

**KLASIFIKASI PENYAKIT BATU GINJAL MELALUI CITRA CT SCAN
DENGAN PENERAPAN METODE CONVOLUTIONAL NEURAL
NETWORK ARSITEKTUR DENSENET-121**

LAPORAN TUGAS AKHIR

Laporan ini Disusun untuk Memenuhi Salah Satu Syarat Memperoleh Gelar Sarjana
Strata 1 (S1) pada Program Studi Teknik Informatika Fakultas Teknologi Industri
Universitas Islam Sultan Agung Semarang



DISUSUN OLEH :

VANI AULIA PRAMESTI

32602000068

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS ISLAM SULTAN AGUNG
SEMARANG**

2024

FINAL PROJECT
CLASSIFICATION OF KIDNEY STONE DISEASE THROUGH CT SCAN
IMAGE WITH THE APPLICATION OF CONVOLUTIONAL NEURAL
NETWORK METHOD OF DENSENET-121 ARCHITECTURE

Proposed to complete the requirement to obtain a bachelor's degree (S-1) at
Informatics Engineering Departement of Industrial Technology Faculty
Sultan Agung Islamic University



Arranged By:
VANI AULIA PRAMESTI
32602000068

MAJORING OF INFORMATICS ENGINEERING
INDUSTRIAL TECHNOLOGY FACULTY
SULTAN AGUNG ISLAMIC UNIVERSITY
SEMARANG
2024

LEMBAR PENGESAHAN PEMBIMBING

Laporan Tugas Akhir dengan judul "Klasifikasi Penyakit Batu Ginjal Melalui Citra CT Scan dengan Penerapan Metode Convolutional Neural Network Arsitektur DenseNet-121" ini disusun oleh :

Nama : Vani Aulia Pramesti

NIM : 32602000068

Program Studi : Teknik Informatika

Telah disahkan oleh dosen pembimbing pada :


Hari : Kamis


Tanggal : 22 Agustus 2024

Mengesahkan,

Pembimbing I

Pembimbing II



Bagus Satrio Waluyo Poetro, S.Kom., M.Cs
NIDN. 1027118801


Ir. Sri Mulyono, M.Eng
NIDN. 0626066601

Mengetahui,

Ketua Program Studi Teknik Informatika
Fakultas Teknologi Industri
Universitas Islam Sultan Agung




Ir. Sri Mulyono, M.Eng
NIDN. 0626066601

LEMBAR PENGESAHAN PENGUJI

Laporan tugas akhir dengan judul “Klasifikasi Penyakit Batu Ginjal Melalui Citra CT Scan dengan Penerapan Metode Convolutional Neural Network Arsitektur DenseNet-121” ini telah dipertahankan di depan dosen penguji Tugas Akhir pada :

Hari : Kamis
Tanggal : 22 Agustus 2024



Penguji I

Penguji II

Moch Taufik, S.T., MIT

NIDN. 0622037502

Badiah, S.T., M.Kom

NIDN. 0619018701

SURAT PERNYATAAN KEASLIAN TUGAS AKHIR

Yang bertanda tangan dibawah ini :

Nama : Vani Aulia Pramesti

NIM : 32602000068

Judul Tugas Akhir : Klasifikasi Penyakit Batu Ginjal Melalui Citra CT dengan Penerapan Metode Convolutional Neural Network Arsitektur DenseNet-121

Dengan bahwa ini saya menyatakan bahwa judul dan isi Tugas Akhir yang saya buat dalam rangka menyelesaikan Pendidikan Strata Satu (S1) Teknik Informatika tersebut adalah asli dan belum pernah diangkat, ditulis ataupun dipublikasikan oleh siapapun baik keseluruhan maupun sebagian, kecuali yang secara tertulis diacu dalam naskah ini dan disebutkan dalam daftar pustaka, dan apabila di kemudian hari ternyata terbukti bahwa judul Tugas Akhir tersebut pernah diangkat, ditulis ataupun dipublikasikan, maka saya bersedia dikenakan sanksi akademis. Demikian surat pernyataan ini saya buat dengan sadar dan penuh tanggung jawab.

Semarang, 22-08-2024

Yang Menyatakan,



Vani Aulia Prasmeti

PERNYATAAN PERSETUJUAN PUBLIKASI KARYA ILMIAH

Saya yang bertanda tangan dibawah ini:

Nama : Vani Aulia Pramesti

NIM : 32602000068

Program Studi : Teknik Informatika

Fakultas : Teknologi Industri

Dengan ini menyatakan Karya Ilmiah berupa Tugas akhir dengan Judul : **Klasifikasi Penyakit Batu Ginjal Melalui Citra CT Scan dengan Penerapan Metode Convolutional Neural Network Arsitektur DenseNet-121** Menyetujui menjadi hak milik Universitas Islam Sultan Agung serta memberikan Hak bebas Royalti Non-Eksklusif untuk disimpan, dialihmediakan, dikelola dan pangkalan data dan dipublikasikan diinternet dan media lain untuk kepentingan akademis selama tetap menyantumkan nama penulis sebagai pemilik hak cipta. Pernyataan ini saya buat dengan sungguh-sungguh. Apabila dikemudian hari terbukti ada pelanggaran Hak Cipta/Plagiarisme dalam karya ilmiah ini, maka segala bentuk tuntutan hukum yang timbul akan saya tanggung secara pribadi tanpa melibatkan Universitas Islam Sultan agung.

Semarang, 22 Agustus 2024

Yang menyatakan,



3000
METRAI
TEMPEL
E#ECCALX253154409

Vani Aulia Pramesti

KATA PENGANTAR

Puji syukur penulis panjatkan kepada Allah SWT, yang telah memberikan rahmat, taufik serta hidayah-Nya, sehingga Tugas Akhir dengan judul Klasifikasi Penyakit Batu Ginjal Melalui Citra CT Scan dengan Penerapan Metode Convolutional Neural Network dapat terselesaikan.

Tanpa lupa penulis mengucapkan terima kasih untuk beberapa pihak yang telah membantu secara materi, pikiran, dan dukungan mental. Saya selaku penulis mengucapkan terima kasih kepada :

1. Rektor UNISSULA Bapak Prof. Dr. H. Gunarto, S.H., M.H., yang mengizinkan penulis menimba ilmu di kampus ini.
2. Dekan Fakultas Teknologi Industri Ibu Dr. Ir. Hj. Novi Marlyana, S.T., M.T., IPU., ASEAN.Eng.
3. Dosen Pembimbing I Bapak Bagus Satrio Waluyo Poetro, S.Kom., M.Cs yang telah membimbing dan memberikan banyak nasehat serta saran.
4. Dosen Pembimbing II Bapak Ir. Sri Mulyono, M.Eng yang telah membimbing dan memberikan banyak nasehat serta saran.
5. Orang tua dan keluarga penulis yang telah membantu secara materi dan mengizinkan untuk menyelesaikan laporan ini.
6. Dan kepada semua pihak yang tidak dapat saya sebutkan satu persatu.

Penulis menyadari bahwa dalam penulisan laporan ini masih terdapat banyak kekurangan, untuk itu penulis mengharap kritik dan saran dari pembaca untuk sempurnanya laporan ini. Semoga dengan ditulisnya laporan ini dapat menjadi sumber ilmu bagi setiap pembaca.

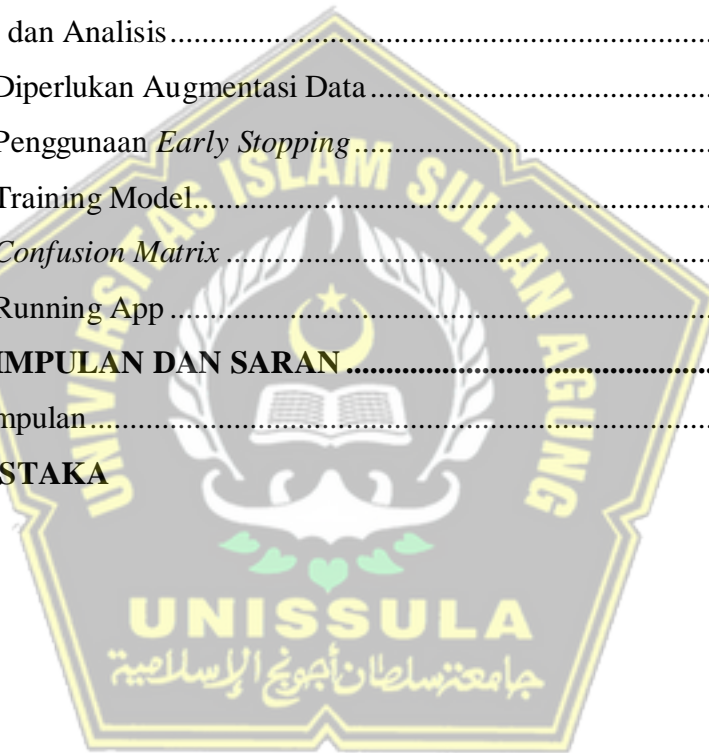
Semarang,

Vani Aulia Pramesti

DAFTAR ISI

COVER	Error! Bookmark not defined.
LEMBAR PENGESAHAN PEMBIMBING	Error! Bookmark not defined.
LEMBAR PENGESAHAN PENGUJI	Error! Bookmark not defined.
SURAT PERNYATAAN KEASLIAN TUGAS AKHIR	Error! Bookmark not defined.
PERNYATAAN PERSETUJUAN PUBLIKASI KARYA ILMIAH	Error! Bookmark not defined.
KATA PENGANTAR	vi
DAFTAR ISI	vii
DAFTAR TABEL	ix
DAFTAR GAMBAR	x
ABSTRAK	xi
BAB I PENDAHULUAN	1
1.1 Latar Belakang.....	1
1.2 Perumusan Masalah.....	3
1.3 Pembatasan Masalah	3
1.4 Tujuan.....	3
1.5 Manfaat.....	3
1.6 Sistematika Penulisan.....	4
BAB II TINJAUAN PUSTAKA DAN DASAR TEORI	6
2.1 Tinjauan Pustaka	6
2.2 Dasar Teori	9
2.2.1 Batu Ginjal	9
2.2.2 CT Scan.....	9
2.2.3 <i>Machine Learning</i>	10
2.2.4 <i>Preprocessing</i>	12
2.2.5 <i>Convolutional Neural Network</i>	13

2.2.6	<i>DenseNet-121</i>	14
2.2.7	<i>Flowchart</i>	15
BAB III METODE PENELITIAN		18
3.1	Studi Literatur.....	18
3.2	Perolehan Data.....	19
3.3	Perancangan Sistem.....	20
BAB IV HASIL DAN ANALISIS PENELITIAN		32
4.1	Hasil dan Analisis.....	32
4.1.1	Diperlukan Augmentasi Data	32
4.1.2	Penggunaan <i>Early Stopping</i>	35
4.1.3	Training Model.....	38
4.1.4	<i>Confusion Matrix</i>	40
4.1.5	Running App	41
BAB V KESIMPULAN DAN SARAN		47
5.1	Kesimpulan.....	47
DAFTAR PUSTAKA		



DAFTAR TABEL

Tabel 2. 1 Fungsi dan Atribut Flowchart	15
Tabel 3. 1 Pembagian Data	21
Tabel 4. 1 Hasil Training Model	37
Tabel 4. 2 Perhitungan Confusion Matrix	40
Tabel 4. 3 Hyperparameter Tuning Hasil Dari Beberapa experiment	44



DAFTAR GAMBAR

Gambar 2. 1 CT Scan	10
Gambar 2. 2 Proses Machine Learning (Aziz dkk., 2023)	11
Gambar 2. 3 Arsitektur Convolutional Neural Network (CNN) (Satwikayana dkk., 2021)	14
Gambar 2. 4 Dense Block (Khultsum dan Taufik, 2023)	15
Gambar 3. 1 Tahap Penelitian	18
Gambar 3. 2 Pengambilan Dataset Batu Ginjal	19
Gambar 3. 3 Performa Model.....	20
Gambar 3. 4 Contoh Rotasi 20 Derajat Ke Kanan	24
Gambar 3. 5 Contoh Rotasi 20 Derajat Ke Kiri	24
Gambar 3. 6 Pre-Training Model.....	27
Gambar 4. 1 Parameter Augmentasi Data.....	33
Gambar 4. 2 Eksperimen Menggunakan Augmentasi.....	34
Gambar 4. 3 Eksperimen Tanpa Augmentasi	34
Gambar 4. 4 Pelatihan Berhenti Karena Adanya Peningkatan Nilai Validasi Loss....	36
Gambar 4. 5 Grafik Akurasi Training dan Validasi	38
Gambar 4. 6 Grafik Loss Training Model dan Validasi Model	39
Gambar 4. 7 Confusion Matrix	40
Gambar 4. 8 Dashboard.....	41
Gambar 4. 9 Browse Files	42
Gambar 4. 10 Klasifikasi Normal	43
Gambar 4. 11 Hasil Klasifikasi Batu Ginjal.....	43

ABSTRAK

Batu ginjal adalah masalah kesehatan yang cukup serius di mata dunia. Jika tidak ditangani dengan baik, batu ginjal dapat menyebabkan nyeri hebat, gangguan fungsi ginjal, dan komplikasi lainnya. Komputer tomografi *CT Scan* resolusi tinggi dapat digunakan untuk membuat diagnosis batu ginjal. Hanya membutuhkan beberapa detik ataupun menit dalam melakukan pengambilan gambar *CT Scan* pada pasien, dan dokter akan menjelaskan hasilnya beberapa hari atau minggu setelah prosedur dilakukan. Klasifikasi penyakit batu ginjal dan ginjal normal melalui citra *CT Scan* dilakukan dalam penelitian ini dengan penerapan algoritma *Convolutional Neural Network* dan arsitektur *DenseNet-121* untuk mempercepat waktu diagnosa. Pengambilan data diperoleh melalui situs *Kaggle*. Data yang digunakan dan telah diolah dalam penelitian ini yaitu 1.101 data *training*, 110 data validasi, 110 data *testing* data tersebut untuk masing-masing kelas. Pra-pemrosesan data dilakukan *resizing* 224x224 dan implementasi augmentasi data. Bangun model untuk melakukan klasifikasi penyakit batu ginjal menerapkan metode CNN dengan arsitektur *DenseNet-121* yang kemudian diproses *deployment local* menggunakan *Streamlit*. Model yang telah dibangun kemudian diuji menggunakan data *testing* dan *confusion matrix*. Hasil penerapan metode CNN dengan arsitektur *DenseNet-121* pada penelitian ini yaitu *accuracy* 98,18%, *precision* 96,36%, *recall* 100%, *f1-score* 98,14%.

Kata Kunci: *Convolutional Neural Network*, *DenseNet-121*, Klasifikasi

ABSTRACT

Kidney stones are a serious health problem in the eyes of the world. If not treated properly, kidney stones can cause severe pain, impaired kidney function, and other complications. High-resolution computer tomography CT scans can be used to make a diagnosis of kidney stones. It only takes a few seconds or minutes to take CT scan images of the patient, and the doctor will explain the results a few days or weeks after the procedure. Classification of kidney stone disease and normal kidney through CT scan images is performed in this study by applying Convolutional Neural Network algorithm and DenseNet-121 architecture to speed up the diagnosis time. Data collection was obtained through the Kaggle website. The data used and processed in this study are 1,101 training data, 110 validation data, 110 testing data for each class. Pre-processing of data is done resizing 224x224 and implementation of data augmentation. Build a model to classify kidney stone disease using the CNN method with DenseNet-121 architecture which is then processed for local deployment using streamlit. The model that has been built is then tested using testing data and confusion matrix. The results of applying the CNN method with DenseNet-121 architecture in this study are accuracy 98.18%, precision 96.36%, recall 100%, f1-score 98.14%.

Keywords: *Convolutional Neural Network*, *DenseNet-121*, Classification

BAB I

PENDAHULUAN

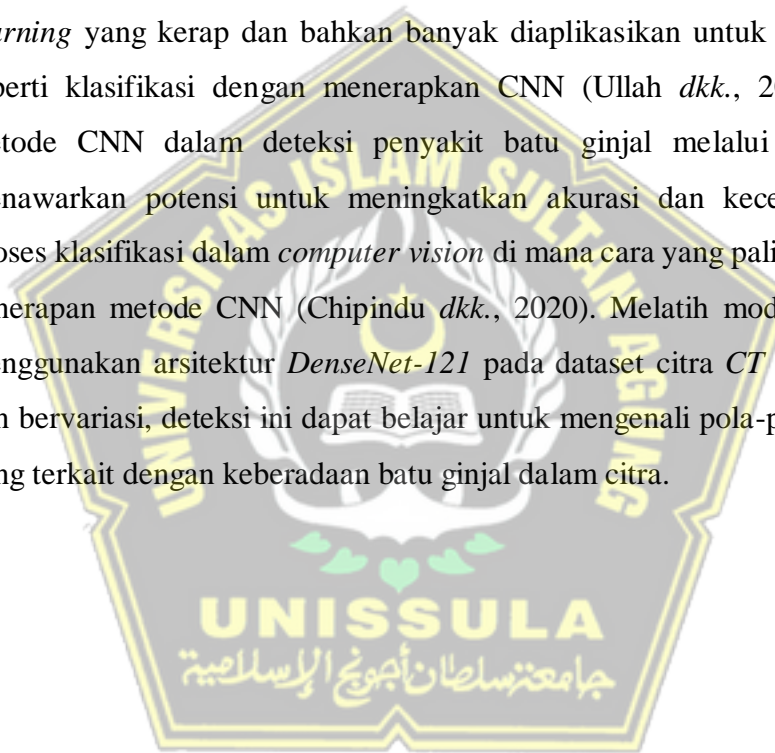
1.1 Latar Belakang

Gangguan ginjal yakni persoalan kesehatan cukup umum di dunia. Massa keras terdiri dari kristal yang menumpuk di organ ginjal, hal ini yang membawa dampak terjadinya batu ginjal pada ginjal seseorang. Batu ginjal dapat terbentuk di ginjal, ureter, kandung kemih, saluran kemih, dan uretra (Hadibrata dan Suharmanto, 2022). Gejala yang timbul pada penderita penyakit ini patut ditangani dengan sungguh-sungguh, jika tidak ditangani dengan baik maka yang terjadi ialah menyebabkan nyeri hebat, gangguan fungsi ginjal, hingga komplikasi. Pola makan dan bentuk gaya hidup yang buruk menjadi salah satu unsur yang mendatangkan penyakit batu ginjal (Ferraro *dkk.*, 2020).

Tes pencitraan, pemeriksaan fisik, dan riwayat medis adalah langkah pertama dalam diagnosis batu ginjal. *Computer Tomografi (CT Scan)* resolusi tinggi *CT* dapat dilakukan dari ginjal ke kandung kemih untuk melihat ukuran dan bentuk batu ginjal. Peralatan *CT Kidney-Ureter-Bladder (KUB)* dapat mendeteksi dan mengukur batu ginjal, menentukan jenis dan komponen utamanya, dengan ginjal kanan sebagai lokasi yang paling banyak terdapat batu (Bano *dkk.*, 2022). Pemeriksaan *Magnetic Resonance Imaging (MRI)* dan Ultrasonografi (USG) digunakan untuk diagnosis dengan menggunakan radiasi non-ionisasi yakni untuk pelayanan pencitraan diagnostik (Rahmawati *dkk.*, 2024). Dokter atau ahli radiologi dapat melakukan analisis manual terhadap gambar *CT*, sebuah proses yang membutuhkan banyak waktu dan rawan kesalahan yang disebabkan oleh faktor manusia. Namun, karena gambar *CT* sangat detail dan kompleks, kesalahan individu masih mungkin terjadi, yang dapat mempengaruhi akurasi diagnosis (Hasby Bik *dkk.*, 2024).

Dalam beberapa tahun terakhir, pertumbuhan dalam bidang kecerdasan buatan telah membuka peluang baru pada peningkatan sistem deteksi penyakit

berbasis citra medis. Dengan menggunakan pembelajaran *deep learning*, gambar medis dapat dianalisis untuk membantu ahli radiologi dan dokter menetapkan prosedur diagnostik dan pengobatan yang lebih efektif. *Convolutional Neural Network* (CNN) sangat efektif untuk tujuan ini karena mereka unggul dalam mengenali pola dan fitur dalam gambar dan memanfaatkan lapisan operasi konvolusi. Maka CNN ialah metode *deep learning* yang bisa dan cocok digunakan untuk analisis gambar medis (Gunawan dan Setiawan, 2022). *Deep learning* yang kerap dan bahkan banyak diaplikasikan untuk *computer vision* seperti klasifikasi dengan menerapkan CNN (Ullah dkk., 2021). Penerapan metode CNN dalam deteksi penyakit batu ginjal melalui citra *CT Scan* menawarkan potensi untuk meningkatkan akurasi dan kecepatan diagnosa. Proses klasifikasi dalam *computer vision* di mana cara yang paling berhasil yaitu penerapan metode CNN (Chipindu dkk., 2020). Melatih model CNN dengan menggunakan arsitektur *DenseNet-121* pada dataset citra *CT Scan* yang besar dan bervariasi, deteksi ini dapat belajar untuk mengenali pola-pola karakteristik yang terkait dengan keberadaan batu ginjal dalam citra.



1.2 Perumusan Masalah

1. Bagaimana melakukan klasifikasi penyakit batu ginjal dari hasil penerapan metode *Convolutional Neural Network* menggunakan arsitektur *DenseNet-121*?
2. Bagaimana membuat sistem untuk menerapkan metode *Convolutional Neural Network* arsitektur *DenseNet-121* terhadap prediksi untuk mempercepat diagnosa kepada pasien?

Adanya rumusan masalah tersebut, penelitian ini mampu menyampaikan interpretasi yang makin terperinci perihal potensi dan keterbatasan penerapan *convolutional neural network* dalam mendeteksi batu ginjal menggunakan citra *CT Scan*.

1.3 Pembatasan Masalah

1. Penelitian ini hanya melakukan klasifikasi ginjal normal dan ginjal dengan batu ginjal. Tidak untuk memprediksi tingkat keparahan penyakit batu ginjal.
2. Pada penelitian ini tidak melakukan deteksi seberapa besar ukuran batu ginjal.

1.4 Tujuan

Tujuan daripada penelitian ini yakni melakukan klasifikasi penyakit batu ginjal dan ginjal normal menggunakan citra *CT Scan* dengan penerapan metode *Convolutional Neural Network* arsitektur *DenseNet-121* untuk mempercepat waktu diagnosa dan hasil agar mendapatkan penanganan lebih lanjut kepada pasien.

1.5 Manfaat

Pada penelitian ini yakni menghasilkan manfaat untuk memprediksi adanya batu ginjal melalui klasifikasi citra *CT Scan* ginjal normal dan ginjal dengan batu ginjal dengan menerapkan metode *Convolutional Neural Network* yang dapat mempercepat waktu dan hasil diagnosa sehingga pasien dapat penanganan lebih serius.

1.6 Sistematika Penulisan

Sistematika penulisan yang diterapkan atas laporan tugas akhir ini disusun dengan struktur yang dijabarkan seperti:

BAB I : PENDAHULUAN

Dalam bab pendahuluan ini dijelaskan latar belakang pemilihan judul, perumusan masalah, pembatasan masalah, serta tujuan dari penelitian.

BAB II : TINJAUAN PUSTAKA

Menampilkan hasil penelitian sebelumnya serta dasar teori untuk mendukung konsep yang berkaitan dengan metode yaitu *Convolutional Neural Network* yang dibangun dengan arsitektur DenseNet-121.

BAB III : METODE PENELITIAN

Menyajikan prosedur yang diterapkan dalam penelitian, yang mencakup seluruh tahapan mulai dari pengumpulan data menggunakan *CT Scan* ginjal, proses klasifikasi data hingga *deployment* pada model.

BAB IV : HASIL DAN ANALISIS PENELITIAN

Menjabarkan hasil serta analisa pada klasifikasi penyakit batu ginjal normal dan dengan batu ginjal yang menerapkan metode CNN dengan arsitektur DenseNet-121 disajikan dalam BAB IV.

BAB V : KESIMPULAN DAN SARAN

Menguraikan kesimpulan serta saran dari penulis terkait penelitian yang dilakukan, mulai dari langkah-langkah awal hingga mencapai konklusi akhir, serta memberikan pandangan dan saran yang relevan berdasarkan hasil yang diperoleh.

BAB II

TINJAUAN PUSTAKA DAN DASAR TEORI

2.1 Tinjauan Pustaka

Convolutional Neural Network (CNN) menjadi sangat populer di bidang deep learning. Metode ini mengubah dimensi gambar tanpa mengubah kualitas gambar (Omori, 2020). Klasifikasi tambahan menerapkan CNN dalam kombinasi pendekatan pembelajaran transfer (*transfer learning*). Keunggulan dari metode ini terletak pada kemampuan untuk melakukan pra-pelatihan menggunakan informasi yang telah dipelajari oleh model CNN yang telah dilatih sebelumnya. Informasi ini kemudian dapat disesuaikan dan diterapkan pada tugas klasifikasi yang baru. Teknik ini tidak hanya dapat meningkatkan kinerja model secara keseluruhan, tetapi juga mengurangi waktu pelatihan yang diperlukan, serta menghasilkan performa model yang lebih baik. (Gulavnai dan Patil, 2019).

Pada penelitian yang dilakukan oleh (Cahya dkk., 2021), melakukan Klasifikasi penyakit mata dengan menerapkan metode CNN arsitektur *Alexnet*. Pada penelitian tersebut memaparkan klasifikasi mata normal, katarak, *glaucoma*, dan *retina disease*. Klasifikasi dilakukan secara 3 tahap yaitu yang pertama *preprocessing* dan menghasilkan ukuran citra menjadi 224x224px. Pada tahap kedua yaitu mengekstraksi *feature* yaitu lapisan konvolusi, lapisan *pooling*, dan lapisan *fully connected*. Pada tahap ketiga yaitu penggunaan *Convolutional Neural Network* dengan menggunakan 150 *epoch*. Hasil dari menerapkan metode *Convolutional Neural Network*, mampu menghasilkan model dengan akurasi sebesar 98.37%.

Penelitian lainnya dilakukan oleh (Yohannes dan Al Rivan, 2022), melakukan klasifikasi beberapa penyakit kanker kulit yakni karsinoma sel badal, keratosis aktinik, *nevy* melanositik, keratosis jinak, dan melanoma. Klasifikasi penyakit kanker kulit yang dilakukan menerapkan metode CNN-SVM. Metode *Convolutional Neural Network* di penelitian tersebut digunakan sebagai ekstraksi

fitur yang dimana menggunakan arsitektur VGG-19 dan *ResNet-50*. *Support Vector Machine* (SVM) di penelitian tersebut diaplikasikan untuk mengklasifikasi kernel linear dan *Radial Basis Function* (RBF) lalu dioptimasi dengan *random* dan *grid*. Termuat enam belas skenario total yang dimana terbagi terdiri dari dua skenario gambar dengan mengimplementasikan *preprocessing resize* dan *preprocessing patch* yang kemudian digunakan ke dalam model. Dengan menggunakan kernel linear optimasi *random* dan nilai *grid*, penelitian tersebut menghasilkan hasil akurasi terbaik, *preprocessing patch* VGG-19, dengan skor akurasi 65,33%, *recall* 65,33%, *precision* 68,51%, dan *f1-score* 65,77%.

Penerapan metode Convolutional Neural Network (CNN) juga pernah dilakukan oleh (Harahap *dkk.*, 2022), yang di mana penelitian ini dilakukan oleh penulis bertujuan untuk memahami seperti apa ketika menerapkan algoritma CNN sebagai deteksi penyakit ginjal berdasarkan citra medis yang digunakan. Pada penelitian ini menggunakan arsitektur AlexNet untuk membangun model. Penelitian ini dibuat menggunakan bahasa pemrograman *python* yang nantinya akan diimplementasikan ke sistem berbasis *web* dengan menggunakan framework *Flask*. Sistem ini dapat digunakan untuk mendeteksi jenis-jenis penyakit ginjal. Data berisi 4 kelas kategori penyakit yaitu ginjal normal, kista ginjal, tumor ginjal, dan batu ginjal. Penelitian ini memperoleh akurasi keseluruhan dari *f1-score* sebesar 68%.

Penelitian sebelumnya yang dilakukan (Ahmed *dkk.*, 2024), penelitian tersebut melakukan klasifikasi citra sinar X-KUB (*Kidney-Ureter-Bladder*) secara akurat. Pada penelitian ini menerapkan model *Transfer Learning* (TL) dengan VGG16 yang dikembangkan untuk kecerdasan buatan dan mengkategorikan mana yang batu ginjal dan mana yang ginjal normal. Pada penelitian tersebut menggunakan teknik *Layer-Wise Relevance Propagation* (LRP). Sebuah teknik kecerdasan buatan yang dapat dijelaskan *Explainable Artificial Intelligence*, dapat meningkatkan transparansi dan efektivitas model

guna solusi perihal pada penelitian tersebut. Dalam penelitian tersebut tampak bahwa model mencapai akurasi pengujian sebesar 97,41%.

Penelitian sebelumnya yang dilakukan (Sulaksono *dkk.*, 2020). Pada penelitian tersebut menerapkan data citra medis *CT Scan* dengan kelas ginjal normal dan ginjal batu ginjal. Dalam penelitian tersebut menerapkan algoritma *Convolutional Neural Network* (CNN) meliputi beberapa *layer* tradisional tanpa menggunakan arsitektur tambahan. CNN digunakan pada penelitian ini untuk melakukan layer ekstraksi fitur pada citra *input* dan sampai dengan *layer* klasifikasi. Data yang diimplementasikan sebagai pelatihan data pada penelitian tersebut yaitu 2.338 citra ginjal normal dan 2.390 citra batu ginjal yang diperoleh dari data publik (*Kaggle*) dan data primer Rumah Sakit. Data yang digunakan untuk *testing* augmentasi dengan jumlah yaitu 540 citra ginjal normal dan 446 citra batu ginjal. Kemudian sebelum melakukan proses pelatihan dan pengujian pada model, diperlukan proses konversi citra menjadi *grayscale* agar citra *input* dapat terbaca dengan baik. Penelitian tersebut diterapkan dengan dua uji analisa yaitu *testing* yang pertama dengan augmentasi data publik dari *Kaggle* dan pengujian kedua dengan augmentasi data primer rumah sakit. Dari penelitian yang dilakukan mendapatkan hasil validasi augmentasi data publik dari *kaggle* pada batu ginjal diperoleh tingkat akurasi *cross validation* yaitu sebesar 99,69%. Sedangkan validasi augmentasi data primer rumah sakit pada batu ginjal diperoleh tingkat akurasi *cross validation* yaitu sebesar 45,43%.

2.2 Dasar Teori

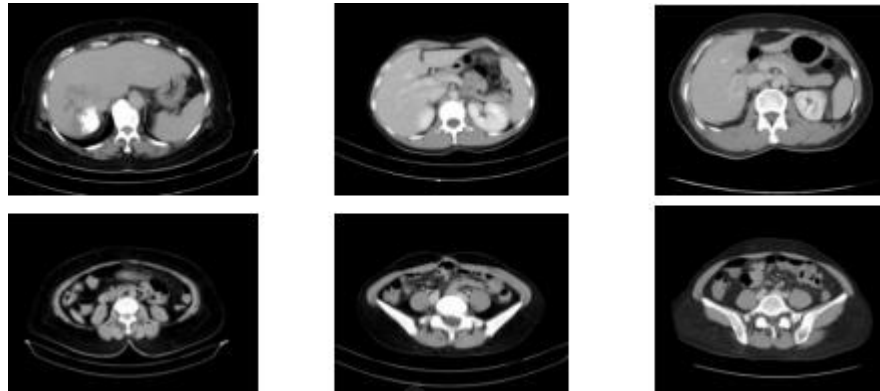
2.2.1 Batu Ginjal

Kidney Stone (Batu ginjal) yaitu salah satu penyebab paling umum kelainan saluran kemih, dan dibandingkan dengan batu di saluran kemih bagian atas, batu ginjal dilaporkan lebih sering di berbagai rumah sakit di Indonesia. Perubahan ini menunjukkan bahwa batu ginjal telah menjadi masalah utama dalam kesehatan banyak pasien (Utami *dkk.*, 2020).

Batu ginjal yaitu batu kecil atau kristal padat yang terbentuk pada organ ginjal atau saluran kemih, seperti ureter, *pelvis*, dan kandung kemih. Batu ginjal dapat timbul dari berbagai zat yang ditemukan dalam air kemih, semacam kalsium, oksalat, asam urat, dan fosfat. Unsur-unsur yang dapat meningkatkan akibat terjadinya batu ginjal meliputi minimnya konsumsi cairan, diet tinggi protein atau garam, riwayat keluarga, dan beberapa kondisi medis tertentu. Salah satu dampak yang paling umum terjadi adalah rasa sakit yang sangat terasa dan sering, terutama ketika batu mulai berpindah dari ginjal menuju saluran kemih. Nyeri punggung bawah dan perut bagian bawah atau pangkal paha akibat dari luruhnya batu ginjal ke saluran kemih (Handayani, 2020).

2.2.2 CT Scan

Pencitraan medis saat ini telah mengalami perkembangan yang sangat pesat, baik dari segi teknologi maupun dalam praktik klinis. Salah satu perkembangan yang terjadi yaitu *Computed Tomography (CT Scan)*. Ide dan pembaruan dianggap sebagai hal biasa di unit radiologi. Perkembangan yang terjadi pada teknologi di bidang radiologi yaitu *Computed Tomography (CT) Scan* (Ramadhani *dkk.*, 2023).



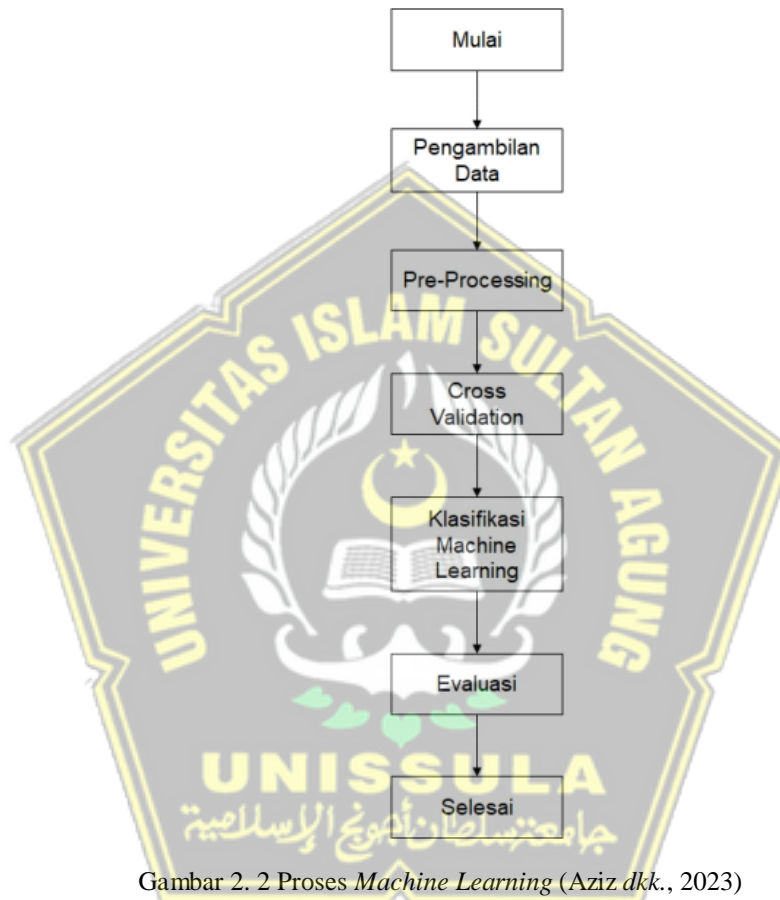
Gambar 2. 1 CT Scan

Computerized Tomography (CT) scan adalah tindakan dalam bidang radiologi yang menggunakan gelombang sinar-X sebagai alat *scanning*. Dengan dukungan sistem komputer, *CT scan* dapat menghasilkan gambaran berbentuk potongan anatomi tubuh manusia tanpa adanya tumpang-tindih organ. (Samei dan Pelc, 2020). *CT Scan* bisa memberikan informasi yang sangat berguna kepada dokter untuk mendiagnosis berbagai kondisi medis, termasuk cedera, tumor, pendarahan, atau infeksi. Proses *CT Scan* melibatkan pengambilan serangkaian gambar menggunakan sinar-X dari berbagai sudut, yang kemudian diolah oleh komputer untuk membuat gambar tiga dimensi dari organ atau struktur yang sedang diperiksa. *CT non-kontras* digunakan untuk mengevaluasi ginjal dan sistem saluran kemih, serta untuk mendeteksi kondisi seperti hematoma dan perforasi usus. Sementara itu, *CT Colonography* digunakan untuk memeriksa nefrolitiasis dan sebagai alat skrining untuk kanker kolorektal. Oleh karena itu, pemeriksaan *CT Scan abdomen* dibagi menjadi dua jenis: *CT non-kontras* dan *CT kontras*, masing-masing dengan tujuan dan aplikasi yang spesifik. (Octavia dkk., 2024).

2.2.3 Machine Learning

Salah satu cabang dari kecerdasan buatan yang dikenal sebagai pembelajaran mesin, di mana ketika komputer yang tidak diprogram secara eksplisit dapat mempelajari pola dan informasi dari data yang diberikan. Dengan

demikian, komputer dapat memahami dan membuat prediksi tanpa perlu instruksi langsung dari manusia. Tujuannya yakni untuk mengembangkan model dan algoritma yang bisa mengenali bentuk dan pola pada data yang ada dan melakukan prediksi berdasarkan data yang diberikan.



Gambar 2. 2 Proses *Machine Learning* (Aziz dkk., 2023)

Metode *machine learning*, salah satu sub bidang kecerdasan buatan yang berpotensi berdampak pada bidang lainnya selain teoritis ilmu komputer, *statistic*, matematika, dan termasuk yang saat ini banyak dilakukan yaitu di bidang kesehatan. Pada dasarnya, tujuan *machine learning* adalah untuk mengembangkan algoritma yang dapat menjalankan sistem pembelajaran mandiri dengan sedikit masukan dari rata-rata orang. Pembelajaran mesin dibagi menjadi dua jenis konsep pembelajaran. *Machine learning* dibagi menjadi dua jenis rancangan pembelajaran. Pertama adalah *Supervised Learning*, yaitu

metode pembelajaran mesin yang membentuk fungsi berlandaskan data pelatihan yang ada. Hal ini metode tersebut bisa dikatakan mempunyai data latih yang detail dan sudah tersedia dan terklasifikasi dengan baik. Data ini digunakan sebagai model data ketika proses pengujian dijalankan terhadap data pengujian baru dan memberikan hasil keluaran awal berlandaskan data tersebut. Kedua ialah *Unsupervised Learning*. Ini adalah metode *machine learning* yang mencoba merepresentasikan bentuk masukan yang berawal dari data pelatihan. Salah satu perbedaan dengan pembelajaran terbimbing adalah tidak adanya klasifikasi data masukan (Saputra, 2024).

2.2.4 Preprocessing

Pemrosesan awal data atau data *preprocessing* adalah serangkaian langkah atau tahapan yang dilakukan terhadap data mentah sebelum data tersebut digunakan untuk analisis lebih lanjut atau pengembangan model. Tujuan utama dari pra-pemrosesan data adalah untuk meningkatkan kualitas data, memastikan keakuratan hasil analisis, dan memperbaiki masalah serta cela yang mungkin muncul pada data mentah (Rahayu dkk., 2024). Selama tahap *preprocessing* analisis gambar, tujuannya adalah untuk meningkatkan dan mengoptimalkan hasil pelatihan yang pada akhirnya mengarah pada peningkatan akurasi pada hasil akhir. Peningkatan ini membantu mencapai akurasi yang lebih tinggi pada hasil akhir (Putra dkk, 2023).

Saat melakukan pra-pemrosesan data gambar, beberapa transformasi umum dilakukan menggunakan operasi matriks. Semua transformasi ini, termasuk shear atau geser, rotasi, *zoom*, translasi, dan penskalaan, dapat digambarkan sebagai matriks transformasi yang diterapkan pada koordinat piksel gambar. Di bawah ini adalah rumus-rumus matriks untuk beberapa transformasi *preprocessing*:

- 1) Rotasi

Rotasi gambar dapat dilakukan dengan sudut tertentu. Untuk rotasi gambar dengan sudut θ (dalam radian), matriks transformasi-nya:

$$R(\theta) = \begin{bmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (1)$$

2) Translasi

Translasi gambar menggeser gambar ke posisi baru. Untuk translasi gambar dengan perpindahan Δx dan Δy .

$$T(\Delta x, \Delta y) = \begin{bmatrix} 1 & 0 & \Delta x \\ 0 & 1 & \Delta y \\ 0 & 0 & 1 \end{bmatrix} \quad (2)$$

3) *Shear* (Geseran)

Untuk *shear* gambar dengan faktor geser S_x di sumbu x atau S_y di sumbu y.

- Shear di Sumbu x:

$$H_x(S_x) = \begin{bmatrix} 1 & S_x & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (3)$$

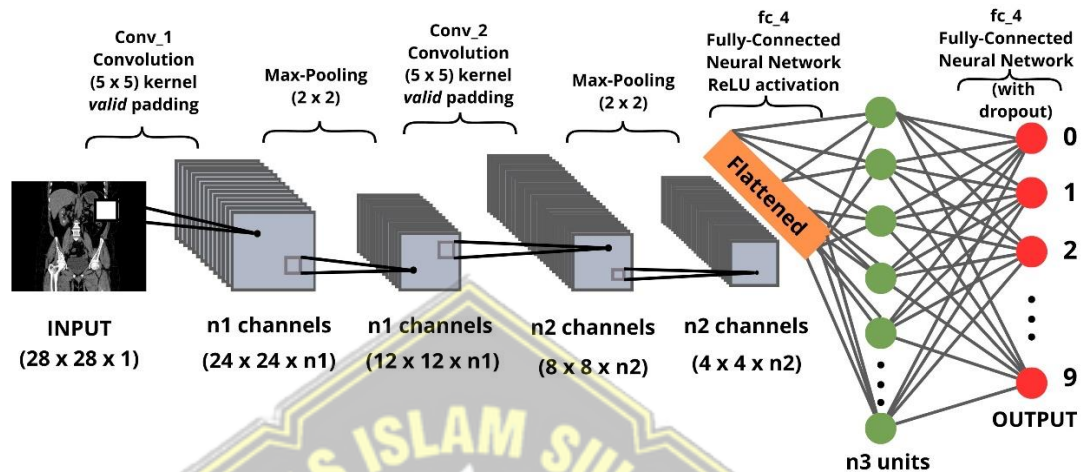
- Shear di Sumbu y:

$$H_y(S_y) = \begin{bmatrix} 1 & 0 & 0 \\ S_y & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (4)$$

2.2.5 Convolutional Neural Network

Metode *Convolutional Neural Network* (CNN) adalah jenis kerangka kerja jaringan saraf tiruan yang dimaksudkan untuk mengolah dan menganalisis data berbentuk grid, seperti gambar. CNN dapat mengenali pola dan fitur dalam data melalui lapisan konvolusi, yang memungkinkan model untuk belajar secara efektif dari struktur dan detail visual. Sangat membantu dalam pengolahan data besar seperti gambar dan video. CNN dirancang khusus untuk memecahkan masalah pengenalan pola dalam data spasial, dimana model ini seringkali memiliki struktur bertahap. Pendekatan CNN terinspirasi oleh cara kerja gambar

manusia, di mana informasi diproses secara hierarki dari fitur sederhana misalnya garis dan sudut, hingga fitur yang lebih kompleks misalnya wajah atau objek.



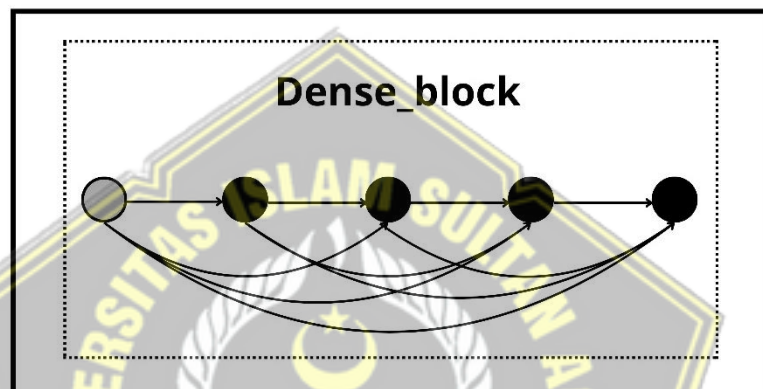
Gambar 2. 3 Arsitektur Convolutional Neural Network (CNN) (Satwikayana *dkk.*, 2021)

Desain CNN yang diperlihatkan Gambar 2.3 merupakan konsep dari keseluruhan CNN yang dibagi menjadi tiga bagian utama yakni *input*, pembelajaran fitur dan klasifikasi. Proses input yaitu dari gambar digital, setelah melakukan proses pembelajaran fitur untuk memproses gambar, diterapkan model konvolusi, yang memperoleh nilai matriks yang ada di dalam gambar. Proses trigger menerapkan ReLU (*Rectified Linear Unit*) dan kemudian proses konvolusi dan aktifkan ReLU. Proses pooling adalah pengumpulan data saat semua vektor dibuat satu dan mendapatkan nilai tertinggi dari satu gambar. CNN mempunyai banyak lapisan, digunakan sebagai langkah proses penyaringan untuk setiap proses pencitraan digital.

2.2.6 DenseNet-121

DenseNet-121 (*Densely Connected Convolutional Network*) adalah arsitektur jaringan saraf konvolusional yang diperkenalkan untuk meningkatkan kinerja dalam pengolahan citra, di mana tiap lapisan terhubung dengan semua lapisan yang lebih dalam pada jaringan tersebut. Misalnya, lapisan pertama terkoneksi dengan *layer* kedua, ketiga, keempat, dan seterusnya. Tujuan dari

pendekatan ini adalah untuk memaksimalkan alur informasi antara lapisan-lapisan terhadap jaringan (Vellaichamy *dkk.*, 2021). *DenseNet* memecahkan masalah dengan menyediakan jalur langsung untuk aliran bertahap melalui tiap-tiap lapisan yang terhubung secara padat. *DenseNet* menggunakan apa yang disebut dengan “*Dense Blocks*”. Dalam setiap *Dense Block*, setiap lapisan menerima *input* dari semua lapisan sebelumnya.




Gambar 2. 4 *Dense Block* (Khultsum dan Taufik, 2023)

Pada Gambar 2.4 merupakan alur atau bentuk kerja daripada *dense block* yang mana lapisan demi lapisan saling terkoneksi.


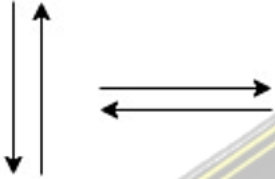
2.2.7 Flowchart

Diagram alur adalah representasi grafis dari suatu proses atau algoritma yang digunakan untuk memvisualisasikan langkah-langkah dan alur kerja suatu sistem atau proses. *Flowchart* membantu dalam memahami, menganalisis, dan mendokumentasikan proses tersebut dengan cara yang lebih sederhana dan jelas (Khesya, 2021).

Tabel 2. 1 Fungsi dan Atribut *Flowchart*

NO	BENTUK	NAMA	FUNGSI SIMBOL
1		<i>Terminal</i>	Akhir dan awalan dalam prosedur.

NO	BENTUK	NAMA	FUNGSI SIMBOL
2		<i>Output/Input</i>	Proses <i>input</i> atau <i>output</i> apa pun macam perangkatnya.
3		<i>Process</i>	Langkah-langkah proses.
4		<i>Decision</i>	Menunjukkan bahwa suatu kondisi tertentu menghasilkan dua kemungkinan, yaitu ya atau tidak.
5		<i>Offline Connector</i>	Menggambarkan hubungan antara proses satu dengan proses lainnya yang berada di halaman berbeda.
6		<i>Predefined Process</i>	Menggantikan kepastian penyimpanan untuk diproses dan untuk menerapkan awal harga.

NO	BENTUK	NAMA	FUNGSI SIMBOL
7		<i>Punched Card</i>	<i>Input</i> ke sistem dari kartu, atau hasil <i>output</i> ditulis ke kartu.
8		<i>Flow</i>	Menunjukkan jalannya arus suatu proses.



BAB III

METODE PENELITIAN

Penelitian ini mengimplementasikan metode algoritma CNN (*Convolutional Neural Network*). Pada penelitian ini berfokus pada implementasi klasifikasi melalui citra medis yaitu *CT Scan* pada kasus medis penyakit batu ginjal. Sistem yang dibuat dapat menghasilkan klasifikasi antara citra medis dengan ginjal normal maupun citra medis dengan batu ginjal. Langkah-langkah yang harus diimplementasikan pada penelitian ini yaitu:



Gambar 3. 1 Tahap Penelitian

3.1 Studi Literatur

Acuan terhadap referensi yang digunakan dari beberapa jurnal, makalah, artikel, dan skripsi terdahulu serta memperluas ide dan pengetahuan melalui situs

web sebagai landasan acuan dan referensi untuk melakukan penelitian ini. Pada tinjauan ini dilakukan bertujuan untuk mempelajari teori sebagai materi untuk menganalisa dan mengenali konsep dari sistem, olah data, klasifikasi, dan metode *Convolutional Neural Network* (CNN) dengan arsitektur *DenseNet-121*.

3.2 Perolehan Data

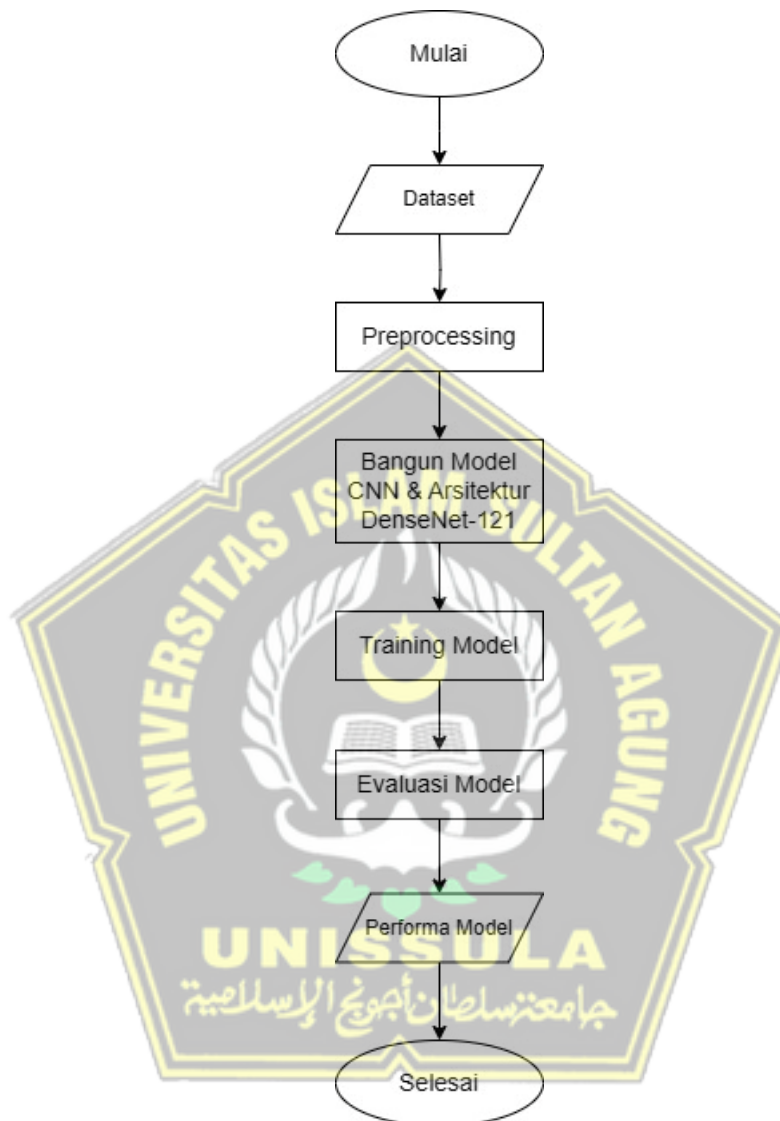
The image shows a screenshot of a Kaggle dataset page. At the top, it displays the creator's name 'MD NAZMUL ISLAM AND 1 COLLABORATOR' and 'UPDATED 3 YEARS AGO'. There are buttons for 'New Notebook' and 'Download (2 GB)'. The main title is 'CT KIDNEY DATASET: Normal-Cyst-Tumor and Stone'. Below the title, it says 'Dataset to detect auto Kidney Disease Analysis'. There are tabs for 'Data Card', 'Code (46)', 'Discussion (7)', and 'Suggestions (0)'. The 'About Dataset' section includes 'Context' (CT KIDNEY DATASET: Normal-Cyst-Tumor and Stone), 'Content', 'Usability' (10.00), 'License' (Data files © Original Authors), and 'Expected update frequency'. The page is overlaid with a large watermark of the Universitas Islam Sultan Agung logo.

Gambar 3. 2 Pengambilan Dataset Batu Ginjal

Sebelum melanjutkan ke proses olah data, langkah pertama penelitian ini yaitu mengumpulkan data. Data citra *CT Scan* ini digunakan untuk *training* data dan *testing*. Dataset berupa citra medis batu ginjal yang diperoleh melalui situs Kaggle¹. Dataset meliputi kondisi ginjal normal dan ginjal dengan batu ginjal. Untuk total data yang akan digunakan yaitu 2.642.

¹ *CT KIDNEY DATASET: Normal-Cyst-Tumor and Stone*

3.3 Performa Model



Gambar 3. 3 Performa Model

1) *Dataset*

Langkah dalam proses pengembangan model adalah membagi data menjadi beberapa bagian. Data tersebut dibagi menjadi tiga set utama yaitu set pelatihan (*training set*), set validasi (*validation set*), dan set pengujian (*test set*). Set pelatihan digunakan untuk melatih model, sementara set validasi digunakan untuk mengevaluasi kinerja model selama proses

pelatihan agar model dapat disesuaikan dan diperbaiki. Setelah model dilatih dan divalidasi, set pengujian digunakan untuk mengukur kinerja akhir model dan memastikan bahwa model dapat melakukan prediksi dengan baik pada data yang belum pernah dilihat sebelumnya.

Tabel 3. 1 Pembagian Data

No	Nama Data	Data Ginjal Normal (Jumlah)	Data Batu Ginjal (Jumlah)	Total Data
1.	Data Latih	1.101	1.101	2.202
2.	Data Uji	110	110	220
3.	Data Validasi	110	110	220
TOTAL				2.642

2) Data *Pre-processing*

Tahap awal pada perancangan model yaitu untuk mengolah data. Sebelum memulai untuk *training* data, proses pra-pemrosesan (*pre-processing*) data dilakukan di penelitian ini. Pra-pemrosesan pada data citra *CT-scan* merujuk pada formasi langkah yang digunakan untuk mengubah data dan mempersiapkan data sebelum dimasukkan ke dalam model. Data yang ada tentunya memiliki ukuran atau dimensi yang berbeda-beda. Tujuan dari *pre-processing* ini ialah untuk meningkatkan kualitas data, variasi data dan meningkatkan akurasi prediksi.

Proses *preprocessing* dilakukan pada penelitian tersebut ada 2 tahapan proses pengolahan data, yaitu sebagai berikut :

a. *Resizing* (Mengubah Ukuran)

Mengubah ukuran citra ke dimensi yang lebih kecil atau lebih besar. Tujuannya untuk memperbaiki kualitas citra dan meratakan ukuran atau dimensi pada citra, karena data citra dimensi atau ukurannya

berbeda-beda. *Resizing* pada data gambar di penelitian tersebut dilakukan dengan ukuran 224 x 224.

b. Augmentasi

Setelah *resizing*, kemudian langkah selanjutnya yaitu melakukan augmentasi pada citra. Tujuannya untuk mencegah *overfitting* dan meningkatkan generalisasi pada model. *Overfitting* terjadi ketika model yang dilatih terlalu cocok dengan training data hingga tingkat yang berlebihan. Hal ini menyebabkan model mengingat data pelatihan dengan sangat baik, namun performanya turun secara signifikan saat diuji pada data yang belum pernah dianalisis atau dilihat sebelumnya. Tanda-tanda *overfitting* mencakup tingkat akurasi yang tinggi pada data pelatihan, namun performa buruk yang dimiliki pada data *testing* atau validasi. Augmentasi pada data yang dilakukan dalam penelitian ini yaitu *rotation*, *zoom*, *horizontal flip*, dan *vertical flip*.

1) *Rescaling*

Rescaling adalah proses yang mengubah ukuran nilai piksel gambar. Nilai gambar dapat diubah menjadi rentang 0 hingga 1 dan nilai piksel dapat dibagi dengan 255. Pada penelitian ini proses *rescaling* gambar dilakukan dengan rentang 1./255.

Andaikan dalam sebuah piksel memiliki nilai matriks kemudian dilakukan *rescaling* dengan rentang 1./255 :

Nilai piksel gambar dibagi dengan 255 yaitu mengurangi dengan 1./255, yang berarti bahwa nilai piksel yang awalnya berada dalam rentang 0–255 diubah menjadi rentang 0–1.

$$\text{Rescaling} = \frac{\text{nilai piksel gambar}}{255}$$

Contoh, misalkan memiliki matriks gambar dengan nilai piksel beriku :

$$\text{Matriks awal} = \begin{bmatrix} 0 & 64 & 128 \\ 192 & 255 & 128 \end{bmatrix}$$

$$\text{Rescaling} = \frac{0}{255}$$

$$= \frac{192}{255}$$

$$= \frac{64}{255}$$

$$= \frac{255}{255}$$

$$= \frac{128}{255}$$

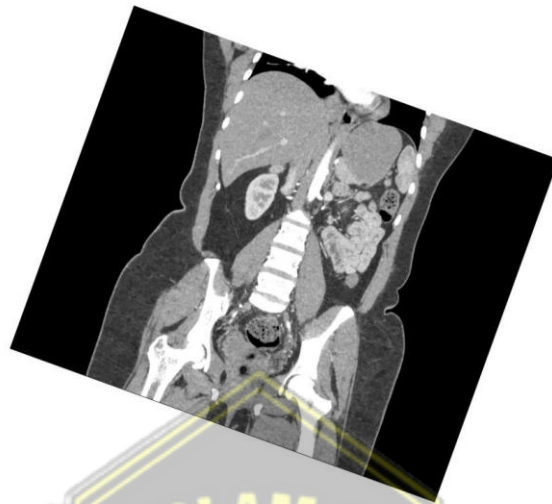
$$= \frac{128}{255}$$

$$\text{Matriks setelah rescaling} = \begin{bmatrix} 0 & 0,251 & 0,502 \\ 0,753 & 1 & 0,502 \end{bmatrix}$$

Fungsi *scaling* adalah untuk menormalkan nilai piksel sehingga semua nilai berada dalam rentang yang sama, yang membantu dalam stabilitas dan kecepatan konvergensi atau mempercepat proses pelatihan model. Fungsi aktivasi seperti *sigmoid* biasanya bekerja baik dan mampu memberikan hasil yang baik pula ketika digunakan pada data yang telah dinormalisasi.

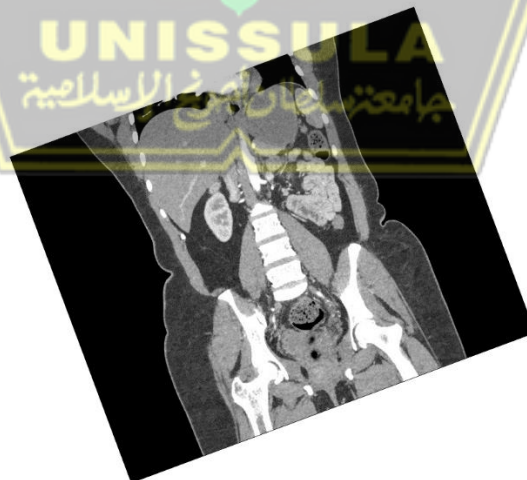
2) *Rotation*

Augmentasi rotasi membantu model memahami objek meskipun hal tersebut muncul dalam berbagai orientasi. Nilai parameter rotasi gambar dalam penelitian ini diputar secara acak hingga 20 derajat ke kiri atau kanan. Dengan menggunakan parameter rotasi ini yang membuat model lebih tahan terhadap rotasi gambar dan menambah variasi data tanpa harus menambah data baru.



Gambar 3. 4 Contoh Rotasi 20 Derajat Ke Kanan

Gambar 3.4 merupakan penggambaran penerapan parameter rotasi dengan nilai rotasi 20 derajat ke kanan. Sebagai gambaran dari proses augmentasi data dengan parameter rotasi yang dilakukan untuk menambah variasi data. Maka ketika ada data baru untuk dilakukan prediksi, model dapat generalisasi dengan seharusnya dan akurat.



Gambar 3. 5 Contoh Rotasi 20 Derajat Ke Kiri

Gambar 3.5 merupakan penggambaran penerapan parameter rotasi dengan nilai rotasi 20 derajat ke kanan. Sebagai gambaran dari proses augmentasi data dengan parameter rotasi yang dilakukan untuk menambah variasi data.

3) *Zoom*

Penerapan parameter *zoom* di mana variasi skala ditambahkan ke data pelatihan yang meningkatkan kemampuan model untuk mengenali objek dari jarak yang berbeda. Penggunaan *zoom* gambar juga membantu model menjadi lebih umum dan tidak bergantung pada ukuran objek yang tetap.

4) *Horizontal dan Vertical Flip*

Salah satu metode augmentasi data dalam pemrosesan gambar adalah *flip* (membalikkan) gambar secara vertikal atau horizontal. Metode ini tidak mengubah nilai piksel pada gambar input, tetapi hanya mengubah posisi piksel dalam gambar.

– Horizontal *Flip*

Horizontal *flip* yaitu membalik gambar secara horizontal. Piksel pada kolom paling kiri bertukar tempat dengan piksel pada kolom paling kanan, kolom kedua dari kiri dengan kolom kedua dari kanan, dan seterusnya.

Berikut gambaran perhitungannya :

Misalkan sebuah citra input memiliki matriks gambar 3x3

$$\text{Matriks} = \begin{bmatrix} 25 & 150 & 75 \\ 11 & 46 & 33 \\ 31 & 100 & 20 \end{bmatrix}$$

$$\text{Matriks Horizontal Flip} = \begin{bmatrix} 75 & 150 & 25 \\ 33 & 46 & 11 \\ 20 & 100 & 31 \end{bmatrix}$$

– *Vertical Flip*

Vertical flip yaitu membalik gambar secara vertikal. piksel pada baris paling atas bertukar dengan piksel pada baris paling bawah, piksel pada baris kedua dari atas bertukar dengan piksel pada baris kedua dari bawah, dan seterusnya.

Berikut gambaran perhitungannya:

Misalkan sebuah citra input memiliki matriks gambar 3x3

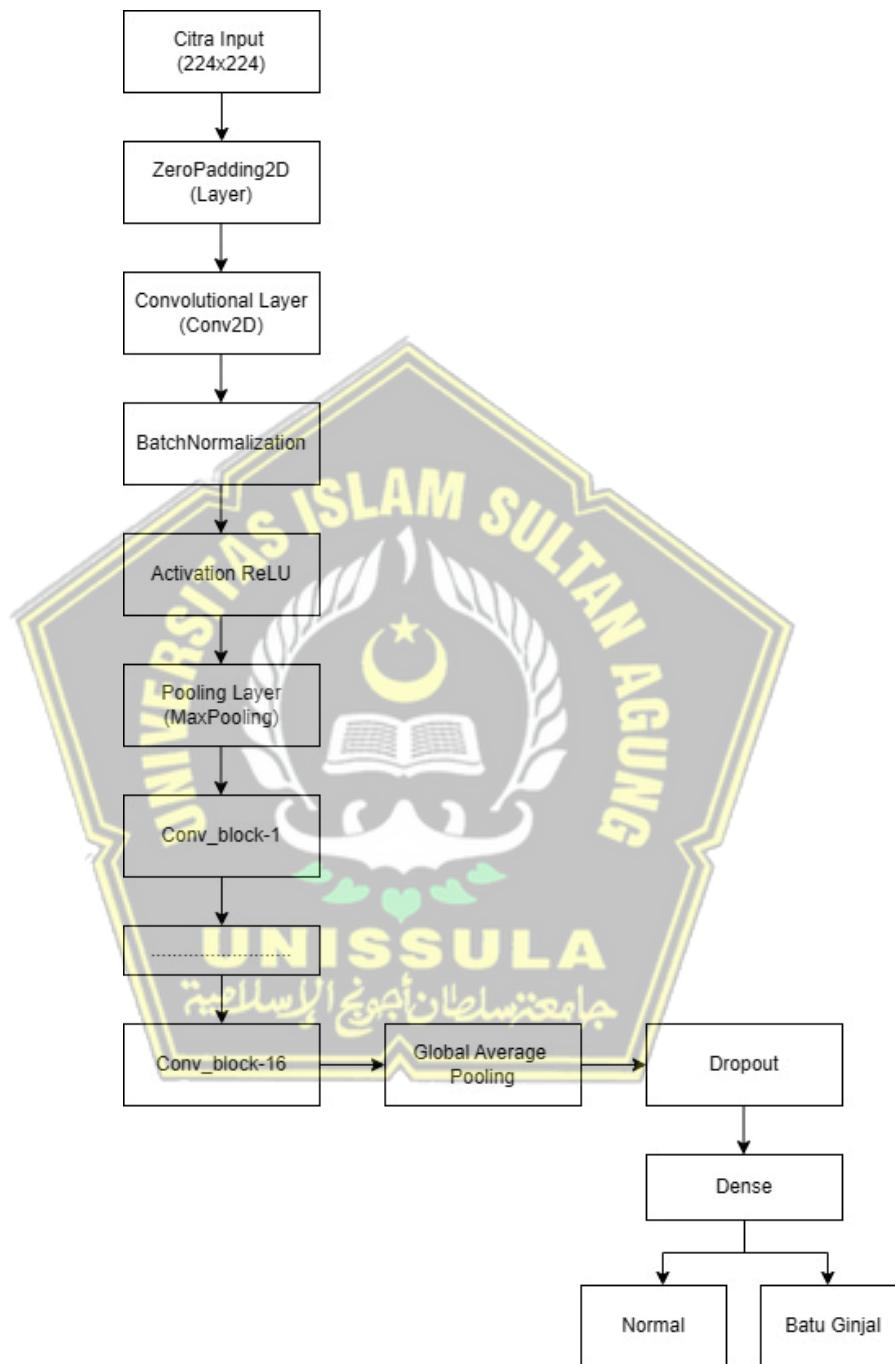
$$\text{Matriks} = \begin{bmatrix} 15 & 128 & 90 \\ 21 & 6 & 123 \\ 66 & 10 & 50 \end{bmatrix}$$

$$\text{Matriks } \textit{Vertical Flip} = \begin{bmatrix} 66 & 10 & 50 \\ 11 & 46 & 33 \\ 15 & 128 & 90 \end{bmatrix}$$

Untuk meningkatkan keragaman data pelatihan, kedua metode ini adalah metode augmentasi data yang sederhana tetapi efektif. Ini membantu model pembelajaran mesin menjadi lebih tahan terhadap variasi dalam data input.



3) Bangun CNN Model dengan Arsitektur DenseNet-121



Gambar 3. 6 Pre-Training Model

Pada Gambar 3.6 merupakan diagram *layer pre-training* model pada data.

a. *Convolutional Layer*

Lapisan Konvolusi adalah salah satu lapisan yang paling penting dalam CNN (*Convolutional Neural Network*) karena bertanggung jawab untuk mengekstraksi fitur-fitur penting dari data gambar. *Conv2D* digunakan untuk melakukan operasi konvolusi pada data *input*. Operasi konvolusi ini dilakukan dengan menggunakan *filter* atau kernel untuk memindai gambar dan mengekstraksi pola-pola visual yang relevan. Setiap kernel dalam lapisan konvolusi memiliki bobot yang dipelajari selama proses pelatihan model.

b. *GlobalAveragePooling*

Global Average Pooling (GAP) mengambil rata-rata dari semua nilai dalam setiap feature map. Dengan kata lain, *Global Average Pooling* menghasilkan satu nilai rata-rata yang menggambarkan semua informasi yang terkandung dalam setiap *feature map*. GAP menggantikan lapisan *fully connected* yang sering digunakan di akhir jaringan dengan operasi yang jauh lebih sederhana dan berdaya guna.

c. *Batch Normalization*

Dalam proses pelatihan jaringan saraf, *batch normalization* adalah metode normalisasi yang digunakan pada *mini-batch* (memperbarui bobot model). Proses ini menghitung rata-rata dan varians dari setiap *mini-batch*, lalu menggunakan nilai-nilai tersebut untuk menormalkan input dari setiap lapisan.

d. Aktivasi ReLU (*Rectified Linear Unit*)

Fungsi aktivasi yang membantu model mempelajari pola data yang kompleks dengan mendefinisikan *output* sebagai input itu sendiri jika *input*-nya positif dan nol jika inputnya negatif.

Matematika/Perhitungan ReLU:

$$f(x) = \max(0, x). \quad (1)$$

e. *Max Pooling*

Max pooling adalah metode untuk mengurangi ukuran gambar yang diproses tanpa kehilangan informasi penting. Ini dilakukan dengan mengambil nilai tertinggi dari sekelompok piksel dan hanya menyimpan nilai tertinggi saja. Hal ini membantu mengurangi jumlah perhitungan yang dilakukan dan mencegah *overfitting*.

f. *Dropout*

Cara untuk membuat model lebih tahan terhadap *overfitting* adalah dengan mematikan bagian jaringan secara acak selama pelatihan (*Dropout*). Ini membuat model tidak terlalu bergantung pada neuron tertentu dan belajar fitur yang lebih umum.

g. *Dense Layer*

Layer dense adalah bagian dari jaringan saraf di mana semua neuron terhubung dengan neuron di lapis sebelumnya, seperti jaringan lengkap yang menghubungkan setiap titik dengan setiap titik lainnya. Pada bagian akhir jaringan, lapis ini biasanya digunakan untuk menggabungkan semua informasi yang telah dipelajari dari lapis sebelumnya dan menghasilkan *output* akhir. Pada penelitian ini menggunakan ukuran neuron 512.

4) Pelatihan Model

Latih model menggunakan data pelatihan. Gunakan fungsi pada Keras dengan menentukan jumlah *epoch* atau literasi pelatihan, ukuran *batch*, dan set pelatihan yang telah dipersiapkan sebelumnya. Memantau kinerja model selama pelatihan dengan melihat metrik-metrik dan akurasi pada set pelatihan dan validasi. Pada penelitian ini menggunakan *batch size* 32. *Optimizer* yang digunakan untuk *compile* model yaitu Adam dengan nilai *learning rate* 0,0001. Untuk *training* model dalam penelitian ini menggunakan 35 *epoch* dengan menambahkan fungsi *early stopping*. Fungsi *early stopping* juga akan digunakan dalam proses *training* model

sebagai langkah antisipasi untuk mengurangi *overfitting* pada model dengan menerapkan nilai *patience* 5 dan *monitoring* pada validasi *loss*.

5) *Compile Model*

Tahap dimana mendefinisikan *loss function*, *optimizer*, dan *metrics* pada model. Kemudian melatih model dengan *training set*, memonitor *loss* dan *accuracy* pada *validation set*. *Compile* model dengan *optimizer* Adam dengan *learning rate* 0,0001.

6) *Evaluasi Model*

Pada penelitian ini melakukan evaluasi terhadap set validasi dengan data yang ada bisa dilihat pada Tabel 3.1. Setelah pelatihan selesai dilakukan, evaluasi model pada set validasi untuk mengukur kemampuan model terhadap data yang tidak digunakan semasa pelatihan. Untuk menjalankan evaluasi model menggunakan metrik akurasi, *precision*, *recall*, dan *f1-score* perlu menghitung jumlah prediksi yang benar dari model terhadap total jumlah data pengujian. Evaluasi model juga dilakukan dengan membandingkan nilai hasil dari *accuracy* dan *loss* pada set *training* dan validasi.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \times 100\% \quad (1)$$

$$Precision = \frac{TP}{TP + FP} \times 100\% \quad (2)$$

$$Recall = \frac{TP}{TP + FN} \times 100\% \quad (3)$$

$$F1-Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \times 100\% \quad (4)$$

Keterangan:

1. TP (*True Positive*) adalah jumlah data positif yang berhasil diklasifikasikan dengan benar oleh model.

2. TN (*True Negative*) adalah jumlah data negatif yang berhasil diklasifikasikan dengan benar oleh model.
3. FP (*False Positive*) adalah jumlah data positif yang gagal diklasifikasikan dengan benar oleh model.
4. FN (*False Negative*) adalah jumlah data negatif yang gagal diklasifikasikan dengan benar oleh model.

7) *Model Deployment*

Streamlit adalah *framework* yang dapat memudahkan untuk membuat aplikasi berbasis *web* secara interaktif dan cepat terutama dalam hal untuk aplikasi *machine learning*. Dalam penggunaan *streamlit*, *script python* dapat diubah menjadi aplikasi berbasis *web* tanpa perlu keahlian dalam pengembangan *web* secara kompleks.

Setelah serangkaian model dibangun, akhir dari evaluasi model yaitu *deploy* model dengan *deployment local (server local)*. Model ini akan digunakan untuk melakukan klasifikasi batu ginjal pada data *input*. *Deploy* model yang dilakukan menggunakan *framework python* yaitu *streamlit*. Sebelum *deployment*, buat kerangka *web* terlebih dahulu dengan *streamlit* untuk ditampilkan, kemudian setelah kerangka dibentuk *save* kerangka dengan nama *app.py*. Menampilkan *app.py* dengan *code* `!wget -q -o - ipv4.icanhazip.com` untuk mendapatkan *index* untuk dimasukkan ke *localtunnel*. Memanggil *app.py* yang sudah tersimpan sebelumnya, lalu *running index* dengan *localtunnel*.

BAB IV

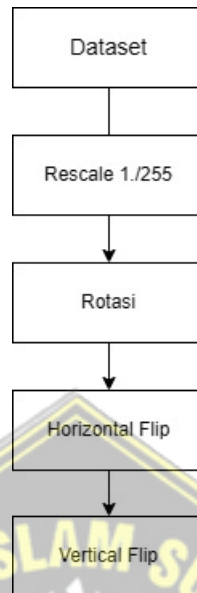
HASIL DAN ANALISIS PENELITIAN

4.1 Hasil

Mengacu Gambar 3.3 alur perancangan sistem dengan beberapa tahapan untuk dapat menghasilkan model yang akan digunakan untuk klasifikasi penyakit batu ginjal. Hasil dan analisis dalam penelitian ini mencakup beberapa aspek, termasuk penerapan teknik augmentasi data, penggunaan fungsi dropout untuk mengurangi *overfitting*, dan penerapan fungsi *early stopping* untuk menghentikan pelatihan model sebelum mencapai *overfitting*. Analisis *confusion matrix* untuk mengevaluasi kinerja model, serta proses *load* model menggunakan *Streamlit*. Penelitian ini melibatkan berbagai percobaan untuk menguji efektivitas dan performa dari metode-metode tersebut.

4.1.1 Diperlukan Augmentasi Data

Dalam *machine learning*, augmentasi data adalah cara untuk meningkatkan jumlah dan keragaman data pelatihan tanpa mengumpulkan data baru. Kemudian proses ini menghasilkan variasi baru dengan menerapkan transformasi tertentu pada data yang sudah ada. Parameter yang diterapkan untuk model meliputi rotasi gambar, *flip* horizontal atau vertikal, perubahan skala, dan *zoom* adalah beberapa contoh augmentasi data.



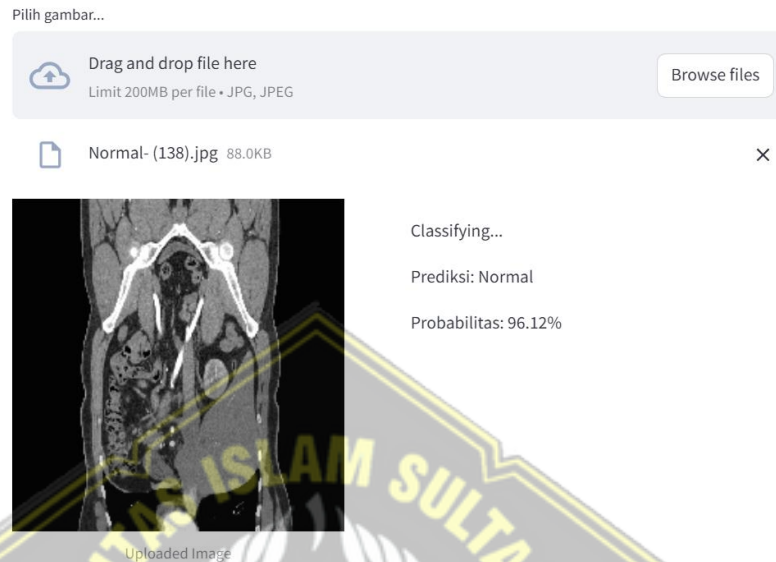
Gambar 4. 1 Parameter Augmentasi Data

Augmentasi data memungkinkan model untuk lebih umum dengan memberikan lebih banyak variasi dalam data pelatihan. Dalam hal ini juga dapat membantu mengurangi adanya *overfitting* yang di mana ketika model terlalu fit atau cocok dengan data pelatihan dan tidak dapat menggeneralisasi terhadap data baru. Proses augmentasi data melibatkan menerapkan berbagai transformasi pada kumpulan data yang sudah ada untuk menghasilkan bentuk data baru. Pada Gambar 4.1 menunjukkan beberapa transformasi yang digunakan dalam penelitian ini dimulai dari perubahan skala sampai dengan *flip* vertikal.

Rotation range dengan nilai 20 derajat gambar diputar secara random atau acak ke kiri atau ke kanan. Kemudian vertikal dan horizontal *flip* sebagai fungsi untuk membalik gambar secara horizontal maupun vertikal sesuai dengan ketentuan yang dibutuhkan. Target *size* atau *resize* data diset dengan ukuran 224 x 224 dengan *batch size* 32 untuk data *training* dan validasi. Pada Gambar 4.2 merupakan diagram *preprocessing* data.

Berikut analisa mengenai implementasi dari augmentasi data dan tanpa augmentasi dengan *testing* dari hasil model :

Klasifikasi Penyakit Batu Ginjal



Gambar 4. 2 Eksperimen Menggunakan Augmentasi

Pada Gambar 4.2 merupakan contoh hasil dari implementasi augmentasi data. Percobaan dilakukan dengan data *testing vertical flip*. Dengan setting awal data augmentasi maka ragam atau variasi data testing mampu mendeteksi dan mengkonfigurasi data yang ada dengan akurat.

Klasifikasi Penyakit Batu Ginjal



Gambar 4. 3 Eksperimen Tanpa Augmentasi

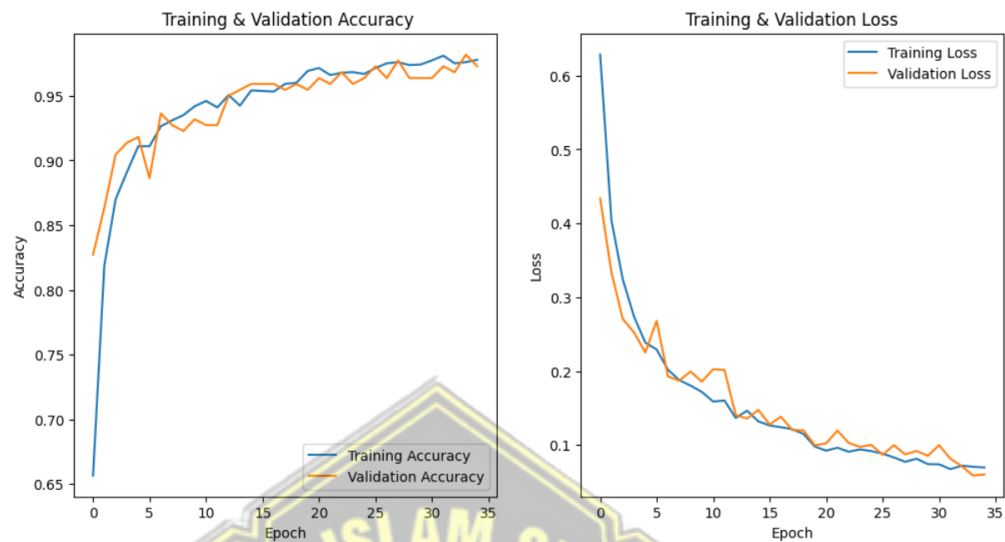
Pada Gambar 4.3 merupakan contoh hasil dari percobaan implementasi model tanpa augmentasi data. Percobaan yang dilakukan menggunakan data dengan kategori *vertical flip*. Hasil menunjukkan adanya *overfitting* pada model dengan data baru yang bervariasi. Hal ini membuat hasil testing dengan data baru yang bervariasi menjadi tidak akurat.

4.1.2 Penggunaan *Early Stopping*

Pada dasarnya *early stopping* digunakan untuk menghentikan pelatihan model ketika kinerja model di atas data validasi tidak lagi meningkat atau bahkan menurun, meskipun kinerja terhadap data pelatihan terus melonjak. Ini memperlihatkan bahwa model telah mulai menempatkan diri dengan data pelatihan, yang dapat mengakibatkan penurunan kinerja mengacu data baru atau data yang belum pernah dilihat sebelumnya.

Fungsi *early stopping* yaitu membantu mencegah model dari *overfitting*, di mana kondisi model terlalu fit atau menghafal data pelatihan sehingga tidak bisa melakukan generalisasi dengan baik pada data baru. Dengan menghentikan pelatihan lebih awal ketika tidak ada lagi peningkatan dalam performa pada data validasi atau terjadi peningkatan pada *validation loss*. *Early stopping* dapat membantu waktu yang diperlukan untuk pelatihan model dan sumber daya komputasi yang diperlukan.

Selama proses pelatihan, matriks tertentu digunakan untuk mengevaluasi kinerja model. Untuk menilai kinerja generalisasi model, *monitoring* dilakukan dengan matriks *validation loss* bagian set validasi. Parameter yang dikenal sebagai *patience* biasanya diperlukan untuk penundaan dini, yang menentukan waktu yang diperbolehkan untuk tidak ada peningkatan dalam kinerja sebelum pelatihan dihentikan. Pada penelitian ini, tingkat *patience* diatur menjadi 5 *patience*. Jika tidak ada peningkatan dalam kinerja atau penurunan nilai *validation loss* dalam beberapa *epoch*, pelatihan akan dihentikan setelah lima *epoch* tanpa peningkatan dalam performa *training* model.



Gambar 4. 4 Pelatihan Berhenti Karena Adanya Peningkatan Nilai Validasi Loss

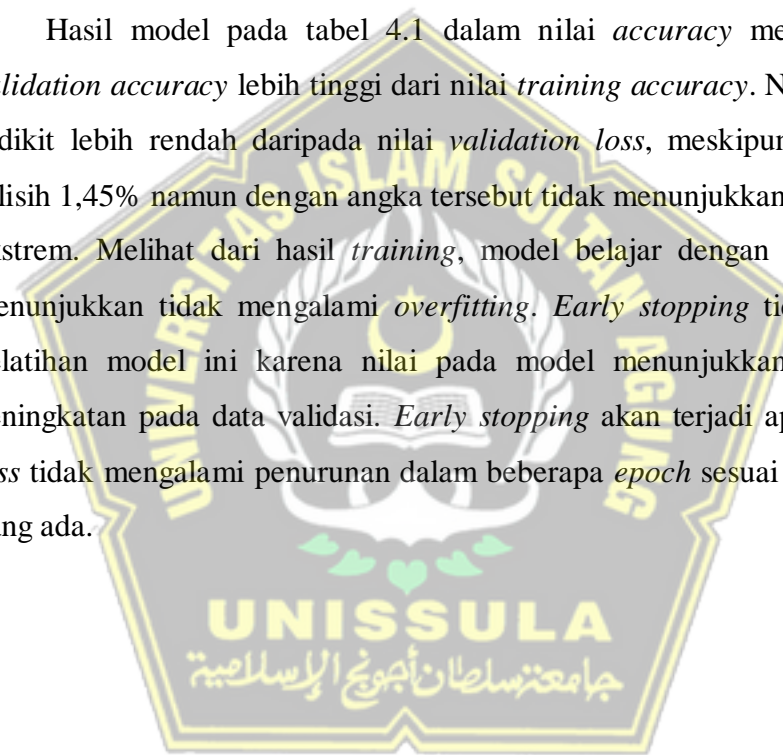
Pada Gambar 4.4 menunjukkan hasil dari *training* model meliputi *accuracy*, *loss*, validasi *loss*, dan validasi akurasi. Dari hasil *training* model tersebut fokus pada nilai *validation loss*, yang di mana nilai *validation loss* dijadikan variabel monitor untuk *early stopping*. Pada *validation loss training epoch* 1 hingga *epoch* 8 menunjukkan adanya penurunan pada nilai *training loss* dan validasi *loss*. *Epoch* 1 hingga 8 peningkatan *training* dan validasi mengalami peningkatan akurasi. Nilai pada *epoch* 8 hingga 10 *training loss* dan validasi *loss* terus menurun dan begitu juga untuk akurasi pelatihan dan akurasi validasi pelatihan terus meningkat secara bertahap. Pada *epoch* 11 hingga 25 *training loss* dan *validation loss* terus mengalami penurunan meskipun ada lonjakan nilai di beberapa *epoch* tetapi setelahnya nilai *loss* cenderung turun. Kemudian *training accuracy* dan *validation accuracy* terus membaik dan meningkat meskipun ada sedikit penurunan di beberapa *epoch* akan tetapi nilai akurasinya cenderung terus meningkat. Pada *epoch* 26 hingga 35 *training accuracy* dan *training loss* terus membaik dan meningkat. Pada nilai *validation loss* dan *validation accuracy* menunjukkan penurun *loss* dan peningkatan *accuracy*. *Epoch* 35 *epoch* terakhir

validation loss turun mencapai nilai 0,0864 dan *validation accuracy* meningkat mencapai nilai 0,9818.

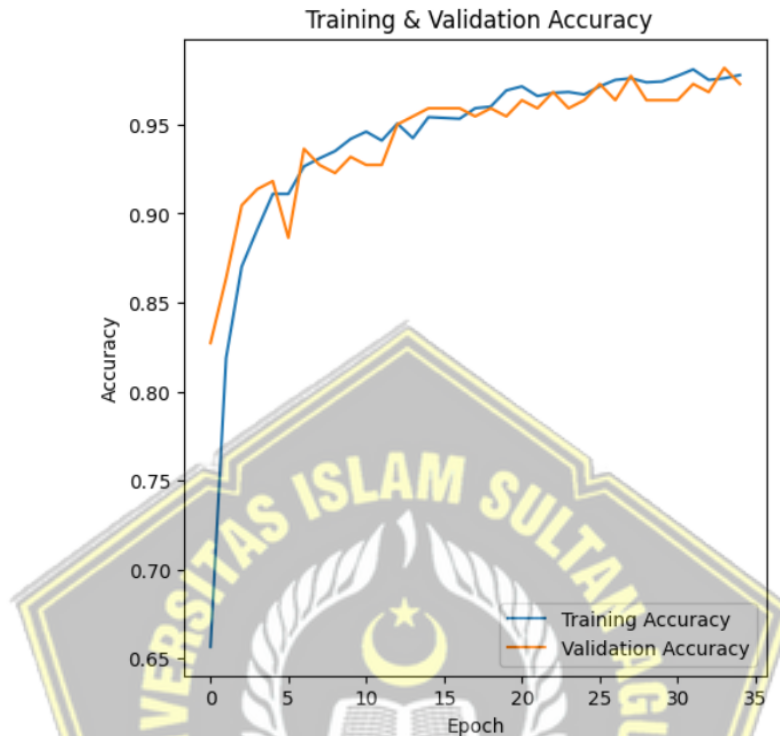
Tabel 4. 1 Hasil *Training Model*

	<i>Training</i>	<i>Validation</i>
<i>Accuracy</i>	97,73%	98,18%
<i>Loss</i>	7,19%	8,64%

Hasil model pada tabel 4.1 dalam nilai *accuracy* menunjukkan nilai *validation accuracy* lebih tinggi dari nilai *training accuracy*. Nilai *training loss* sedikit lebih rendah daripada nilai *validation loss*, meskipun ada perbedaan selisih 1,45% namun dengan angka tersebut tidak menunjukkan perbedaan yang ekstrem. Melihat dari hasil *training*, model belajar dengan baik dan model menunjukkan tidak mengalami *overfitting*. *Early stopping* tidak terjadi pada pelatihan model ini karena nilai pada model menunjukkan perbaikan dan peningkatan pada data validasi. *Early stopping* akan terjadi apabila *validation loss* tidak mengalami penurunan dalam beberapa *epoch* sesuai dengan *patience* yang ada.

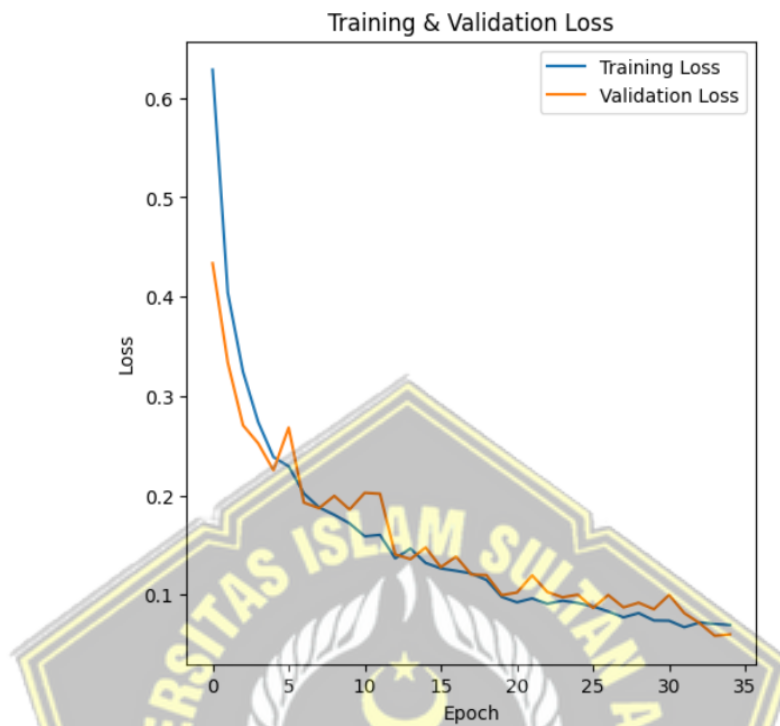


4.1.3 Training Model



Gambar 4.5 Grafik Akurasi *Training* dan Validasi

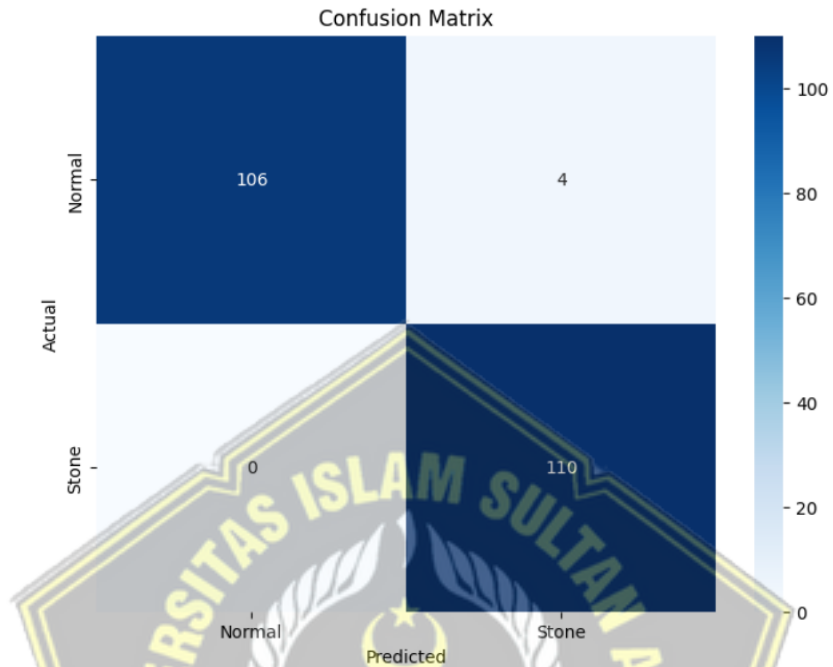
Pada Gambar 4.5 merupakan hasil akurasi dari *training* dan validasi model. Garis dengan warna biru menunjukkan hasil *training* model, sedangkan garis dengan warna *orange* menunjukkan hasil validasi model. Pada grafik tersebut masing-masing pelatihan *training* dan validasi model mengalami peningkatan dilihat dari nilai *accuracy* setiap *epoch*, meskipun ada beberapa *epoch* yang mengalami sedikit penurunan namun tidak signifikan dan cenderung peningkatan terjadi kembali.



Gambar 4. 6 Grafik *Loss Training* Model dan Validasi Model

Pada Gambar 4.6 merupakan grafik hasil *loss* dari *training* dan validasi model. Garis dengan warna biru menunjukkan hasil *loss training* model, sedangkan garis dengan warna *orange* menunjukkan hasil *loss* validasi model. Pada grafik tersebut masing-masing pelatihan *training* dan validasi model mengalami penurunan dilihat dari nilai *loss* setiap *epoch*, meskipun ada beberapa *epoch* yang mengalami sedikit peningkatan namun tidak signifikan dan cenderung peningkatan terjadi kembali. Peningkatan nilai *loss* secara teratur dan signifikan.

4.1.4 Confusion Matrix



Gambar 4. 7 Confusion Matrix

Gambar 4.7 menunjukkan hasil *confusion matrix* dengan model yang sudah ada menggunakan data *testing*. Jumlah *true positive* 106 (normal) dan jumlah *true negative* 110 (batu ginjal). Nilai *false positive* 4 dan *false negative* bernilai 0.

Tabel 4. 2 Perhitungan *Confusion Matrix*

<i>Confusion Matrix</i>	Perhitungan	Hasil
<i>Accuracy</i>	$\frac{106 + 110}{106 + 110 + 4 + 0} \times 100\%$	98,18%
<i>Precision</i>	$\frac{106}{106 + 4} \times 100\%$	96,36%
<i>Recall</i>	$\frac{106}{106 + 0} \times 100\%$	100%
<i>F1-Score</i>	$2 \times \frac{96,36 \times 100}{96,36 + 100} \times 100\%$	98,14%

Hasil *accuracy* sebesar 98,18%, hasil *precision* 96,36%, hasil *recall* 100% dan hasil *F1-score* 98,14%.

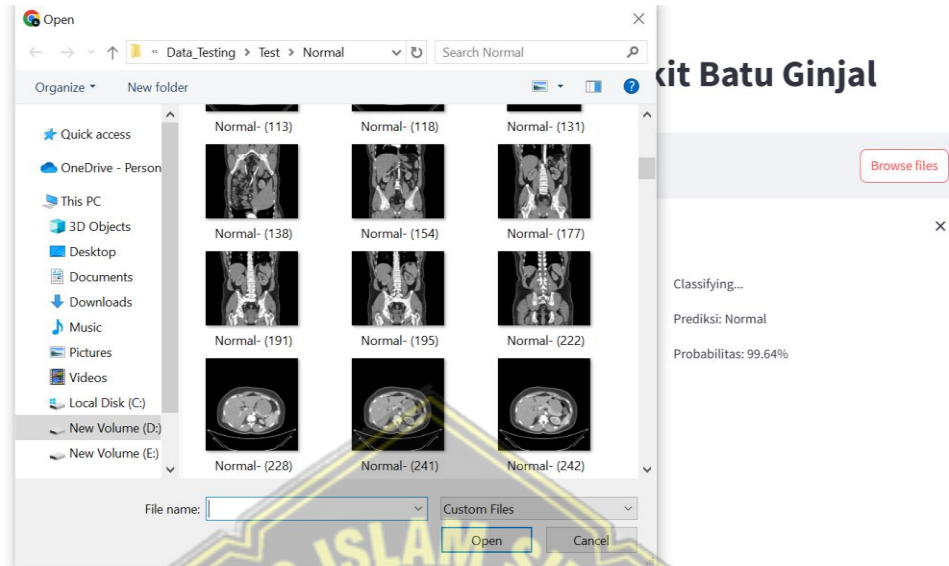
4.1.5 Running App

- 1) *Dashboard*



Gambar 4. 8 *Dashboard*

Menampilkan halaman sistem untuk klasifikasi penyakit batu ginjal menggunakan *streamlit*. *Button browse files* untuk *upload file* di penyimpanan dokumen.



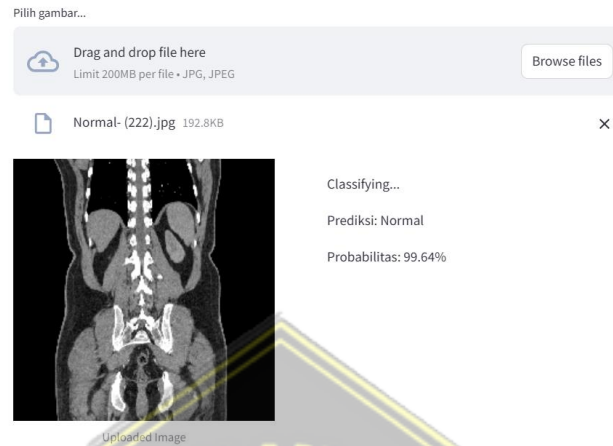
Gambar 4. 9 *Browse Files*

Pada Gambar 4.9 dapat *upload file* yang akan digunakan untuk prediksi.

2) **Prediksi**

Langkah selanjutnya yaitu prediksi. Setelah melakukan *upload file* di penyimpanan dokumen, sistem *running* dan menampilkan hasil prediksi penyakit batu ginjal dan ginjal normal. Serta menampilkan nilai probabilitas dari klasifikasi.

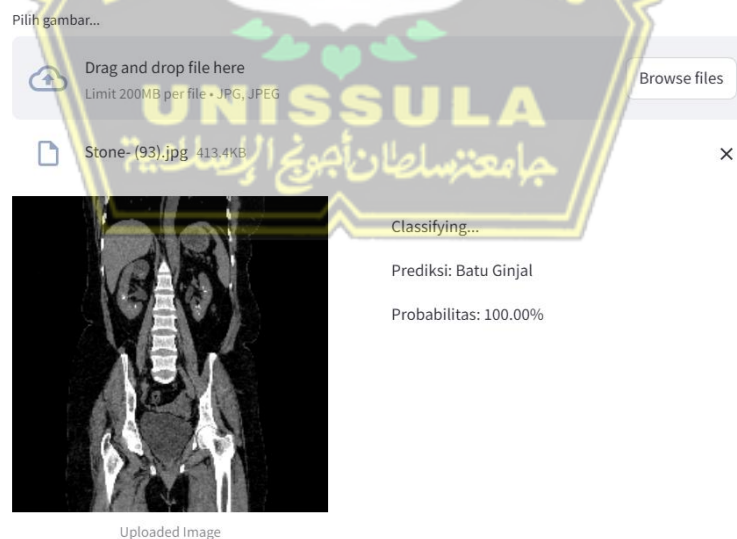
Klasifikasi Penyakit Batu Ginjal



Gambar 4. 10 Klasifikasi Normal

Pada Gambar 4.10 menunjukkan prediksi yaitu ginjal normal. Implementasi *testing* tersebut menggunakan data kondisi ginjal normal. Dari hasil tersebut klasifikasi penyakit batu ginjal model mampu generalisasi terhadap data *testing* dengan baik.

Klasifikasi Penyakit Batu Ginjal



Gambar 4. 11 Hasil Klasifikasi Batu Ginjal

Pada Gambar 4.11 menunjukkan hasil prediksi yaitu ginjal dengan batu ginjal. Dari hasil tersebut klasifikasi penyakit batu ginjal model mampu generalisasi terhadap data *testing* dengan baik.

4.2 Analisis

4.2.1 Experiment

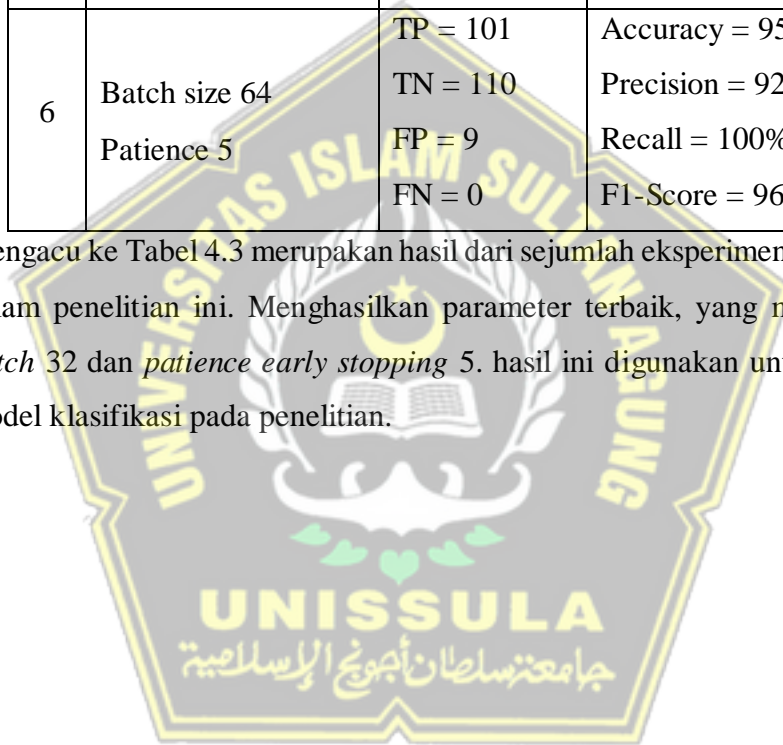
Dalam penelitian ini, beberapa percobaan dilakukan untuk melakukan *hyperparameter tuning* guna memperoleh parameter terbaik dalam membangun performa model. Percobaan tersebut melibatkan pengaturan *batch size* dengan nilai 16, 32, dan 64, serta pengaturan *patience* pada *early stopping* dengan nilai 3 dan 5. Setelah parameter diatur, performa model dievaluasi menggunakan *confusion matrix* untuk menilai hasil akhir.

Tabel 4. 3 *Hyperparameter Tuning* Hasil Dari Beberapa *experiment*

NO	Parameter	Confusion Matrix	
1.	Batch size 16 Patience 3	TP = 105 TN = 109 FP = 5 FN = 1	Accuracy = 97,27% Precision = 95,61% Recall = 99,09% F1-Score = 97,32%
2	Batch size 32 Patience 3	TP = 105 TN = 104 FP = 5 FN = 6	Accuracy = 95% Precision = 95,41% Recall = 94,54% F1-Score = 94,97%
3	Batch size 64 Patience 3	TP = 103 TN = 108 FP = 7 FN = 2	Accuracy = 95,90% Precision = 93,91% Recall = 98,18% F1-Score = 96%
4	Batch size 16 Patience 5	TP = 105 TN = 110	Accuracy = 97,72% Precision = 95,65%

NO	Parameter	Confusion Matrix	
		FP = 5 FN = 0	Recall = 100% <i>F1-Score</i> = 97,77%
5	Batch size 32 Patience 5	TP = 106 TN = 110 FP = 4 FN = 0	Accuracy = 98,18% Precision = 96,49% Recall = 100% <i>F1-Score</i> = 98,21%
6	Batch size 64 Patience 5	TP = 101 TN = 110 FP = 9 FN = 0	Accuracy = 95,90% Precision = 92,43% Recall = 100% <i>F1-Score</i> = 96,06%

Mengacu ke Tabel 4.3 merupakan hasil dari sejumlah eksperimen yang dilakukan dalam penelitian ini. Menghasilkan parameter terbaik, yang memiliki ukuran *batch* 32 dan *patience early stopping* 5. hasil ini digunakan untuk membangun model klasifikasi pada penelitian.



BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil dan analisis dalam penelitian ini, dapat disimpulkan bahwa model Convolutional Neural Network (CNN) yang dibangun dengan arsitektur DenseNet-121 menunjukkan performa yang sangat baik dalam mengklasifikasikan penyakit batu ginjal. *DenseNet-121* memiliki keunggulan dalam penggunaan kembali fitur (*feature reuse*) yang memungkinkan model untuk belajar representasi fitur yang lebih luas. Arsitektur ini juga memecahkan masalah *vanishing gradient* yang kerap terjadi pada jaringan yang sangat dalam. Untuk meningkatkan kinerja model, teknik pra-pemrosesan data seperti augmentasi gambar sangat penting. Meningkatkan keragaman dataset dan mencegah *overfitting* dibantu oleh augmentasi gambar.

Dari penelitian ini mendapatkan hasil model yang baik dengan *accuracy train* dan *accuracy validation* model yang dapat generalisasi dengan baik. Hasil *confusion matrix* pada penelitian ini meliputi hasil *accuracy* sebesar 98,18%, hasil *precision* 96,36%, hasil *recall* 100%, dan hasil *f1-score* 98,14%. Hal demikian, metode CNN dengan arsitektur *DenseNet-121* terbukti efektif dalam klasifikasi penyakit batu ginjal dan ini membuat kemajuan besar dalam domain diagnosa medis berbasis kecerdasan buatan. *Deployment* dapat dilakukan menggunakan *streamlit* untuk memudahkan *user* atau dokter dapat melakukan klasifikasi dengan mudah.

5.2 Saran

1. Penelitian selanjutnya dapat membuat *GUI* yang mudah untuk digunakan dan menarik atau bisa dibuat lebih lanjut dengan tampilan aplikasi berbasis android.

2. Penelitian selanjutnya dapat pengembangan lebih lanjut untuk *deploy* aplikasi *web* dengan *hosting* atau *cloud*.



DAFTAR PUSTAKA

- Ahmed, F. *dkk.* (2024) “Identification of kidney stones in KUB X-ray images using VGG16 empowered with explainable artificial intelligence,” *Scientific Reports*, 14(1), hal. 1–14. doi: 10.1038/s41598-024-56478-4.
- Aziz, F., Wahab, A. dan Muna, P. (2023) “Journal Pharmacy and Application of Computer Sciences Penerapan Metode Machine Learning Untuk Mengklasifikasi Penanganan Perawatan Pasien Application of Machine Learning Method To Classification of Patient Care Treatment,” *Journal Pharmacy and Application of Computer Sciences*, 1(2), hal. 57–62. Tersedia pada: <https://journalanr.arlisakamadani.com/index.php/jopacs/article/view/37/32>.
- Bano, S. *dkk.* (2022) “Diagnosis of Urinary Tract Urolithiasis using Computed Tomography,” *Pakistan Journal of Health Sciences*, 03(c), hal. 03–06. doi: 10.54393/pjhs.v3i01.46.
- Cahya, F. N. *dkk.* (2021) “Klasifikasi Penyakit Mata Menggunakan Convolutional Neural Network (CNN),” *Sistemasi*, 10(3), hal. 618. doi: 10.32520/stmsi.v10i3.1248.
- Chipindu, L. *dkk.* (2020) “Maize Kernel Abortion Recognition and Classification Using Binary Classification Machine Learning Algorithms and Deep Convolutional Neural Networks,” *AI (Switzerland)*, 1(3), hal. 361–375. doi: 10.3390/ai1030024.
- Ferraro, P. M. *dkk.* (2020) “Re: Risk of Kidney Stones: Influence of Dietary Factors, Dietary Patterns, and Vegetarian-Vegan Diets,” *Journal of Urology*, 204(3), hal. 612. doi: 10.1097/JU.0000000000001173.01.
- Gulavnai, S. dan Patil, R. (2019) “Deep Learning for Image Based Mango Leaf Disease Detection,” *International Journal of Recent Technology and Engineering*, 8(3S3), hal. 54–56. doi: 10.35940/ijrte.c1030.1183s319.
- Gunawan, D. dan Setiawan, H. (2022) “Convolutional Neural Network dalam Analisis Citra Medis,” *KONSTELASI: Konvergensi Teknologi dan Sistem Informasi*, 2(2),

- hal. 376–390. doi: 10.24002/konstelasi.v2i2.5367.
- Hadibrata, E. dan Suharmanto (2022) “Faktor-Faktor Yang Berhubungan Dengan Terjadinya Batu Ginjal,” *Jurnal Penelitian Perawat Profesional*, 4(3), hal. 1041–1046. Tersedia pada: <http://jurnal.globalhealthsciencegroup.com/index.php/JPPP>.
- Handayani, N. M. S. (2020) “Analisis Kadar Kalsium Oksalat Pada Batu Ginjal,” *International Journal of Applied Chemistry Research*, 2(1), hal. 23. doi: 10.23887/ijacr.v2i1.28723.
- Harahap, F. A. A. dkk. (2022) “Implementasi Algoritma Convolutional Neural Network Untuk Mendeteksi Penyakit Ginjal Implementation of Convolutional Neural Network for Detecting Kidney Disease,” *Teknologi Informasi, Komputer dan Aplikasinya (JTika)*, 4(2), hal. 212–219. Tersedia pada: <http://jtika.if.unram.ac.id/index.php/JTIKA/>.
- Hasby Bik, A., Tri Anggraeny, F. dan Yulia Puspaningrum, E. (2024) “Klasifikasi Penyakit Ginjal Menggunakan Algoritma Hibrida Cnn-Elm,” *JATI (Jurnal Mahasiswa Teknik Informatika)*, 8(3), hal. 3836–3844. doi: 10.36040/jati.v8i3.9807.
- Khesya, N. (2021) “Mengenal Flowchart Dan Pseudocode Dalam Algoritma Dan Pemrograman,” *Definitions*. doi: <https://doi.org/10.31219/osf.io/dq45e>.
- Khultsum, U. dan Taufik, G. (2023) “Komparasi Kinerja DenseNet 121 dan MobileNet untuk Klasifikasi Citra Penyakit Daun Kentang,” *JURIKOM (Jurnal Riset Komputer)*, 10(2), hal. 558. doi: 10.30865/jurikom.v10i2.6047.
- Omori, Y. (2020) “Image Augmentation for Eye Contact Detection Based on Combination of Pre-trained Alex-Net CNN and SVM,” *Journal of Computers*, 15(3), hal. 85–97. doi: 10.17706/jcp.15.3.85-97.
- Putra, A. S. J., Subroto, I. M. I. dan Poetro, B. S. W. (2023) “Identifikasi Kematangan Buah Jeruk Medan Menggunakan K-Nearest Neighbor berbasis Metrik RGB,” *Jurnal Transistor Elektro dan Informatika (TRANSISTOR EI)*, 5(3), hal. 50112. Tersedia pada: <http://jurnal.unissula.ac.id/online/index.php/EI>.

- Rahayu, P. *dkk.* (2024) *Buku Ajar Data Mining*.
- Rahmawati, D. *dkk.* (2024) “Teknik Pemeriksaan CT Scan Abdomen Pada Kasus Tumor Intra Abdomen Di Instalasi Radiologi Rumah Sakit X,” *Jurnal Ilmu Kesehatan dan Gizi(JIG)*, 2(1), hal. 22–40. Tersedia pada: <https://doi.org/10.55606/jikg.v2i1.2093>.
- Ramadhani, R. *dkk.* (2023) “Analisis Prosedur Pemeriksaan CT Scan Abdomen Tiga Fase Pada Klinis,” *Journal of Educational Innovation and Public Health*, 1(4), hal. 275–283. Tersedia pada: <https://doi.org/10.55606/innovation.v1i4.1984>.
- Rizky Octavia, A. *dkk.* (2024) “Analisis Pemeriksaan CT Scan Abdomen Dengan Kontras Pada Klinis Ileus Obstruktif di Rumah Sakit Bhayangkara Makassar,” *Journal of Educational Innovation and Public Health*, 2(1), hal. 123–135. Tersedia pada: <https://doi.org/10.55606/innovation.v2i1.2104>.
- Saputra, D. (2024) “Upaya Pendidikan Menggunakan Machine Learning,” *Jurnal Riset Sistem Informasi*, 1(1), hal. 57–62. Tersedia pada: <https://doi.org/xx.xxxx/xxxxxxx.xxxxxx>.
- Satwikayana, S., Adi Wibowo, S. dan Vendyansyah, N. (2021) “Sistem Presensi Mahasiswa Otomatis Pada Zoom Meeting Menggunakan Face Recognition Dengan Metode Convolutional Neural Network Berbasis Web,” *JATI (Jurnal Mahasiswa Teknik Informatika)*, 5(2), hal. 785–793. doi: 10.36040/jati.v5i2.3762.
- Sulaksono, N. *dkk.* (2020) “Jurnal Imejing Diagnostik,” *Jurnal Imejing Diagnostik (JImeD)*, 6, hal. 103. Tersedia pada: <http://ejournal.poltekkes-smg.ac.id/ojs/index.php/jimed/index>.
- Ullah, M. Z. *dkk.* (2021) “An attention-based convolutional neural network for acute lymphoblastic leukemia classification,” *Applied Sciences (Switzerland)*, 11(22). doi: 10.3390/app112210662.
- Utami, W. B. *dkk.* (2020) “Penyuluhan 2G (Cegah Batu Ginjal dan Sayangi Ginjal Dengan Pola Hidup Sehat) Masyarakat RW VI Kelurahan Semanggi Kecamatan Pasar Kliwon,” 3, hal. 596–602.

- Vellaichamy, A. S. *dkk.* (2021) “Multiple Plant Leaf Disease Classification Using Densenet-121 Architecture,” *International Journal of Electrical Engineering and Technology*, 12(5), hal. 38–57. doi: 10.34218/ijeet.12.5.2021.005.
- Yohannes, R. dan Al Rivan, M. E. (2022) “Klasifikasi Jenis Kanker Kulit Menggunakan CNN-SVM,” *Jurnal Algoritme*, 2(2), hal. 133–144. doi: 10.35957/algoritme.v2i2.2363.

