kaki diperoleh dari berbagai sumber daring yang bersifat publik, terpercaya, dan relevan secara medis. Sumber data meliputi hasil pencarian Google, seperti jurnal ilmiah (misalnya dari PubMed Central, Springer, dan ResearchGate), artikel kesehatan, situs layanan informasi terbuka seperti Openverse, CDC Public Health Image Library (PHIL), dan Wikimedia Commons yang menyediakan gambar berlisensi *Creative Commons*, serta *platform* media sosial dari sumber resmi dan terpercaya sebagai data tambahan. *Dataset* yang digunakan berupa citra dasar (*image*) kaki penderita filariasis (parasit) dan limfedema (non-parasit) dengan format PNG. Hasil pengumpulan untuk kedua kelas tersebut diperoleh sebanyak 520 citra, di mana masing-masing kelas berjumlah 260 citra. Citra yang diperoleh memiliki variasi dari segi sudut pengambilan serta segi resolusi, sehingga mampu memberikan beragam data untuk proses analisis. Seluruh penggunaan *dataset* ini, dilakukan sepenuhnya untuk tujuan penelitian non-komersial dengan tetap memperhatikan prinsip etika akademik, ketentuan hak cipta, serta memastikan tidak ada informasi pribadi atau identitas subjek yang disertakan dalam penelitian ini.

3.5 *Preprocessing*

Penelitian diawali dengan tahap *preprocessing* pada citra. Proses ini berperan penting untuk memastikan data yang digunakan dalam pelatihan model sudah sesuai dan konsisten. Tujuan utama dari tahap *preprocessing* adalah menyiapkan citra agar lebih terstruktur, mudah diolah, serta mampu mendukung kinerja model secara optimal. Beberapa proses yang dilakukan meliputi:

3.5.1 Hapus Latar (*Remove Background*)

Tahap pertama yang dilakukan dalam *preprocessing* adalah menghapus latar belakang (*background*) pada citra secara manual. Tujuan dari proses ini adalah agar objek utama pada citra terlihat lebih

jelas dan tidak terganggu oleh komponen visual di latar belakang yang tidak relevan. Dengan demikian, model nantinya akan lebih fokus dan mudah dalam mengenali ciri penting dari objek yang akan diteliti.

3.5.2 *Crop & Resize*

a) *Crop*

Proses *crop* dilakukan terlebih dahulu untuk menyesuaikan area objek yang diperlukan, yaitu mulai dari bagian lutut hingga telapak kaki. Proses ini dilakukan secara manual, karena citra kaki pada *dataset* memiliki variasi posisi dan jarak pengambilan gambar yang berbeda-beda.

b) *Resize*

Setelah proses *crop*, setiap citra akan dilakukan *resize* menggunakan *ImageDataGenerator* untuk menyeragamkan ukurannya menjadi 224×224 piksel sesuai dengan kebutuhan arsitektur CNN. Ukuran ini dipilih karena menjadi standar umum yang banyak digunakan model CNN untuk menjaga konsistensi ukuran *input*. Langkah-langkah ini dilakukan agar data yang digunakan pada tahap pelatihan dan pengujian memiliki format dan kualitas yang seragam.

3.5.3 *Split Data*

Tahap pembagian data dilakukan sebelum proses pelatihan model, di mana pada penelitian ini menggunakan pembagian data 70:15:15. Pembagian dilakukan dengan dua skema, yaitu secara manual *single split* (statis) yang dibagi menjadi data latih dan data uji, serta penerapan metode *cross validation (k-fold)* pada data latih untuk menghasilkan data valid internal untuk memastikan proporsi data pada setiap *fold* tetap seimbang. Setiap citra dipilih secara acak untuk

dimasukkan ke dalam kategori data latih, data valid, dan data uji. Data latih digunakan untuk membangun kemampuan model, data valid digunakan untuk memantau kinerja model selama proses pelatihan, dan data uji digunakan untuk mengevaluasi performa akhir model pada data yang belum pernah dilihat selama proses pelatihan.

3.5.4 Augmentasi Data

Pada tahap ini dilakukan augmentasi data, yaitu proses menambah variasi pada *dataset* dengan dilakukan penerapan berbagai transformasi, seperti *brightness*, *sharpness*, dan *shear*. Augmentasi hanya dilakukan pada data latih untuk memperkaya variasi data citra, sehingga model menjadi lebih tangguh, tidak mudah *overfitting*, dan mampu mengenali pola meski terdapat perbedaan pada citra masukan. Sementara itu, data valid dan uji tidak diterapkan augmentasi agar hasil evaluasi model sesuai dengan citra yang sebenarnya.

3.5.5 Normalisasi Citra

Pada tahap ini, nilai piksel pada setiap citra yang semula berada pada rentang 0 hingga 255 diproses melalui normalisasi berbasis standar ImageNet, dengan mengurangi nilai rata-rata tiap *channel*. Dalam penelitian ini, OpenCV dimanfaatkan untuk memproses gambar yang secara *default* menyajikan citra dalam format BGR, sehingga susunan *channel* berubah dari RGB ke BGR (Xian *et al.*, 2024). Proses ini dilakukan agar format citra konsisten dengan data yang digunakan saat ResNet dilatih dari awal (*pretrained*). Selain itu, normalisasi ini juga bertujuan untuk menyeimbangkan distribusi nilai piksel pada data dan mempercepat proses pelatihan model. Sehingga, algoritma *deep learning* bekerja lebih stabil dan efisien dalam proses pelatihan dengan distribusi nilai *input* yang berada pada rentang yang lebih konsisten.

3.6 *Convolution Neural Network (CNN)*

Pada tahap proses CNN berfungsi sebagai alat untuk mengenali pola visual yang memiliki dua peran berbeda. Untuk model *hybrid*, CNN berperan sepenuhnya sebagai ekstraksi fitur (*feature extraction*), mulai dari proses *convolutional* hingga *pooling layer*, di mana pada layer ini digunakan *Global Average Pooling (GAP)*, untuk menghasilkan vektor numerik yang selanjutnya diberikan kepada XGBoost sebagai *classifier*. Sementara itu untuk model dasar, CNN berperan sebagai model yang utuh, mulai dari proses *convolutional* hingga *fully connected layer* untuk melakukan klasifikasi secara mandiri.

3.7 *Extreme Gradient Boosting (XGBoost)*

Proses XGBoost pada tahap ini berfungsi sebagai *classifier* dalam arsitektur *hybrid* CNN-XGBoost. Proses ini diawali dengan menerima *input feature* berupa vektor numerik hasil dari proses *Global Average Pooling (GAP)*. XGBoost kemudian melakukan *boosting iteratif*, yaitu proses di mana pohon keputusan (*decision trees*) dibangun secara berulang, untuk memperbaiki kesalahan prediksi pohon sebelumnya. Setelah proses ini selesai, tahap akhir XGBoost menghasilkan output prediksi yang diklasifikasikan sebagai "Filariasis Limfatik (Parasit)" atau "Limfedema (Non-Parasit), untuk kemudian dievaluasi performanya. Selain itu, pada model CNN murni, proses klasifikasi tidak melibatkan XGBoost, karena tugas pengambilan keputusan secara penuh dilakukan oleh *fully connected layer*.

3.8 Model

Pada tahap ini dibangun dua jenis model, yaitu model CNN dan model CNN-XGBoost. Model CNN-XGBoost, merupakan model *hybrid* yang menggabungkan CNN dan XGBoost, dimana CNN berperan dalam mengekstraksi fitur dari citra, kemudian hasil fitur tersebut dijadikan *input* ke

XGBoost untuk proses klasifikasi akhir yang bertujuan untuk meningkatkan akurasi dan performa klasifikasi. Sedangkan model CNN, digunakan sebagai model dasar yang berfungsi untuk melihat performa dari arsitektur murni dalam melakukan pengklasifikasian citra. Model ini hanya memanfaatkan hasil ekstraksi dari fitur dan klasifikasi internal CNN di tahap sebelumnya. Kedua model ini dilatih menggunakan data latih dan disempurnakan dengan data valid untuk menghasilkan model final yang akan dievaluasi performanya menggunakan data uji.

3.8.1 *Hyperparameter Model CNN dan Hybrid CNN-XGBoost*

Hyperparameter merupakan variabel yang ditentukan sebelum proses pelatihan (*training*) model. Hal ini dilakukan untuk mengatur proses pembelajaran model agar tetap terkontrol secara optimal, baik dari segi kompleksitas, performa, dan kemampuan generalisasi. Tabel 4 menunjukkan *hyperparameter* yang digunakan pada model CNN dan *Hybrid CNN-XGBoost*.

Tabel 4. *Hyperparameter* yang Digunakan pada Kedua Model

Algoritma	Parameter	Keterangan
CNN Murni	batch_size	Jumlah citra yang digunakan dalam satu iterasi pelatihan (<i>training</i>).
	neuron	Jumlah <i>neuron</i> yang digunakan pada <i>Dense layer</i> untuk mengolah fitur sebelum <i>output</i>
	dropout	Persentase <i>neuron</i> yang dinonaktifkan sementara saat pelatihan untuk mencegah <i>overfitting</i> .
	learning_rate	Mengatur seberapa cepat model memperbarui bobot (<i>weight</i>) saat belajar dari data.
	epoch	Jumlah total iterasi model dalam mempelajari seluruh data latih.

Algoritma	Parameter	Keterangan
Hybrid CNN- XGBoost	patience	Jumlah <i>epoch</i> sebelum pelatihan dihentikan jika performa tidak membaik.
	n_estimators	Jumlah pohon keputusan (<i>decision tree</i>) yang akan dibangun selama pelatihan.
	max_depth	Kedalaman maksimum pohon dalam menangkap fitur.
	learning_rate	Seberapa cepat model memperbarui bobot saat belajar.
	subsample	Proporsi data latih yang digunakan secara acak untuk membangun setiap pohon.
	colsample_bytree	Proporsi fitur yang digunakan pada setiap pohon.
	min_child_weight	Jumlah data minimum yang dibutuhkan di setiap cabang pohon untuk mencegah pohon membentuk cabang terlalu spesifik.
	gamma	Membatasi <i>split</i> pohon agar tidak terlalu spesifik untuk mencegah <i>overfitting</i> .
	reg_alpha	Regularisasi L1 untuk mengurangi kompleksitas model.
	reg_lambda	Regularisasi L2 untuk menstabilkan bobot model.
early_stopping_rounds	Jumlah iterasi pohon untuk menghentikan pelatihan ketika model tidak membaik.	

3.9 Evaluation Model

Tahap *evaluation* model dilakukan untuk menilai kinerja model yang telah dibangun. Pada tahap ini digunakan *confusion matrix* untuk mengetahui prediksi benar dan salah dari setiap kelas yang ada. Hasil dari *confusion matrix* tersebut kemudian akan dihitung menggunakan metrik evaluasi seperti *accuracy*, *precision*, *recall*, serta *f1-score*, yang berfungsi sebagai dasar

kualitas model dalam melakukan klasifikasi. Berdasarkan nilai metrik evaluasi yang diperoleh, dilakukan perbandingan antara model *hybrid* CNN-XGBoost dengan model CNN murni untuk menentukan model yang memberikan performa kinerja terbaik.

V. SIMPULAN DAN SARAN

5.1 Simpulan

Berdasarkan penelitian yang telah dilakukan, terdapat beberapa kesimpulan terkait performa model yang dikembangkan dalam penelitian ini. Kesimpulan tersebut disusun berdasarkan hasil pengujian dan evaluasi yang diperoleh, sebagaimana diuraikan sebagai berikut.

- a. Penelitian ini berhasil mengimplementasikan model *hybrid* CNN-XGBoost menggunakan arsitektur ResNet50 untuk membedakan citra kaki antara filariasis limfatik (parasit) dan limfedema (non-parasit). Model ini mampu memanfaatkan ekstraksi fitur yang dihasilkan dari CNN dan menciptakan klasifikasi oleh XGBoost secara lebih efektif.
- b. Evaluasi kinerja menunjukkan bahwa model CNN mencapai *mean training accuracy* sebesar 87.11% dan *mean valid accuracy* mencapai 89.83%, dengan performa validasi terbaik pada *fold* ke-5 menghasilkan *accuracy* sebesar 94.87% untuk data uji. Sementara itu, model *hybrid* CNN-XGBoost menunjukkan performa lebih stabil dan unggul, dengan *mean training accuracy* 93.07% dan *mean validation accuracy* 91.83%, serta *fold* terbaik pada *fold* ke-1. Model *hybrid* ini mencapai *accuracy* sebesar 96.15%, yang menegaskan bahwa integrasi antara CNN dan XGBoost meningkatkan kemampuan klasifikasi menjadi lebih optimal.
- c. Penerapan augmentasi data terbukti meningkatkan performa model melalui peningkatan nilai *accuracy*. Teknik ini menunjukkan bahwa augmentasi membantu mengurangi risiko *overfitting* dengan menambah variasi data selama pelatihan. Hal ini membuat proses pembelajaran menjadi lebih stabil dan mampu meningkatkan kemampuan generalisasi model terhadap data uji yang belum dilihat sebelumnya.

5.2 Saran

Terdapat beberapa saran yang diberikan untuk pengembangan penelitian sehingga mencapai kinerja yang lebih optimal di masa yang akan datang, yang diuraikan sebagai berikut.

- a. Menggunakan arsitektur CNN atau pendekatan *hybrid* yang berbeda untuk menentukan model terbaik dalam melakukan klasifikasi penyakit kaki gajah (*elephantiasis*).
- b. Memperbanyak jumlah dan variasi *dataset* dengan menambahkan citra pada berbagai kondisi pencahayaan, sudut pengambilan, serta tingkat keparahan penyakit.
- c. Melakukan augmentasi data yang lebih variasi seperti rotasi, translansi, dan berbagai transformasi lainnya untuk memperkuat data latih dalam membedakan karakteristik citra.
- d. Melakukan optimasi *hyperparameter* pada model CNN dan XGBoost lebih mendalam untuk meningkatkan kinerja dan kemampuan generalisasi model terhadap data yang belum pernah dilihat.

DAFTAR PUSTAKA

- Abdullah, A., & Yahya, S. (2025). Kajian Pemanfaatan Deep Learning Dalam Pembelajaran Pada Lembaga Pelatihan. *Jurnal Transformasi*, 7(1), 25–41.
<https://transformasi.kemenag.go.id/index.php/journal/article/view/333>
- Ajiboye, A. O. (2024). Hybrid Skin Lesion Detection Integrating CNN and XGBoost for Accurate Diagnosis. *International Journal of Computer (IJC) International Journal of Computer*, 53(1), 14–71.
<https://ijcjournal.org/index.php/InternationalJournalOfComputer/index>
- Akbar, A. T., Saifullah, S., & Prapcoyo, H. (2024). Klasifikasi Ekspresi Wajah Menggunakan Covolutional Neural Network. *Jurnal Teknologi Informasi Dan Ilmu Komputer*, 11(6), 1399–1412.
<https://doi.org/10.25126/jtiik.202411888>
- Ali, Z. A., Abduljabbar, Z. H., Tahir, H. A., Sallow, A. B., & Almufti, S. M. (2023). Exploring the Power of eXtreme Gradient Boosting Algorithm in Machine Learning: a Review. *Academic Journal of Nawroz University*, 12(2), 320–334. <https://doi.org/10.25007/ajnu.v12n2a1612>
- Alsaleh, D., & Sainte, L. M. (2021). Arabic Text Classification Using Convolutional Neural Network and Genetic Algorithms. *IEEE Access*, 9, 91670–91685. <https://doi.org/10.1109/ACCESS.2021.3091376>
- Amjad, M., Ahmad, I., Ahmad, M., Wróblewski, P., Kamiński, P., & Amjad, U. (2022). Prediction of Pile Bearing Capacity Using XGBoost Algorithm: Modeling and Performance Evaluation. *Applied Sciences (Switzerland)*, 12(4). <https://doi.org/10.3390/app12042126>
- Anhar, & Putra, R. A. (2023). Perancangan dan Implementasi Self-Checkout System pada Toko Ritel menggunakan Convolutional Neural Network (CNN). *ELKOMIKA: Jurnal Teknik Energi Elektrik, Teknik Telekomunikasi, & Teknik Elektronika*, 11(2), 466–478.
<https://doi.org/10.26760/elkomika.v11i2.466>

- Ardanari, R., Stikes, A., & Kediri, K. H. (2020). Kajian Pustaka Penyakit Tropis Terabaikan: Aspek Holistik Pada Pasien Kusta dan Filariasis. *Jurnal Ilmu Kesehatan*, 8(2), 138–146.
- Ashari, Latif, N., & Astuti, A. (2019). Pengolahan Citra Digital Untuk Menentukan Bobot Sapi Menggunakan Metode Canny Edge Detection. *Jurnal Ilmiah Ilmu Komputer*, 5(1), 1–6. <http://ejournal.fikom-unasman.ac.id>
- Aswir, E. S., & Wasilah. (2024). Implementasi Algoritma YOLOv3 Menggunakan Fitur Ekstraktor ResNeXt. *Pepadun*, 5(2), 131–140. <https://doi.org/10.23960/pepadun.v5i2.215>
- Azmi, K., Defit, S., & Sumijan. (2023). Implementasi Convolutional Neural Network (CNN) Untuk Klasifikasi Batik Tanah Liat Sumatera Barat. *Jurnal Unitek*, 16(1), 28–40.
- Borman, P. (2018). Lymphedema diagnosis, treatment, and follow-up from the view point of physical medicine and rehabilitation specialists. *Turkish Journal of Physical Medicine and Rehabilitation*, 64(3), 179–197. <https://doi.org/10.5606/tftrd.2018.3539>
- Branda, F., Ali, A. Y., Ceccarelli, G., Albanese, M., Binetti, E., Giovanetti, M., Ciccozzi, M., & Scarpa, F. (2025). Assessing the Burden of Neglected Tropical Diseases in Low-Income Communities: Challenges and Solutions. *Viruses*, 17(1), 1–27. <https://doi.org/10.3390/v17010029>
- Briliantio, J., Santosa, N., Ardian, G., & Hakim, L. (2020). Penerapan Convolutional Neural Network untuk Handwriting Recognition pada Aplikasi Belajar Aritmatika Dasar Berbasis Web. *Teknik Informatika Unika St. Thomas*, 5(2), 137–146.
- Centers for Disease Control and Prevention. (1967). *Filariasis (elephantiasis) of the lower legs and feet (PHIL ID 16536)*. CDC Public Health Image Library. <https://phil.cdc.gov/Details.aspx?pid=16536>
- Chanwimalueang, N., & Ekataksin, W. (2013). Ease of Use and Effectiveness of Twisting Tourniquet Decongestive Technique for Gigantic Lymphedema Management. *24th International Society of Lymphology Congress*, 1–36. <https://www.lymphedema-thailand.com/15017078/ease-of-use-and-effectiveness-of-twisting-tourniquet-decongestive-technique-for-gigantic-lymphedema-management>
- Chanwimalueang, N., Ekataksin, W., Piyaman, P., Pattanapen, G., & Hanboon, B. K. (2015). Twisting Tourniquet© Technique: Introducing Schnogh, a novel device and its effectiveness in treating primary and secondary

- lymphedema of extremities. *Cancer Medicine*, 4(10), 1514–1524. <https://doi.org/10.1002/cam4.498>
- Damstra, R. J., Dickinson-Blok, J. L., & Voesten, H. G. J. M. (2020). Shaving Technique and Compression Therapy for Elephantiasis Nostras Verrucosa (Lymphostatic Verrucosis) of Forefeet and Toes in End-Stage Primary Lymphedema: A 5 Year Follow-Up Study in 28 Patients and a Review of the Literature. *Journal of Clinical Medicine*, 9(10), 1–13. <https://doi.org/10.3390/jcm9103139>
- Dani, A. R., & Handayani, I. (2024). Klasifikasi Motif Batik Yogyakarta Menggunakan Metode GLCM dan CNN. *Teknologi Terpadu*, 10(2), 142–156. <https://journal.nurulfikri.ac.id/index.php/jtt>
- de Sire, A., Fusco, N., Sajjadi, E., Lippi, L., Cisari, C., & Invernizzi, M. (2021). Lymphedema Rehabilitation Using Self-Adaptive Inelastic Compression in Breast Cancer: A Proof-of-Principle Study. *Applied Sciences (Switzerland)*, 11(4), 1–8. <https://doi.org/10.3390/app11041901>
- Dheo, R. P., Masfiah, S., & Maghfiroh, A. F. A. (2019). Perilaku Pencarian Pengobatan dan Perawatan Mandiri pada Penderita Filariasis di Kabupaten Wonosobo. *Promosi Kesehatan Indonesia*, 14(2), 22–135.
- DocHealer. (2017, November 17). *Grades of Lymphedema on Lower Limbs*. Wikimedia Commons. https://commons.wikimedia.org/wiki/File:Lower_Limb_Lymphedema.png
- DocHealer. (2018, November 23). *The 50 Years Old Male with Congenital Lymphedema 50 Years*. Wikimedia Commons. https://commons.wikimedia.org/wiki/File:Dr_Ming-Huei_Cheng%27s_lymphedema_Sample.png
- Duhon, B. H., Phan, T. T., Taylor, S. L., Crescenzi, R. L., & Rutkowski, J. M. (2022). Current Mechanistic Understandings of Lymphedema and Lipedema: Tales of Fluid, Fat, and Fibrosis. *International Journal of Molecular Sciences*, 23(12), 1–14. <https://doi.org/10.3390/ijms23126621>
- Emhandyksa, M., Soesanti, I., & Susilowati, R. (2023). Pengembangan Deep Learning untuk Sistem Deteksi Dini Komplikasi Kaki Diabetik Menggunakan Citra Termogram. *Jurnal Teknologi Informasi Dan Ilmu Komputer*, 10(6), 1241–1252. <https://doi.org/10.25126/jtiik.2023107382>
- Farmin, A. R., Wabula, F., Ilmandry, H., Isabela, M. L., Raharjo, M., & Sianipar, R. (2025). Literature Review The Impact of Machine Learning

- in Modern Industries. *Jurnal Ilmiah Mahasiswa*, 3(1), 177–182. <https://doi.org/10.59603/niantanasikka.v3i1.680>
- Fathurohman, A. (2021). Machine Learning Untuk Pendidikan: Mengapa dan Bagaimana. *JURNAL INFORMATIKA DAN TEKNOLOGI KOMPUTER*, 1(3), 57–62.
- Fuadah, Y. N., Ubaidullah, I. D., Ibrahim, N., Taliningsing, F. F., Sy, N. K., & Pramudhito, M. A. (2022). Optimasi Convolutional Neural Network dan K-Fold Cross Validation pada Sistem Klasifikasi Glaukoma. *Teknik Energi Elektrik, Teknik Telekomunikasi, & Teknik Elektronika*, 10(3), 728–741. <https://doi.org/10.26760/elkomika.v10i3.728>
- Garza, R., Skoracki, R., Hock, K., & Povoski, S. P. (2017). A comprehensive overview on the surgical management of secondary lymphedema of the upper and lower extremities related to prior oncologic therapies. *BMC Cancer*, 17, 1–9. <https://doi.org/10.1186/s12885-017-3444-9>
- Handoko, A. A., Rosid, M. A., & Indahyanti, U. (2024). Implementasi Convolutional Neural Network (CNN) Untuk Pengenalan Tulisan Tangan Aksara Bima. *SMATIKA JURNAL*, 14(1), 96–110. <https://doi.org/10.32664/smatika.v14i01.1196>
- Hanifi, S., Liu, X., Lin, Z., & Lotfian, S. (2020). A Critical Review of Wind Power Forecasting Methods-Past, Present and Future. *MDPI*, 13(15), 1–24. <https://doi.org/10.3390/en13153764>
- Hidaya, J. R., & Jemakmun. (2025). Implementasi Klasifikasi Citra Berbasis Tensorflow Untuk Mendeteksi Penyakit Tanaman Pada Aplikasi Agrosan. *JU*, 15(1), 124–130.
- Hidayat, T., Hadinata, E., Damanik, I. S., Vikki, Z., & Irvanizam, I. (2023). Implementation of Hybrid CNN-XGBoost Method for Leukemia Detection Problem. *Infolitika Journal of Data Science*, 1(1), 15–21. <https://doi.org/10.60084/ijds.v1i1.87>
- Ibrahim, N., Sa'idah, S., Hidayat, B., & Darana, S. (2022). Klasifikasi Grade Telur Ayam Negeri secara non- Invasive menggunakan Convolutional Neural Network. *Jurnal Teknik Energi Elektrik, Teknik Telekomunikasi, & Teknik Elektronika*, 10(2), 297. <https://doi.org/10.26760/elkomika.v10i2.297>
- Izza, F., Khotimah, K., Sadilah Arif, N., & Rifat Aqillah, N. (2024). Literature Review: Sistem Cerdas untuk Klasifikasi Penyakit Gigi Berdasarkan Citra Rontgen Metode: Convolutional Neural Network (CNN) dan Decision Support System (DSS). *Jurnal Artificial Inteligent Dan Sistem*

- Penunjang Keputusan*, 2(2), 142–145.
<https://jurnalmahasiswa.com/index.php/aidanspk>
- Janiesch, C., Zschech, P., & Heinrich, K. (2021). Machine Learning and Deep Learning. *Electronic Markets*, 31(3), 685–695.
<https://doi.org/10.1007/s12525-021-00475-2/Published>
- Jebastine, J. (2024). Detection and Classification of Brain Tumor Using Convolution Extreme Gradient Boosting Model and an Enhanced Salp Swarm Optimization. *Neural Processing Letters*, 56(2), 1–20.
<https://doi.org/10.1007/s11063-024-11590-4>
- Jiao, W., Hao, X., & Qin, C. (2021). The Image Classification Method with CNN-XGBoost Model Based on Adaptive Particle Swarm Optimization. *MDPI*, 12(4), 1–22. <https://doi.org/10.3390/info12040156>
- Jumadi, J., & Sartika, D. (2021). Pengolahan Citra Digital Untuk Identifikasi Objek Menggunakan Metode Hierarchical Agglomerative Clustering. *Jurnal Sains Dan Teknologi*, 10(2), 148–156.
- Jumrah, J., & Pulkadang, S. (2025). Application of Deep Learning Method in Learning. *Formosa Journal of Sustainable Research*, 4(6), 1011–1028.
<https://doi.org/10.55927/fjsr.v4i6.308>
- Kapoor, A. (2024). ML Approach: Algorithms, Real-World Applications and Research Directions. *International Journal of Commerce and Management*, 1(1), 1–23. <https://ssrn.com/abstract=5021508> or <http://dx.doi.org/10.2139/ssrn.5021508>
- Kapoor, C. (2019, April 22). An Elephantine Problem Haunts Bihar, Jharkhand, Uttar Pradesh. *DNA India*.
<https://www.dnaindia.com/india/report-an-elephantine-problem-haunts-bihar-jharkhand-uttar-pradesh-2741585>
- Kawashima, H., Sasame, A., Ogaki, Y., & Nakayama, T. (2025). High Prevalence of Nephrocalcinosis in Hypophosphatasia Patients with the ALPL c.1559del Gene Variant. *JMA Journal*, 8(1), 204–208.
<https://doi.org/10.31662/jmaj.2024-0138>
- Kermelita, D., Hadi, U. K., Soviana, S., & Tiuria, R. (2023). Karakteristik Kejadian dan Capaian Program Eliminasi Filariasis di Provinsi Bengkulu. *ACTA VETERINARIA INDONESIA*, 11(3), 175–181.
<http://www.journal.ipb.ac.id/indeks.php/actavetindones>
- Kurniadi, D., Shidiq, R. M., & Mulyani, A. (2025). Perbandingan Penggunaan Optimizer dalam Klasifikasi Sel Darah Putih Menggunakan Convolutional Neural Network. *Jurnal Nasional Teknik Elektro Dan*

Teknologi Informasi, 14(1), 77–86.
<https://doi.org/10.22146/jnteti.v14i1.17162>

Kurniawan, D., Wahyudi, M., Pujiastuti, L., Bina Sarana Informatika, U., & Antar Bangsa, S. (2024). Deteksi dan Prediksi Cerdas Penyakit Paru-Paru dengan Algoritma Random Fores. *Journal Computer Science*, 3(1).
<https://www.kaggle.com/>

Lee, S. Y., Loll, E. G., Hassan, A. E. S., Cheng, M., Wang, A., & Farmer, D. L. (2022). Genetic and Molecular Determinants of Lymphatic Malformations: Potential Targets for Therapy. *Journal of Developmental Biology*, 10(1), 1–13. <https://doi.org/10.3390/jdb10010011>

Lewsirirat, T., Titijaroonroj, T., Apichonbancha, S., Uthatham, A., Suwanruangsri, V., Suwan, N., Bokerd, S., Prapassaro, T., Chinchalongporn, W., & Yodrabum, N. (2025). Deep Learning-Based Classification of Lymphedema and Other Lower Limb Edema Diseases Using Clinical Images. *Scientific Reports*, 15(1), 1–20.
<https://doi.org/10.1038/s41598-025-97564-5>

Lisaldy, F., Ismail, & Iryani, D. (2024). Lex AI sebagai Tata Kelola Kecerdasan Buatan Sui Generis di Indonesia. *Technology and Economics Law Journal*, 3(1), 429–444. <https://doi.org/10.21143/TELJ.vol3.no1.1051>

Liu, W., Meng, Q., Li, Z., & Hu, X. (2021). Applications of Computer Vision In Monitoring the Unsafe Behavior of Construction Workers: Current Status and Challenges. *MDPI*, 11(9), 1–27.
<https://doi.org/10.3390/buildings11090409>

Liverpool School of Tropical Medicine. (2008). *Royal opening of CTID*.
www.liv.ac.uk/lstm/learning

Mackenzie, C. D., Kapa, D. R., Krishnasastry, S., Douglass, J., Hoerauf, A., & Ottesen, E. A. (2024). Managing Lymphedema Induced by Lymphatic Filariasis: Implementing and Improving Care at the Individual and Programmatic Levels. *American Journal of Tropical Medicine and Hygiene*, 111(4), 3–21. <https://doi.org/10.4269/ajtmh.23-0905>

Madjid, F. M., & Kurniawan, T. B. (2022). Prediksi Visibility Menggunakan LSTM dan MLP di Bandara Sultan Mahmud Badaruddin II Palembang. *Jurnal Sistem Informasi (JSI)*, 14(1), 2497–2507.
<http://ejournal.unsri.ac.id/index.php/jsi/index>

Mampitiya, L., Rathnayake, N., Leon, L. P., Mandala, V., Azamathulla, H. M., Shelton, S., Hoshino, Y., & Rathnayake, U. (2023). Machine Learning Techniques to Predict the Air Quality Using Meteorological Data in Two

- Urban Areas in Sri Lanka. *Environments - MDPI*, 10(8), 1–0. <https://doi.org/10.3390/environments10080141>
- Manju, N., Harish, B. S., & Prajwal, V. (2019). Ensemble Feature Selection and Classification of Internet Traffic using XGBoost Classifier. *International Journal of Computer Network and Information Security*, 11(7), 37–44. <https://doi.org/10.5815/ijcnis.2019.07.06>
- Marchica, P., Darpa, S., Magno, S., Rossi, C., Forcina, L., Capizzi, V., Oieni, S., Amato, C., Piazza, D., & Gebbia, V. (2021). Integrated Treatment of Breast Cancer-related Lymphedema: A Descriptive Review of the State of the Art. *Anticancer Research*, 41(7), 3233–3246. <https://doi.org/10.21873/anticancer.15109>
- McGinity, E. N., Bray, W. F., & Granzow, J. W. (2024). The Current State of Lymphedema Surgery. *Lymphatics*, 2(4), 212–227. <https://doi.org/10.3390/lymphatics2040017>
- Medical doctors. (2015, January). *Lymphedema Limbs*. Wikimedia Commons. https://commons.wikimedia.org/wiki/File:Lymphedema_limbs.JPG
- Mitjà, O., Marks, M., Bertran, L., Kollie, K., Argaw, D., Fahal, A. H., Fitzpatrick, C., Fuller, L. C., Garcia Izquierdo, B., Hay, R., Ishii, N., Johnson, C., Lazarus, J. V., Meka, A., Murdoch, M., Ohene, S. A., Small, P., Steer, A., Tabah, E. N., ... Asiedu, K. (2017). Integrated Control and Management of Neglected Tropical Skin Diseases. *PLoS Neglected Tropical Diseases*, 11(1), 1–13. <https://doi.org/10.1371/journal.pntd.0005136>
- Mubarok, M. H., & Septian, F. (2025). Prediksi GDP dengan RF dan XGBoost Berdasarkan Aspek Sosial, Ekonomi, dan Lingkungan. *The 4th MDP Student Conference 2025*, 4(1), 338–346.
- Mulyani, A., Kurniadi, D., & Rahmat, A. (2025). Optimization of Malaria Cell Image Classification Using Pretrained Resnet50 Architecture with Data Augmentation and Fine-Tuning. *Engineering Science Letter*, 4(2), 83–92. <https://doi.org/10.56741/IISTR.esl.001244>
- Murdiansyah, D. T. (2024). Prediksi Stroke Menggunakan Extreme Gradient Boosting. *JIKO (Jurnal Informatika Dan Komputer)*, 8(2), 419–426. <https://doi.org/10.26798/jiko.v8i2.1295>
- Nagy, K. S., & Kapusta, J. (2023). TwIdw—A Novel Method for Feature Extraction from Unstructured Texts. *MDPI*, 13(11), 1–15. <https://doi.org/10.3390/app13116438>

- Narahari, S. R. (2020). *Skin diseases in Integrative Medicine*. Charak Samhita New Edition (1st Ed. Pp.203). CSRTSDC. <https://doi.org/10.47468/CSNE.2021.e01.s09.074>
- Nurdin, M. A., Wihandika, R. C., & Utaminingrum, F. (2020). Deteksi Pergerakan Arah Mata menggunakan Convolution Neural Network berdasarkan Facial Landmark. *L Pengembangan Teknologi Informasi Dan Ilmu Komputer*, 4(10), 3338–3345. <http://j-ptiik.ub.ac.id>
- Nurhadi, & Hendrik, B. (2024). Tinjauan Sistematis Peran Jaringan Syaraf Tiruan dan Deep Learning Dalam Diagnosis Demam Berdarah dan Tifus. *Jurnal Informatika, Manajemen Dan Komputer*, 16(2), 276–288.
- Nurhakiki, J., Yahfizham, Y., William, J., Ps, I. V, Estate, M., Percut, K., Tuan, S., & Serdang, K. D. (2024). Studi Kepustakaan: Pengenalan 4 Algoritma Pada Pembelajaran Deep Learning Beserta Implikasinya. *Jurnal Pendidikan Berkarakter*, 2(1), 270–281. <https://doi.org/10.51903/pendekar.v2i1.598>
- Ong, W., Lee, A., Tan, W. C., Fong, K. T. D., Lai, D. D., Tan, Y. L., Low, X. Z., Ge, S., Makmur, A., Ong, S. J., Ting, Y. H., Tan, J. H., Kumar, N., & Hallinan, J. T. P. D. (2024). Oncologic Applications of Artificial Intelligence and Deep Learning Methods in CT Spine Imaging—A Systematic Review. *MDPI*, 16(17), 1–31. <https://doi.org/10.3390/cancers16172988>
- Osei-Poku, P., Tritten, L., Fordjour, F., & Kwarteng, A. (2025). Cell-free DNA as a complementary diagnostic tool for neglected tropical diseases towards achieving the WHO NTDs elimination by 2030. *The Journal of Liquid Biopsy*, 7, 1–7. <https://doi.org/10.1016/j.jlb.2024.100283>
- Paide, M. L., Siagian, J. L. S., & Arianto, F. M. (2023). Faktor-Faktor Yang Berhubungan Dengan Kejadian Filariasis Di Distrik Kwoor Kabupaten Tambrau. *Health Information : Jurnal Penelitian*, 15(2), 1–10.
- Pakiding, D., Selao, A., & Wahyuddin, W. (2025). Implementasi Computer Vision dalam Mendeteksi Penyakit pada Tanaman Cabai dan Tomat Menggunakan Algoritma Convolutional Neural Networks. *Indonesian Journal of Machine Learning and Computer Science*, 5(3), 841–850. <https://doi.org/10.57152/malcom.v5i3.1989>
- Pasupuleti, V., Thuraka, B., Kodete, C. S., & Malisetty, S. (2024). Enhancing Supply Chain Agility and Sustainability through Machine Learning: Optimization Techniques for Logistics and Inventory Management. *Logistics*, 8(3), 1–16. <https://doi.org/10.3390/logistics8030073>

- Paudel, P., Paudel, N., & Timalsina, M. (2025). Right Lower Limb Lymphatic Aplasia in Lymphoscintigraphy: A Case Report. *Journal of Medical Case Reports*, *19*(1), 1–6. <https://doi.org/10.1186/s13256-025-05374-0>
- Primasari, M. (2020). Lymphedema: Diagnosis dan Terapi. *MEDICINUS*, *33*(22), 43–49.
- Putra, A. E., Kartini, K., & Sari, A. P. (2024). Metode Convolutional Neural Network dan Extreme Gradient Boost untuk Mengklasifikasi Penyakit Pneumonia. *JASIEK (Jurnal Aplikasi Sains, Informasi, Elektronika Dan Komputer)*, *6*(1), 33–40. <https://doi.org/10.26905/jasiek.v6i1.11464>
- Qadriah, L., & Cut, L. S. (2024). Deep Learning Untuk Mendeteksi Objek Buah Dengan Algoritma Convolutional Neural Network (CNN) Menggunakan MatLab. *Literasi Informatika*, *3*(2), 1–10.
- Rahmi, I. R., Sutningsih, D., Hestingsih, R., & Saraswati, L. D. (2022). Faktor-Faktor yang Berhubungan dengan Kasus Filariasis di Indonesia : Sistematis Review. *Jurnal Epidemiologi Kesehatan Komunitas*, *7*(2), 501–521.
- Rajamanickam, A., & Babu, S. (2025). Unraveling the Dynamics of Human Filarial Infections: Immunological Responses, Host Manifestations, and Pathogen Biology. *Pathogens*, *14*(3), 1–29. <https://doi.org/10.3390/pathogens14030223>
- Ramadhan, M. A. R., Ananta, T. A. N., Zakkyfriza, A. A., H, I. V., & Fauzan, Y. N. (2024). Perbandingan Jumlah Layer Pada Convolutional Neural Network Untuk Meningkatkan Akurasi Dalam Klasifikasi Gambar. *Jurnal Riset Sistem Informasi Dan Teknik Informatika*, *2*(5), 211–217. <https://doi.org/10.61132/mercurius.v2i4.301>
- Raup, A., Ridwan, W., Khoeriyah, Y., Supiana, & Zaqiah, Q. Y. (2022). Deep Learning dan Penerapannya dalam Pembelajaran. *Jurnal Ilmiah Ilmu Pendidikan*, *5*(9), 3258–3267. <http://Jiip.stkipyapisdampu.ac.id>
- Real Hope for Haiti [@realhopeforhaiti]. (2024, December 21). *Lymphatic Filariasis (Elephantiasis) as a Neglected Tropical Disease*. Instagram. https://www.instagram.com/p/DD2YU_3KQjs/?img_index=2
- Ridhovan, A., & Suharso, A. (2022). Penerapan Metode Residual Network (ResNet) Dalam Klasifikasi Penyakit Pada Daun Gandum. *JUPI (Jurnal Ilmiah Penelitian Dan Pembelajaran Informatika)*, *7*(1), 58–65.
- Rozali, C., Zein, A., & Sita Eriana, E. (2024). Artificial Intelligence (AI) Dimasa Depan : Tantangan Dan Peluang. *Jurnal Informatika Utama*, *2*(2), 66–71. <https://doi.org/10.55903/jitu.v2i1.177>

- Salehin, I., & Kang, D. K. (2023). A Review on Dropout Regularization Approaches for Deep Neural Networks within the Scholarly Domain. *Electronics (Switzerland)*, *12*(14), 1–23. <https://doi.org/10.3390/electronics12143106>
- Samat, N. A., Salleh, M. N. M., & Ali, H. (2020). The Comparison of Pooling Functions in Convolutional Neural Network for Sentiment Analysis Task. *Advances in Intelligent Systems and Computing*, *978*, 202–210. https://doi.org/10.1007/978-3-030-36056-6_20
- Sathyanarayanan, S. (2024). Confusion Matrix-Based Performance Evaluation Metrics. *African Journal of Biomedical Research*, *27*(4), 4023–4031. <https://doi.org/10.53555/ajbr.v27i4s.4345>
- Setiawan, F. B., Kurnianingsih, F. A., Riyadi, S., & Pratomo, L. H. (2021). Pattern Recognition untuk Deteksi Posisi pada AGV Berbasis Raspberry Pi (Pattern Recognition for AGV's Position Detection Based on Raspberry Pi). *Jurnal Nasional Teknik Elektro Dan Teknologi Informasi*, *10*(1), 49–56.
- Simatupang, E. (2023). Bulletin of Data Science Jaringan Syaraf Tiruan Pengaruh Konsumsi Durian Pada Ibu Hamil Dengan Menggunakan Metode Backpropagation. *Bulletin of Data Science*, *3*(2), 132–142. <https://ejurnal.seminar-id.com/index.php/bulletinds>
- Singh, S., Shukla, R. K., Singh, A., & Acharya, S. (2024). Chronic Cellulitis in Elephantiasis: A Rare Debilitating Phenomenon. *Cureus*, *16*(7), 1–11. <https://doi.org/10.7759/cureus.65855>
- Solehudin, M., & Al-Nur, W. R. (2025). Dampak Penggunaan Meta Artificial Intelligence Pada WhatsApp Terhadap Cara Berpikir Mahasiswa Saat Diskusi di Kelas. *Jurnal Kependidikan*, *13*(1), 45–58. <https://doi.org/10.24090/jk.v13i1.13189>
- Supiyandi, Mujib, M. A., Azis, K., Abdillah, R., & Iskandar, S. N. (2024). Penerapan Teknologi Pengolahan Citra dalam Analisis Data Visual pada Tinjauan Komprehensif. *Jurnal Kendali Teknik Dan Sains*, *2*(3), 179–187. <https://doi.org/10.59581/jkts-widyakarya.v2i3.3796>
- Sutisna, T., Raharja, A. R., Hariyadi, E., & Putra, H. C. P. (2024). Penggunaan Computer Vision untuk Menghitung Jumlah Kendaraan dengan Menggunakan Metode SSD (Single Shoot Detector). *Journal Of Social Science Research*, *4*(2), 1–8.

- Tetko, I. V., van Deursen, R., & Godin, G. (2024). Be aware of overfitting by hyperparameter optimization! *Journal of Cheminformatics*, *16*(1), 1–11. <https://doi.org/10.1186/s13321-024-00934-w>
- Thongsuwan, S., Agarwal, P., & Jaiyen, S. (2019). A Deep One-Pass Learning based on Pre-Training Weights for Smartphone-Based Recognition of Human Activities and Postural Transitions. *Communications in Mathematics and Applications*, *10*(3), 541–560. <https://doi.org/10.26713/cma.v10i3.1269>
- Tjahjadi, E. V., & Santoso, B. (2023). Klasifikasi Malware Menggunakan Teknik Machine Learning. *Jurnal Ilmiah Ilmu Komputer*, *2*(1), 1–11. <https://www.kaggle.com/datasets/amauricio/pe-files-malwares>.
- Varghese, S. A. (2021). Secondary Lymphedema: Pathogenesis. *Journal of Skin and Sexually Transmitted Diseases*, *3*, 7–15. https://doi.org/10.25259/jstsd_3_2021
- Wang, R., Wang, H., He, Z., Zhu, J., & Zuo, H. (2024). WeldNet: a Lightweight Deep Learning Model for Welding Defect Recognition. *Research Paper*, *68*, 2963–2974. <https://doi.org/10.21203/rs.3.rs-3828347/v1>
- Wang, W., Chakraborty, G., & Chakraborty, B. (2021). Predicting the Risk of Chronic Kidney Disease (CKD) Using Machine Learning Algorithm. *MDPI*, *20*(1), 1–17. <https://doi.org/10.3390/app11010202>
- Wang, X., & Li, H. (2023). Investigation of Deep Learning Based Semantic Segmentation Models for Autonomous Vehicles. *IJACSA) International Journal of Advanced Computer Science and Applications*, *14*(11), 2023. www.ijacsa.thesai.org
- Watanabe, S. (2025). *Tree-Structured Parzen Estimator: Understanding Its Algorithm Components and Their Roles for Better Empirical Performance*. <http://arxiv.org/abs/2304.11127>
- Westenra, S. L. (n.d.). *Elephantiasis in the right leg of an inhabitant of the West Indies*. Wellcome Collection. Retrieved March 19, 2026, from <https://wellcomecollection.org/works/p76nzjds/images?id=jd3vf6tf&resultPosition=7>
- Xian, Z., Huang, R., Towey, D., & Yue, C. (2024). Convolutional Neural Network Image Classification Based on Different Color Spaces. *Tsinghua Science and Technology*, *30*(1), 402–417. <https://doi.org/10.26599/tst.2024.9010001>
- Yulianti, E. H., Soesanto, O., & Sukmawaty, Y. (2022). Penerapan Metode Extreme Gradient Boosting (XGBOOST) pada Klasifikasi Nasabah

Kartu Kredit. *JOMTA Journal of Mathematics: Theory and Applications*, 4(1), 21–26.

Yuziani, Rahayu, M. S., & Mellaratna, W. P. (2021). Hubungan Tingkat Pengetahuan Dengan Kepatuhan Pengobatan Massal Filariasis di Kecamatan Baktiya Aceh Utara. *Jurnal Kedokteran Dan Kesehatan Malikussale*, 7(1), 95–111.