

**SISTEM DIAGNOSIS OSTEOARTHRITIS LUTUT BERDASARKAN CITRA
X-RAY MENGGUNAKAN METODE CNN JENIS RESNET**

SKRIPSI



**UIN SUNAN AMPEL
S U R A B A Y A**

Disusun Oleh
NUR HIDAYAH
H92219055

**PROGRAM STUDI MATEMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI SUNAN AMPEL
SURABAYA**

2023

PERNYATAAN KEASLIAN

Saya yang bertanda tangan di bawah ini,

Nama : NUR HIDAYAH

NIM : H92219055

Program Studi : Matematika

Angkatan : 2019

Menyatakan bahwa saya tidak melakukan plagiat dalam penulisan skripsi saya yang berjudul "SISTEM DIAGNOSIS OSTEOARTHRITIS LUTUT BERDASARKAN CITRA X-RAY MENGGUNAKAN METODE CNN JENIS RESNET". Apabila suatu saat nanti terbukti saya melakukan tindakan plagiat, maka saya bersedia menerima sanksi yang telah ditetapkan.

Demikian pernyataan keaslian ini saya buat dengan sebenar-benarnya.

Surabaya, 11 Januari 2023

Yang menyatakan,



NUR HIDAYAH
NIM. H92219055

LEMBAR PERSETUJUAN PEMBIMBING

Prproposal skripsi oleh

Nama : NUR HIDAYAH
NIM : H92219055
Judul skripsi : SISTEM DIAGNOSIS OSTEOARTHRITIS LUTUT
BERDASARKAN CITRA X-RAY MENGGUNAKAN
METODE CNN JENIS RESNET


telah diperiksa dan disetujui untuk diujikan.

Pembimbing I



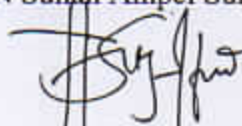
Dian Candra Rin Novitasari, M. Kom
NIP. 198511242014032001

Pembimbing II



Aris Fanani, M. Kom
NIP. 198701272014031002

Mengetahui,
Ketua Program Studi Matematika
UIN Sunan Ampel Surabaya



Yuniar Farida, M. T
NIP. 197905272014032002

PENGESAHAN TIM PENGUJI SKRIPSI

Skripsi oleh

Nama : NUR HIDAYAH
NIM : H92219055
Judul Skripsi : SISTEM DIAGNOSIS OSTEOARTHRITIS LUTUT
BERDASARKAN CITRA X-RAY MENGGUNAKAN
METODE CNN JENIS RESNET

Telah dipertahankan di depan Tim Penguji
pada tanggal 13 Januari 2023

Mengesahkan,
Tim Penguji

Penguji I

Yuniar Farida, M. T
NIP. 197905272014032002

Penguji II

Lutfi Hakim, M. Ag
NIP. 197312252006041001

Penguji III

Dian Candra Rini Novitasari, M. Kom
NIP. 198511242014032001

Penguji IV

Aris Fanani, M. Kom
NIP. 198701272014031002

Mengetahui,

Dekan Fakultas Sains dan Teknologi
UIN Sunan Ampel Surabaya



Saepul Hamdani, M. Pd
NIP. 196507312000031002



UIN SUNAN AMPEL
SURABAYA

KEMENTERIAN AGAMA
UNIVERSITAS ISLAM NEGERI SUNAN AMPEL SURABAYA
PERPUSTAKAAN

Jl. Jend. A. Yani 117 Surabaya 60237 Telp. 031-8431972 Fax.031-8413300
E-Mail: perpus@uinsby.ac.id

LEMBAR PERNYATAAN PERSETUJUAN PUBLIKASI
KARYA ILMIAH UNTUK KEPENTINGAN AKADEMIS

Sebagai sivitas akademika UIN Sunan Ampel Surabaya, yang bertanda tangan di bawah ini, saya:

Nama : NUR HIDAYAH
NIM : H92219055
Fakultas/Jurusan : SAINS DAN TEKNOLOGI / MATEMATIKA
E-mail address : nh8861@gmail.com

Demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Perpustakaan UIN Sunan Ampel Surabaya, Hak Bebas Royalti Non-Eksklusif atas karya ilmiah :

Skripsi Tesis Desertasi Lain-lain (.....)
yang berjudul :

SISTEM DIAGNOSIS OSTEDARTHRTIS LUTUT BERDASARKAN CITRA X-RAY
MENGGUNAKAN METODE CNN JENIS RESNET .

beserta perangkat yang diperlukan (bila ada). Dengan Hak Bebas Royalti Non-Eksklusif ini Perpustakaan UIN Sunan Ampel Surabaya berhak menyimpan, mengalih-media/format-kan, mengelolanya dalam bentuk pangkalan data (database), mendistribusikannya, dan menampilkan/mempublikasikannya di Internet atau media lain secara *fulltext* untuk kepentingan akademis tanpa perlu meminta ijin dari saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan atau penerbit yang bersangkutan.

Saya bersedia untuk menanggung secara pribadi, tanpa melibatkan pihak Perpustakaan UIN Sunan Ampel Surabaya, segala bentuk tuntutan hukum yang timbul atas pelanggaran Hak Cipta dalam karya ilmiah saya ini.

Demikian pernyataan ini yang saya buat dengan sebenarnya.

Surabaya, 27 Januari 2023

Penulis

(NUR HIDAYAH)
nama terang dan tanda tangan

ABSTRAK

SISTEM DIAGNOSIS OSTEOARTHRITIS LUTUT BERDASARKAN CITRA X-RAY MENGGUNAKAN METODE CNN JENIS RESNET

Osteoarthritis lutut (OA lutut) merupakan penyakit sendi yang paling banyak menyumbang angka kecacatan pada usia lanjut. OA lutut ini dapat di diagnosis, salah satunya adalah dengan menggunakan citra x-ray. Berdasarkan data tersebut, OA lutut dapat dibedakan menjadi lima tingkatan, yaitu *grade 0*, *grade 1*, *grade 2*, *grade 3*, *grade 4*. Penelitian ini, bertujuan untuk mendiagnosis masing-masing tingkatan pada osteoarthritis lutut menggunakan salah satu metode *deep learning* yaitu *Convolutional Neural Network* jenis ResNet-50. Pada penelitian ini, hasil optimal diperoleh pada data yang telah dilakukan augmentasi dengan pembagian data menggunakan *k-fold cross validation*. Jumlah *batch size* yang digunakan untuk mendapatkan hasil optimal yaitu sebanyak 32. Pada uji coba tersebut, secara berturut-turut nilai sensitivitas, akurasi, dan spesifisitas yang didapatkan adalah 99.61%, 99.54%, dan 99.89%. Hasil tersebut diharapkan mampu menjadi alternatif dalam melakukan diagnosis osteoarthritis lutut.

Kata kunci: Citra X-ray, CNN, *Deep Learning*, Osteoarthritis Lutut, Residual Network (ResNet)

UIN SUNAN AMPEL
S U R A B A Y A

ABSTRACT

KNEE OSTEOARTHRITIS DIAGNOSIS SYSTEM BASED ON X-RAY IMAGES USING CNN MODEL RESNET

Knee osteoarthritis is a joint disease that contributes the most to disability rates, especially in elderly patients. Knee OA can be diagnosed, one of which is by using x-ray images. Based on these data, knee OA can be divided into five levels: grade 0, grade 1, grade 2, grade 3, and grade 4. This study aims to diagnose each stage of knee osteoarthritis using one of the deep learning methods, namely Convolutional Neural Network type ResNet50. In this study, optimal results were obtained on data that had been augmented by dividing the data using k-fold cross validation. The number of batch size used to obtain optimal results was 32. The sensitivity, accuracy, and specificity values obtained in this trial were 99.61%, 99.54%, and 99.89%. These results are expected to be an alternative in diagnosing knee osteoarthritis.

Keywords: Kata kunci: X-ray Image, CNN, *Deep Learning*, Knee Osteoarthritis, Residual Network (ResNet)

UIN SUNAN AMPEL
S U R A B A Y A

DAFTAR ISI

HALAMAN JUDUL	i
LEMBAR PERSETUJUAN PEMBIMBING	ii
PENGESAHAN TIM PENGUJI SKRIPSI	iii
HALAMAN PERNYATAAN KEASLIAN	iv
MOTTO	v
HALAMAN PERSEMBAHAN	vi
KATA PENGANTAR	vii
DAFTAR ISI	ix
DAFTAR TABEL	xi
DAFTAR GAMBAR	xii
ABSTRAK	xiv
ABSTRACT	xv
I PENDAHULUAN	1
1.1. Latar Belakang Masalah	1
1.2. Rumusan Masalah	8
1.3. Tujuan Penelitian	8
1.4. Manfaat Penelitian	8
1.5. Batasan Masalah	9
1.6. Sistematika Penulisan	10
II TINJAUAN PUSTAKA	12
2.1. Osteoarthritis (OA)	12
2.2. Osteoarthritis Lutut	13
2.3. Citra Digital	16
2.3.1. Citra RGB	17
2.3.2. Citra <i>Grayscale</i>	18
2.3.3. Citra Biner	19
2.4. Augmentasi Data	20

2.5. <i>K-Fold Cross Validation</i>	21
2.6. <i>Convolutional Neural Network (CNN)</i>	21
2.6.1. <i>Convolution Layer</i>	23
2.6.2. <i>Batch Normalization</i>	24
2.6.3. <i>Rectified Linear Unit (ReLU)</i>	25
2.6.4. <i>Pooling Layer</i>	25
2.6.5. <i>Fully Connected Layer</i>	26
2.6.6. <i>Softmax Layer</i>	27
2.6.7. <i>Cross Entropy</i>	28
2.7. <i>Deep Residual Network (ResNet)</i>	28
2.8. <i>Confusion Matrix</i>	30
2.9. <i>Penyakit dalam Perspektif Islam</i>	32
III METODE PENELITIAN	35
3.1. <i>Jenis Penelitian</i>	35
3.2. <i>Sumber Data</i>	35
3.3. <i>Kerangka Penelitian</i>	36
IV HASIL DAN PEMBAHASAN	39
4.1. <i>Preprocessing</i>	39
4.2. <i>Diagnosis OA Lutut Menggunakan Metode CNN Jenis ResNet</i>	42
4.3. <i>Analisis Hasil Diagnosis OA Lutut Sebelum Augmentasi</i>	58
4.4. <i>Analisis Hasil Diagnosis OA Lutut Setelah Augmentasi</i>	62
4.5. <i>Aplikasi Diagnosis OA Lutut</i>	66
4.6. <i>Sistem Diagnosis dalam Perspektif Islam</i>	72
V PENUTUP	76
5.1. <i>Kesimpulan</i>	76
5.2. <i>Saran</i>	76
DAFTAR PUSTAKA	77

DAFTAR TABEL

2.1	<i>Confusion Matrix</i> sebanyak 5 kelas	30
4.1	Hasil Uji Coba Sebelum Augmentasi	59
4.2	Hasil Uji Coba Sebelum Augmentasi	60
4.3	Grafik Proses <i>Training</i> pada Model Optimal	61
4.4	Hasil <i>Confusion Matrix</i> pada Model Optimal	61
4.5	Hasil Uji Coba Sesudah Augmentasi	63
4.6	Hasil Uji Coba Sesudah Augmentasi	64
4.7	Grafik Proses <i>Training</i> pada Model Optimal	65
4.8	Hasil <i>Confusion Matrix</i> pada Model Optimal	65

UIN SUNAN AMPEL
S U R A B A Y A

DAFTAR GAMBAR

2.1 Perbandingan Struktur Sendi Normal dan Osteoarthritis	12
2.2 Tingkatan Kelas pada Osteoarthritis Lutut pada: (a) Citra MRI; (b) Citra USG	14
2.3 Tingkatan Kelas pada Osteoarthritis Lutut	15
2.4 Koordinat Kartesius Citra RGB	17
2.5 Contoh Citra RGB dan Nilai Pikselnya	18
2.6 Contoh Citra <i>Grayscale</i> dan Nilai Pikselnya	19
2.7 Contoh Citra Biner dan Nilai Pikselnya	19
2.8 Proses Pembagian Data Menggunakan <i>K-Fold Cross Validation</i>	21
2.9 Arsitektur CNN	22
2.10 Proses <i>Convolution Layer</i>	24
2.11 Contoh <i>Pooling Layer</i>	26
2.12 Proses <i>Fully Connected Layer</i>	27
2.13 <i>Residual Block</i>	29
2.14 Arsitektur ResNet-50	29
3.1 Citra X-ray Osteoarthritis Lutut	36
3.2 Diagram Alir Penelitian	37
4.1 Sampel Cita pada <i>Grade 4</i> yang akan diaugmentasi	40
4.2 Hasil Augmentasi	41
4.3 Matriks <i>Input</i> dengan <i>Padding = 3</i>	42
4.4 Perhitungan <i>Convolution Layer</i> dengan <i>Stride = 2</i>	43
4.5 Perhitungan <i>Convolution Layer</i> dengan <i>Stride = 2</i>	43
4.6 Visualisasi <i>feature map</i> pada Proses <i>Convolution Layer</i>	45
4.7 Visualisasi <i>feature map</i> pada Proses <i>Batch Normalization</i>	47
4.8 Visualisasi <i>feature map</i> pada Proses ReLU	48
4.9 Operasi <i>Max Pooling</i> dengan <i>Stride = 2</i>	49

4.10 Operasi <i>Max Pooling</i> dengan <i>Stride = 2</i>	49
4.11 Operasi <i>Max Pooling</i> dengan <i>Stride = 2</i>	50
4.12 Operasi <i>Addition Layer</i>	51
4.13 Operasi <i>Average Pooling</i> dengan <i>Stride = 2</i>	51
4.14 Operasi <i>Average Pooling</i> dengan <i>Stride = 2</i>	52
4.15 Hasil <i>Average Pooling</i>	53
4.16 Tampilan Awal Aplikasi Diagnosis OA Lutut	67
4.17 Tampilan Aplikasi Diagnosis OA Lutut	67
4.18 Tampilan Aplikasi Diagnosis OA Lutut	68
4.19 Tampilan Aplikasi Diagnosis OA Lutut	68
4.20 Tampilan Aplikasi Diagnosis OA Lutut	69
4.21 Tampilan Aplikasi Diagnosis OA Lutut	70
4.22 Tampilan Aplikasi Diagnosis OA Lutut	71
4.23 Tampilan Aplikasi Diagnosis OA Lutut	71



UIN SUNAN AMPEL
S U R A B A Y A

BAB I

PENDAHULUAN

1.1. Latar Belakang Masalah

Manusia merupakan makhluk Allah yang senantiasa hidup berdampingan dengan beraneka ragam ujian. Allah SWT menurunkan ujiannya kepada manusia tidak serta merta tanpa disertai sebuah alasan. Bukan hanya karena sebuah kesalahan yang dilakukan oleh manusia, namun ujian juga dapat diturunkan untuk menguji keimanan dan kesabaran, sebagaimana termaktub dalam Al-Quran Surah Al-'Ankabut ayat 2 yang berbunyi:

أَحْسِبِ النَّاسَ أَنْ يُتْرَكُوا أَنْ يَقُولُوا آمَنَّا وَهُمْ لَا يُفْتَنُونَ

Artinya: Apakah manusia mengira bahwa mereka akan dibiarkan hanya dengan mengatakan, "Kami telah beriman" dan mereka tidak diuji? (QS. Al-'Ankabut: 2). Ayat tersebut menjelaskan bahwa setiap orang pasti akan diuji, karena keimanan seseorang tidak dapat dibuktikan hanya dengan sebuah pernyataan "kami telah beriman kepada Allah". Namun, hakikat iman yang sesungguhnya adalah keteguhan dan kesabaran dalam menghadapi segala bentuk ujian yang diberikan.

Salah satu bentuk ujian dari Allah yang sering diberikan kepada manusia adalah berupa penyakit. Berbagai macam jenis penyakit dapat menyerang tubuh manusia, baik penyakit yang menular maupun tidak menular. Penyakit Tidak Menular (PTM) seringkali tidak terdeteksi sejak dini karena minim atau bahkan tidak menunjukkan gejala. Oleh karena itu, tingkat kematian yang disebabkan oleh

PTM ini juga terbilang sangat tinggi. Pada tahun 2021, *World Health Organization* (WHO) menjelaskan bahwa setiap tahunnya PTM membunuh sekitar 41 juta jiwa yang setara dengan 71% dari seluruh kematian global (WHO, 2022).

Salah satu penyakit tidak menular yang kerap kali diderita oleh masyarakat adalah penyakit arthritis. Arthritis merupakan penyakit yang ditunjukkan dengan adanya pembengkakan dan nyeri pada sendi (Li, 2022). Arthritis ini adalah salah satu penyakit kronis yang biasanya menyerang pria dan wanita pada usia 55 tahun keatas. Macam-macam tipe dari penyakit arthritis meliputi osteoarthritis, rheumatoid arthritis, psoriatic arthritis, fibromyalgia, dan gout (Khaleghi, 2020). Osteoarthritis (OA) merupakan jenis arthritis yang paling banyak diderita (Venkatachalam et al., 2018). OA dapat mempengaruhi kinerja seluruh bagian dari sendi, yang melibatkan tulang rawan, tulang subkondral, sinovium, kapsul, ligamen, dan otot-otot di sekitarnya (Hussain and Khuntela, 2020). Secara klinis, OA dapat ditandai dengan nyeri sendi, kekakuan, dan keterbatasan fungsional (Yahaya et al., 2021). OA ini dapat menyerang pada bagian jari, lutut, dan pinggul (Victoria, 2022). Osteoarthritis lutut merupakan jenis osteoarthritis yang paling umum terjadi (Yoo et al., 2018).

Osteoarthritis lutut adalah penyakit sendi yang menyumbang angka kecacatan paling banyak pada usia lanjut (Kan et al., 2019). Prevalensi OA lutut akan terus bertambah seiring bertambahnya usia dan obesitas. OA lutut menyerang sekitar 13% wanita dan 10% pria yang berusia 60 tahun keatas. Pada usia 70 tahun keatas, kemungkinan terkena OA lutut meningkat hingga 40%. Dari data tersebut, menunjukkan bahwa wanita memiliki kemungkinan yang lebih besar dibandingkan pria untuk terkena penyakit OA lutut. Hal tersebut dapat terjadi karena pada usia yang lebih tua, anggota tubuh yang ada pada manusia kembali melemah,

sebagaimana yang dijelaskan dari Al-Qur'an pada Surah Ar-Rum ayat 54 yang berbunyi:

اللَّهُ الَّذِي خَلَقَكُمْ مِنْ ضَعْفٍ ثُمَّ جَعَلَ مِنْ بَعْدِ ضَعْفٍ قُوَّةً ثُمَّ جَعَلَ
مِنْ بَعْدِ قُوَّةٍ ضَعْفًا وَشَيْبَةً يَخْلُقُ مَا يَشَاءُ وَهُوَ الْعَلِيمُ الْقَدِيرُ

Artinya: Allah, Dialah yang menciptakan kamu dari keadaan lemah, kemudian Dia menjadikan (kamu) sesudah keadaan lemah itu menjadi kuat, kemudian Dia menjadikan (kamu) sesudah kuat itu lemah (kembali) dan beruban. Dia menciptakan apa yang dikehendaki-Nya dan Dialah Yang Maha Mengetahui lagi Maha Kuasa. Berdasarkan firman Allah tersebut, diterangkan bahwa manusia pada usia tua akan menjadi lemah kembali. Oleh karena itu, manusia dengan usia tua akan lebih rentan terkena penyakit OA lutut.

Penderita OA lutut seringkali tidak mengetahui jika terkena OA lutut. Hal tersebut terjadi karena sebagian besar OA lutut tidak menunjukkan adanya gejala. Salah satu penelitian menunjukkan bahwa hanya 15% dari keseluruhan kasus OA lutut yang menunjukkan adanya gejala (Hsu and Siwiec, 2018). Sehingga, OA lutut lebih banyak diketahui jika sudah sangat parah atau berada pada stadium lanjut. OA lutut pada stadium lanjut tidak dapat disembuhkan kecuali dengan melakukan artroplasti lutut. Namun, tindakan artroplasti ini membutuhkan biaya yang sangat besar (Cui et al., 2020). Oleh karena itu, untuk meminimalisir terjadinya hal tersebut, dapat dilakukan dengan adanya diagnosis sejak awal. Proses diagnosis sejak dini juga merupakan bentuk dari ikhtiar manusia dalam meminimalisir resiko terhadap penyakit OA lutut. Hal tersebut sejalan dengan salah satu kaidah fikih yang berbunyi:

دَرْءُ الْمَفَاسِدِ أَوْلَىٰ مِنْ جَلْبِ الْمَصَالِحِ

Artinya: "Menghilangkan kemudharatan itu lebih didahulukan daripada mengambil sebuah kemaslahatan". Dari kaidah fikih tersebut, menjelaskan bahwa sesuatu yang menyebabkan mudharat harus dihilangkan terlebih dahulu. Hal tersebut sejalan dengan proses diagnosis yang merupakan salah satu bentuk pencegahan OA lutut yang berguna untuk mengetahui pengobatan yang akan dilakukan selanjutnya.

Terdapat beberapa cara yang telah diterapkan untuk diagnosis OA lutut, yang meliputi pemeriksaan fisik, pemeriksaan arthroscopic, aspirasi sendi, dan pemeriksaan radiologi (Teoh et al., 2022). Pemeriksaan radiologi dianggap lebih efisien daripada alternatif lainnya, karena pemeriksaan radiologi mampu memberikan informasi mengenai gambaran dan informasi yang lebih jelas tentang organ dalam tubuh (Bork et al., 2019). Terdapat beberapa bentuk yang didapatkan dari pemeriksaan radiologi pada OA lutut, diantaranya berbentuk citra x-ray, *Magnetic Resonance Image* (MRI), dan *Ultrasonografi* (USG) (Thomas et al., 2022). Hingga saat ini, citra x-ray lebih sering dijadikan sebagai alternatif untuk diagnosis OA lutut karena biaya yang lebih terjangkau dan waktu pengambilan gambar yang lebih singkat. Meskipun demikian, citra x-ray mampu memberikan hasil yang jelas dan dapat dijadikan bahan untuk diagnosis secara akurat. Citra tersebut dapat di diagnosa secara lebih mudah dengan menerapkan metode *deep learning* sebagai bentuk pengembangan teknologi.

Metode *deep learning* telah banyak diterapkan untuk klasifikasi, dan telah terbukti menjadi metode terbaik dalam ekstraksi fitur, pencitraan medis, dan klasifikasi objek (Tekade and Rajeswari, 2018). Macam-macam metode *deep learning* yang umum digunakan antara lain *Deep Neural Network* (DNN), *Recurrent*

Neural Network (RNN), dan *Convolutional Neural Network* (CNN) (Koklu et al., 2021). Metode CNN mampu mempelajari berbagai atribut yang diinputkan pada sistem dengan baik, sehingga CNN mampu memperoleh karakteristik data lebih detail dan memiliki kemampuan deteksi yang lebih baik dibandingkan ANN dan DNN (Wang et al., 2020)(Feng et al., 2021). Pada penelitian mengenai identifikasi kesalahan yang didasarkan pada sinyal untrasonik, CNN mampu mendapatkan nilai AUC sebesar 99.8% sedangkan RNN dan DNN hanya menghasilkan nilai AUC 96.1% dan 98.7% (Zhang et al., 2018).

Penelitian lain yang menunjukkan keunggulan CNN juga dilakukan pada klasifikasi citra medis dengan menggunakan 3 metode yakni CNN, ANN, dan *Support Vector Machine* (SVM). Pada penelitian tersebut, akurasi yang diberikan oleh CNN adalah 100.0% sedangkan ANN dan SVM secara berturut-turut menghasilkan akurasi sebesar 88.9% dan 94.4% (Maruyama et al., 2018). CNN memiliki beberapa jenis arsitektur, diantaranya meliputi AlexNet, MobileNet, GoogleNet, ResNet, dan lain sebagainya (Naz et al., 2022). ResNet merupakan arsitektur CNN yang terkenal pada tahun 2015 dan menjadi pemenang pada kompetisi *ImageNet Large Scale Visual Challenge 2015* (ILSVRC 2015). Arsitektur ResNet memiliki keunggulan mampu melatih ratusan hingga ribuan lapisan tanpa mengurangi kinerjanya. Arsitektur ini memiliki kemampuan representasi yang kuat meskipun melakukan kinerja yang banyak (Razzak et al., 2020).

Berdasarkan pada kelebihan-kelebihan yang dimiliki oleh arsitektur ResNet, membuat arsitektur ini banyak diterapkan untuk masalah klasifikasi gambar pada penelitian-penelitian terdahulu. Salah satu penelitian sebelumnya membahas mengenai klasifikasi COVID-19 dengan membandingkan beberapa metode deep learning yang meliputi ResNet, GoogleNet, VGG-16, AlexNet, MobileNet, dan

Inception-V3. Nilai sensitivitas dari penelitian tersebut secara berturut-turut adalah 100.0% 96.6% 96.6% 98.3% 93.3% dan 88.3%. Dari hasil tersebut, menunjukkan bahwa nilai sensitivitas yang diberikan metode ResNet mampu melampaui nilai sensitivitas yang dihasilkan oleh arsitektur lainnya (Maheen et al., 2021). Selain itu, ResNet juga telah banyak diaplikasikan untuk identifikasi di bidang medis. Salah satunya diterapkan pada penelitian mengenai klasifikasi kanker melanoma. Pada penelitian tersebut, ResNet mampu memberikan akurasi sebesar 83.0% (Budhiman et al., 2019). Selain itu, penelitian menggunakan ResNet juga dilakukan pada klasifikasi diabetik retinopati dan mampu memberikan hasil akurasi sebesar 86.67% (Elsawah et al., 2020). Pada penelitian lain, ResNet diaplikasikan untuk deteksi karsinoma duktal invasif dan memberikan akurasi sebesar 91.0% (Chapala and Sujatha, 2020).

Berdasarkan susunan *layer*nya, ResNet memiliki beberapa model seperti ResNet-18, ResNet-34, ResNet-50, ResNet-101, dan ResNet-152 (Sukegawa et al., 2021). Dari beberapa model ResNet tersebut, ResNet-50 memiliki keunggulan yaitu mampu mengekstraksi fitur dengan kualitas tinggi sehingga menjadikan model tersebut mempunyai kinerja yang baik dalam masalah klasifikasi gambar (Wen et al., 2020). Hal tersebut juga dibuktikan dengan adanya penelitian mengenai deteksi Covid menggunakan citra x-ray yang membandingkan model ResNet. Berdasarkan penelitian tersebut, ResNet-50 mampu memberikan nilai sensitivitas tertinggi dibandingkan dengan arsitektur lainnya (Showkat and Qureshi, 2022). Selain itu, penelitian mengenai diagnosis kanker serviks juga membandingkan antara ResNet-18, ResNet-50 dan ResNet-101. Hasil dari penelitian tersebut menunjukkan bahwa ResNet-50 dan ResNet-101 memberikan nilai akurasi 100%. Namun, waktu yang dihabiskan oleh ResNet-50 lebih cepat daripada ResNet-101 (Novitasari et al.,

(2022). Sehingga, berdasarkan penelitian-penelitian tersebut, metode ResNet-50 telah terbukti dapat melakukan klasifikasi gambar dengan kinerja yang baik. Proses diagnosis menggunakan metode ResNet-50 ini juga merupakan salah satu bentuk usaha manusia dalam mencari pengobatan, sebab setiap penyakit yang diturunkan oleh Allah SWT pasti disertai dengan obatnya. Sebagaimana sabda Rasulullah dalam hadits yang diriwayatkan oleh Imam Bukhari Nomor 5678:

حَدَّثَنَا مُحَمَّدُ بْنُ الْمُثَنَّى حَدَّثَنَا أَبُو أَحْمَدَ الزُّبَيْرِيُّ حَدَّثَنَا عُمَرُ بْنُ سَعِيدِ بْنِ أَبِي حُسَيْنٍ قَالَ: حَدَّثَنِي عَطَاءُ بْنُ أَبِي رَبَاحٍ عَنْ أَبِي هُرَيْرَةَ رَضِيَ اللَّهُ عَنْهُ عَنِ النَّبِيِّ صَلَّى اللَّهُ عَلَيْهِ وَسَلَّمَ قَالَ: «مَا أَنْزَلَ اللَّهُ دَاءً إِلَّا أَنْزَلَ لَهُ شِفَاءً».

Artinya: Telah menceritakan kepada kami [Muhammad bin al-Mutsanna] telah menceritakan kepada kami [Abu Ahmad Az Zubairi] telah menceritakan kepada kami [’Umar bin Sa’id bin Abu Husain] dia berkata: telah menceritakan kepadaku [’Atha’ bin Abu Rabah] dari [Abu Hurairah RA] Nabi Shallallahu ’Alaihi Wasallam beliau bersabda: ”Allah tidak akan menurunkan penyakit melainkan menurunkan obatnya juga” (H.R. Bukhori 5678). Hadits yang diriwayatkan oleh Imam Bukhori tersebut menjelaskan bahwa segala bentuk penyakit yang Allah turunkan pasti disertai dengan penawarnya. Oleh karena itu, adanya penelitian untuk diagnosis OA lutut ini merupakan salah satu bentuk ikhtiar dalam mencari pengobatan.

Berdasarkan penelitian-penelitian dan pemaparan yang telah dijelaskan sebelumnya, metode CNN dengan arsitektur ResNet-50 memiliki kinerja yang baik dalam pemrosesan gambar, sehingga penelitian ini memanfaatkan metode CNN jenis ResNet-50 untuk mendiagnosis osteoarthritis lutut berdasarkan data citra x-ray. Penelitian ini diharapkan dapat membangun sistem diagnosis citra osteoarthritis

lutut menggunakan metode CNN jenis ResNet-50. Dengan itu, sistem yang telah dibuat diharapkan kedepannya dapat menjadi alternatif sebagai pendeteksi dini osteoarthritis lutut.

1.2. Rumusan Masalah

Berdasarkan permasalahan yang dijelaskan pada latar belakang, penelitian ini disusun dengan rumusan masalah sebagai berikut:

1. Bagaimana hasil optimal ResNet-50 dalam diagnosis osteoarthritis lutut menggunakan citra x-ray berdasarkan uji coba pembagian data dan *batch size*?
2. Bagaimana perbandingan hasil diagnosis OA lutut menggunakan metode ResNet-50 pada data augmentasi dan non augmentasi?

1.3. Tujuan Penelitian

Berdasarkan rumusan masalah yang dijelaskan sebelumnya, penelitian ini dibuat dengan tujuan sebagai berikut:

1. Dapat mengetahui hasil optimal ResNet-50 dalam diagnosis osteoarthritis lutut menggunakan citra x-ray berdasarkan uji coba pembagian data dan *batch size*.
2. Dapat mengetahui perbandingan hasil diagnosis OA lutut menggunakan metode ResNet-50 pada data augmentasi dan non augmentasi.

1.4. Manfaat Penelitian

Penelitian ini diharapkan dapat membawa manfaat bagi berbagai lapisan masyarakat, seperti yang dijelaskan berikut:

1. Manfaat Teoritis

Penelitian ini diharapkan dapat menjadi referensi bagi penulis-penulis selanjutnya yang mengangkat topik mengenai diagnosis osteoarthritis lutut atau dalam proses diagnosis menggunakan arsitektur CNN jenis ResNet.

2. Manfaat Praktis

(a) Bagi Penulis

Menambah pengetahuan baru bagi penulis dalam memahami penyakit osteoarthritis lutut dan proses diagnosisnya menggunakan arsitektur CNN jenis ResNet.

(b) Bagi Ahli Medis

Membantu ahli medis dalam melakukan diagnosis osteoarthritis lutut dengan lebih efisien yang mampu mendapatkan akurasi yang maksimal.

(c) Bagi Masyarakat Umum

Memberi edukasi kepada masyarakat umum tentang osteoarthritis lutut dan menambah keilmuan dalam bidang sains, teknologi maupun kedokteran.

1.5. Batasan Masalah

Ruang lingkup penelitian ini akan sangat luas jika tidak ada batasan permasalahan. Batasan permasalahan yang digunakan dalam penelitian ini yaitu:

1. Data citra yang digunakan pada penelitian ini merupakan data citra x-ray osteoarthritis lutut pada 5 kelas (*grade 0*, *grade 1*, *grade 2*, *grade 3*, dan *grade 4*).
2. Tahap *preprocessing* yang digunakan pada penelitian ini yaitu augmentasi citra pada *grade 4*.

3. Metode pendekatan yang digunakan untuk diagnosis osteoarthritis lutut adalah CNN dengan model ResNet-50.

1.6. Sistematika Penulisan

Penelitian ini secara umum tersusun dari tiga bagian, yaitu bagian awal, isi, dan akhir.

1. Bagian Awal

Bagian awal ini berisi halaman judul, lembar persetujuan, pernyataan keaslian, motto, kata pengantar, abstrak, daftar isi, dan daftar gambar.

2. Bagian Isi

Bagian ini merupakan pokok pembahasan yang terdiri dari beberapa bab terkait penelitian ini, meliputi:

(a) BAB I Pendahuluan

Berisi pemaparan mengenai latar belakang masalah, perumusan masalah, tujuan penelitian, manfaat penelitian, metodologi penelitian, dan sistematika penulisan.

(b) BAB II Tinjauan Pustaka

Berisi tentang *literature review* yang menjelaskan definisi, teori, dan persamaan yang digunakan dalam penelitian. Diantaranya meliputi osteoarthritis, osteoarthritis lutut, citra, augmentasi data, *Convolutional Neural Network* (CNN), ResNet, *confusion matrix*, dan integrasi osteoarthritis lutut dalam Islam.

(c) BAB III Metodologi Penelitian

Berisi pemaparan mengenai jenis penelitian, pengumpulan data, dan kerangka penelitian.

(d) BAB IV Hasil dan Pembahasan

Berisi penyajian hasil yang diperoleh dalam penelitian ini, dan analisis hasil untuk menarik kesimpulan sesuai dengan tujuan penelitian.

(e) BAB V Penutup

Berisi tentang kesimpulan dan saran dari hasil penelitian.

3. Bagian Akhir

Bagian ini berisi daftar pustaka dan lampiran-lampiran.



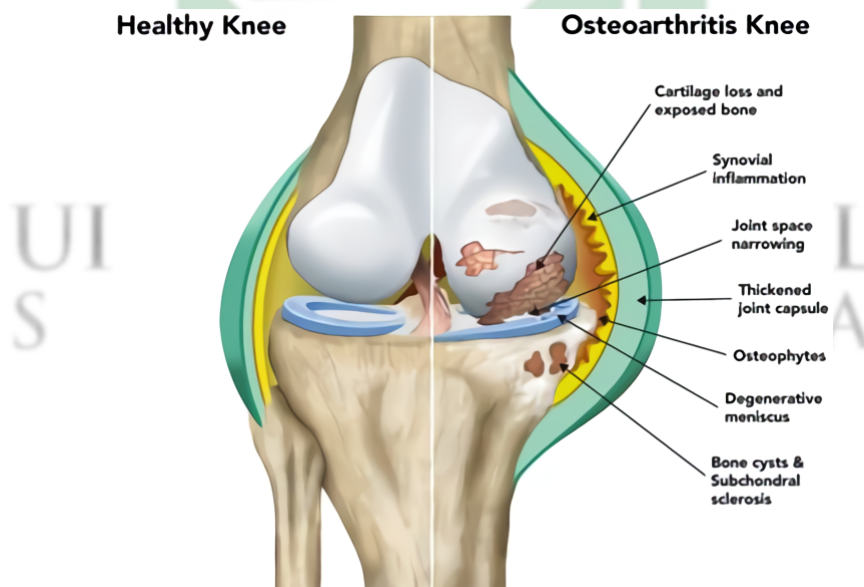
UIN SUNAN AMPEL
S U R A B A Y A

BAB II

TINJAUAN PUSTAKA

2.1. Osteoarthritis (OA)

OA merupakan peradangan pada sendi yang disebabkan karena penipisan dan kerusakan tulang rawan (Pavone et al., 2021). OA akan mempengaruhi struktur sendi penderitanya, termasuk pada tulang rawan *artikular*, tulang *subkondral*, *meniskus*, membran *sinovial*, dan bantalan lemak *infrapatellar*. Perubahan yang terjadi karena OA ini dapat berupa degradasi tulang rawan, tulang *subkondral*, pembentukan osteofit, serta perubahan *sinovium* dan kapsul sendi (Chow and Chin, 2020). Perubahan tersebut lebih jelasnya ditampilkan pada Gambar 2.1.



Sumber: (Prostem, 2019)

Gambar 2.1 Perbandingan Struktur Sendi Normal dan Osteoarthritis

2.2. Osteoarthritis Lutut

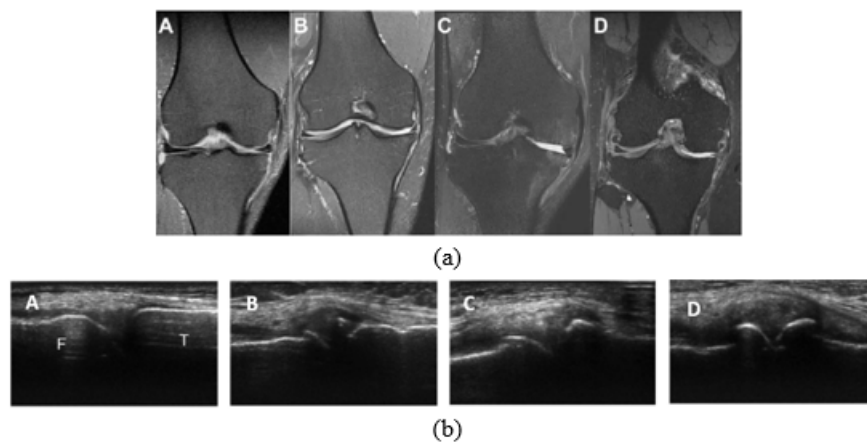
Osteoarthritis lutut atau yang dikenal dengan radang sendi lutut merupakan masalah kesehatan yang mengenai tulang dan sendi di sekitar lutut (Jones et al., 2019). OA lutut ini mempengaruhi 3 kompartemen yang ada pada sendi lutut yaitu medial, lateral, dan sendi petlofemoral (Lespasio et al., 2017). Pada sendi lutut yang merupakan sendi sinovial terbesar mausia, terdapat pertemuan antara dua tulang yaitu tulang femur yang terdapat pada bagian bawah, dan tulang tibia yang berada pada bagian atas dimana pada masing-masing ujungnya terdapat tulang rawan (Jang et al., 2021). Pada penderita OA lutut, tulang rawan yang ada pada dua sendi tersebut mengalami kerusakan yang menyebabkan tulang femur dan tibia bergesekan secara langsung dan menimbulkan rasa nyeri (Thienkarochanakul et al., 2020).

OA lutut dapat terjadi karena adanya beberapa penyebab, diantaranya adalah karena penuaan, inflamasi lokal, faktor genetik, tekanan mekanik dan proses sel, serta biokimia. Gejala yang ditimbulkan karena penyakit OA lutut ini meliputi nyeri pada sendi, pembengkakan sendi, perubahan bentuk sendi, sendi terasa kaku dan tidak dapat digerakkan dengan sempurna (Peterson, 2020). OA lutut dapat diketahui dengan beberapa langkah diagnosis, yaitu dengan pemberian informasi mengenai riwayat kesehatan ke dokter yang dilanjutkan dengan pemeriksaan fisik. Selanjutnya, akan dilakukan pemeriksaan lanjutan untuk memastikan prediksi awal dokter mengenai kondisi pasien, pemeriksaan tersebut meliputi (Teoh et al., 2022):

1. Pemindaian dengan melakukan pengambilan foto x-ray, MRI, ataupun USG untuk melihat kondisi tulang.
2. Tes darah untuk mengetahui penyebab lain dari osteoarthritis lutut.
3. Analisis cairan sendi untuk melihat kondisi peradangan pada sendi.

Berdasarkan beberapa alternatif pemeriksaan lanjutan diatas, alternatif diagnosis yang paling sering dilakukan adalah pemindaian. Hal tersebut disebabkan karena dengan dilakukannya pemindaian, dokter dapat lebih jelas melihat keadaan tulang sehingga memudahkan dalam proses diagnosis.

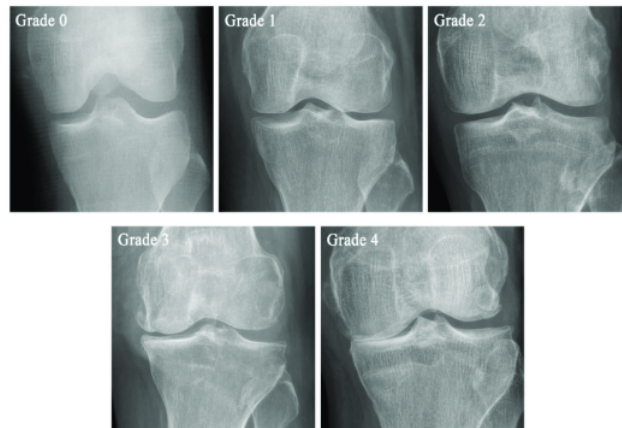
Diagnosis OA lutut dengan pemindaian mempunyai beberapa pilihan alternatif, meliputi pengambilan citra x-ray, MRI, ataupun USG. Contoh citra MRI dan USG pada OA lutut ditunjukkan pada Gambar 2.2.



Sumber: (Kim et al., 2020) (Tölli et al., 2017)

Gambar 2.2 Tingkatan Kelas pada Osteoarthritis Lutut pada: (a) Citra MRI; (b) Citra USG

Berdasarkan Gambar 2.2, dapat diketahui bahwa masing-masing dari citra MRI dan USG memiliki 4 *grade*. Namun, pemindaian dengan dua citra tersebut jarang dilakukan karena memiliki harga yang relatif mahal dan membutuhkan waktu yang lebih lama (Zeng et al., 2021). Selain itu, pada citra USG juga menampilkan tulang lutut dari sisi samping yang tidak terlihat jelas struktur tulangnya. Oleh karena itu, alternatif pemindaian yang sering dilakukan untuk diagnosis OA lutut adalah dengan citra x-ray seperti yang ditunjukkan oleh Gambar 2.3.



Sumber: (Jang et al., 2021)

Gambar 2.3 Tingkatan Kelas pada Osteoarthritis Lutut

Menurut Kellgren dan Lawrence, OA lutut dapat dibedakan berdasarkan beberapa parameter yaitu derajat osteofit, penyempitan celah sendi, perubahan bentuk tulang, dan *sklerosis tulang subkondral* (Kellgren and Lawrence, 1957). Secara lebih lengkap, perbedaan pada masing-masing tingkatan adalah sebagai berikut:

Grade 0: Normal

Grade 1: Curiga terdapat osteofit dan penyempitan sendi

Grade 2: Osteoarthritis ringan, ditandai dengan osteofit yang jelas dan terdapat penyempitan pada anteroposterior genu.

Grade 3: Osteoarthritis sedang, ditandai dengan osteofit sedang dan deformitas ruang antar sendi cukup besar

Grade 4: Osteoarthritis berat, ditandai dengan osteofit besar, deformitas ruang antar sendi yang berat dan sklerosis pada tulang subkondral

Berdasarkan tingkat keparahannya, pengobatan untuk penderita OA lutut meliputi metode konservatif dan operatif. Metode konservatif diterapkan pada penderita

osteoarthritis ringan yang meliputi perubahan aktivitas fisik, penurunan berat badan, dan latihan memperkuat otot. Apabila pengobatan konservatif dianggap gagal, maka dilakukan metode lanjutan yaitu metode operatif yang meliputi perbaikan tulang rawan, *osteotomi* dan penggantian sendi lutut (Philips et al., 2022).

2.3. Citra Digital

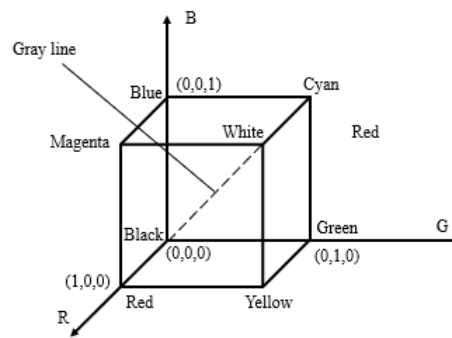
Citra digital merupakan representasi dari fungsi intensitas cahaya yang ada pada bidang dua dimensi (Haq, 2020)(Damanik, 2022). Secara matematis, citra digital dapat diartikan sebagai fungsi dua dimensi $f(x, y)$ yang merupakan proyeksi objek 3 dimensi ke dalam bidang 2 dimensi, dimana titik x, y merepresentasikan koordinat spasial citra, dan fungsi f pada titik x, y merepresentasikan tingkat keabuan (Sihotang, 2019). Selain itu, citra digital juga merupakan representasi matriks dari citra dua dimensi yang berisi informasi tentang nilai piksel di dalam citra (Dougherty, 2020). Citra digital dengan $I \times J$ piksel dapat diwakili oleh matriks dengan baris I dan kolom J seperti yang ditunjukkan pada Persamaan 2.1.

$$f(x, y) = \begin{bmatrix} f(1,1) & f(1,2) & f(1,3) & \cdots & f(1,J) \\ f(2,1) & f(2,2) & f(2,3) & \cdots & f(2,J) \\ f(3,1) & f(3,2) & f(3,3) & \cdots & f(3,J) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ f(I,1) & f(I,2) & f(I,3) & \cdots & f(I,J) \end{bmatrix} \quad (2.1)$$

Citra digital dapat dikategorikan berdasarkan nilai pikselnya, diantaranya adalah sebagai berikut (Yudhana et al., 2019):

2.3.1. Citra RGB

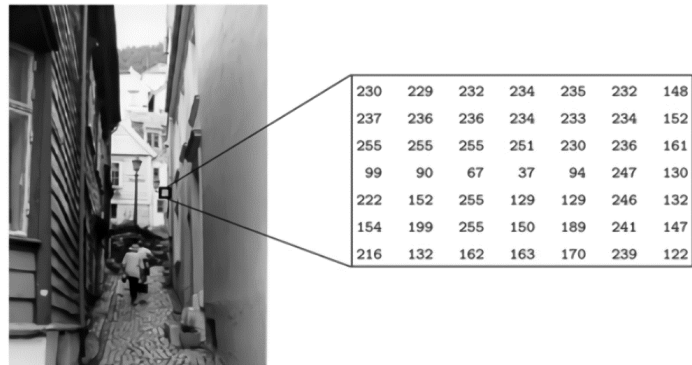
Citra RGB merupakan citra yang tersusun dari tiga kombinasi warna primer yaitu *Red* (R), *Green* (G), dan *Blue* (B) (Rizani, 2021). Model dari citra RGB dapat diwakili oleh sistem koordinat kartesian, seperti yang ditampilkan pada 2.4.



Gambar 2.4 Koordinat Kartesius Citra RGB

Berdasarkan 2.4, koordinat sumbu x, y, z mewakili R,G,B dimana setiap titik dalam ruangan tiga dimensi tersebut merepresentasikan kecerahan yang berada pada rentang 0 hingga 1. Pada koordinat awal $(0, 0, 0)$ gambar akan memiliki warna hitam, sedangkan pada simpul terjauh dengan koordinat $(1, 1, 1)$ berwarna putih. Garis lurus antara hitam dan putih disebut sebagai garis abu-abu yang berubah dari hitam menuju putih. Selain garis dan titik-titik tersebut, tiga titik sudut lainnya merupakan warna pelengkap dari tiga warna primer yaitu kuning, cyan, dan magenta (Xiong et al., 2018).

Setiap piksel pada citra RGB memiliki nilai terendah 0 dan tertinggi 255 serta mempunyai 8 bit pada tiap lapisannya (Singh, 2021). Sehingga secara keseluruhan citra RGB mempunyai 24 bit (Szyc, 2019). Contoh citra RGB dan informasi mengenai nilai-nilai pikselnya ditunjukkan 2.5.

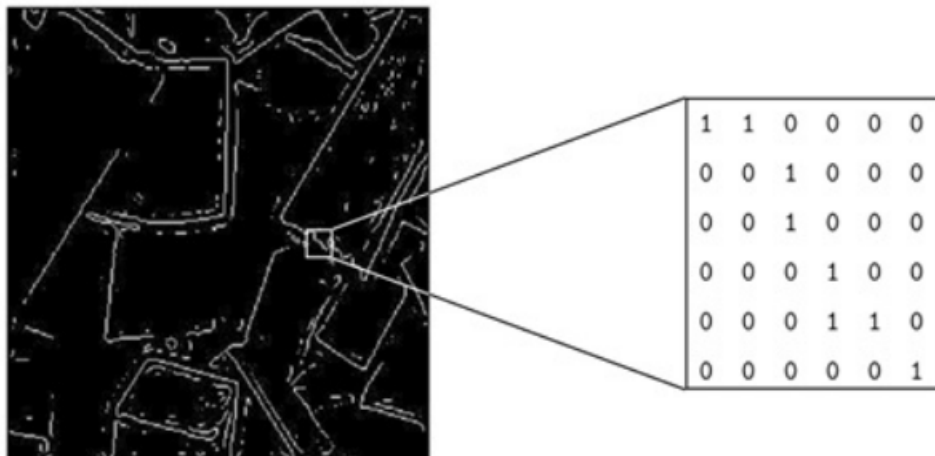


Sumber: (Kurnianto, 2013)

Gambar 2.6 Contoh Citra *Grayscale* dan Nilai Pixselnya

2.3.3. Citra Biner

Citra Biner merupakan jenis citra yang hanya memiliki nilai 0 dan 1. Nilai 0 mewakili warna hitam, dan nilai 1 mewakili warna putih (Sathesh and Adam, 2021). Citra biner ini merupakan citra yang paling sederhana dan paling ringan dalam penyimpanannya. Contoh citra biner ditunjukkan pada Gambar 2.7.



Sumber: (Kurnianto, 2013)

Gambar 2.7 Contoh Citra Biner dan Nilai Pixselnya

2.4. Augmentasi Data

Augmentasi data merupakan metode yang digunakan untuk memperbanyak jumlah data dengan cara memodifikasi data yang sudah ada (Sharma et al., 2020). Terdapat beberapa cara untuk melakukan proses augmentasi, diantaranya adalah dengan modifikasi warna dan transformasi geometrik (Shorten and Khoshgoftaar, 2019). Modifikasi warna dapat dilakukan dengan histogram equalization, modifikasi kontras, penajaman gambar, dan blurring (Mikołajczyk and Grochowski, 2018). Sedangkan augmentasi data dengan transformasi geometrik dapat dilakukan dengan rotasi, refleksi, dan dilatasi (Lai and Donsig, 2018). Modifikasi warna umumnya digunakan untuk augmentasi data tulang karena mampu memperbanyak data tanpa merubah posisi tulang seperti transformasi geometrik. Pada citra x-ray, metode modifikasi warna yang paling cocok digunakan adalah modifikasi kontras. Modifikasi kontras merupakan proses memetakan kembali intensitas dari sebuah gambar pada rentang yang baru (Maskuri et al., 2020). Modifikasi kontras pada citra dapat dilakukan secara *random* menggunakan Persamaan 2.2 sebagai berikut:

$$g(x) = \alpha f(x) + \beta \quad (2.2)$$

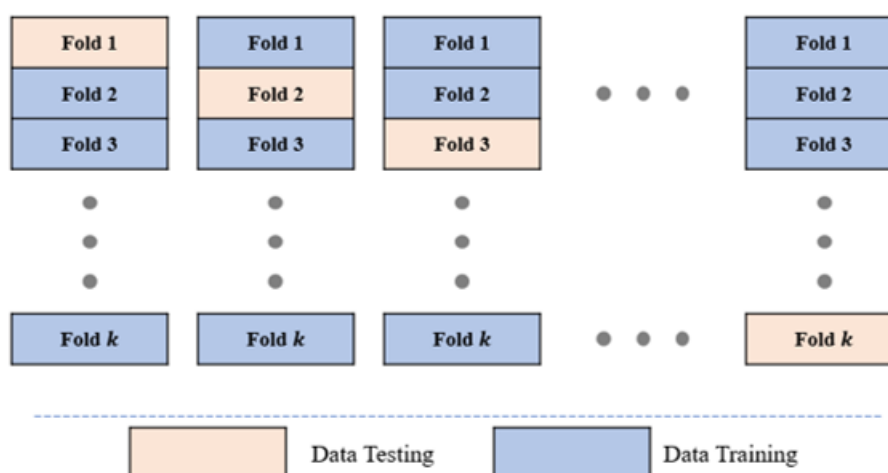
Dengan $\alpha > 0$ adalah bobot dan β adalah bias. Fungsi $g(x)$ adalah gambar *output* dan $f(x)$ adalah nilai pixel pada gambar *input*. Sehingga, dari Persamaan 2.2 dapat dibentuk Persamaan 2.3

$$g(i, j) = \alpha f(i, j) + \beta \quad (2.3)$$

Dimana (i, j) menunjukkan lokasi nilai pikse dengan i menunjukkan letak baris dan j letak kolom.

2.5. *K-Fold Cross Validation*

K-Fold Cross Validation merupakan suatu metode yang digunakan untuk mengetahui kinerja sistem klasifikasi (Tabrizchi et al., 2021). Metode ini bekerja dengan membagi data menjadi k partisi, dimana salah satu partisi digunakan sebagai data *testing* dan partisi lainnya dijadikan sebagai data *training*. Selanjutnya, proses *training* berulang sebanyak k kali dengan masing-masing data *testing* berada pada partisi yang berbeda. Proses diatas menjadikan *output* dari *k-fold cross validation* ini memiliki sebanyak k hasil akurasi. Hal tersebut merupakan keunggulan yang dimiliki oleh *k-fold cross validation* dibandingkan dengan melakukan pembagian data secara biasa. Proses pembagian data menggunakan *k-fold cross validation* ditunjukkan pada Gambar 2.8.



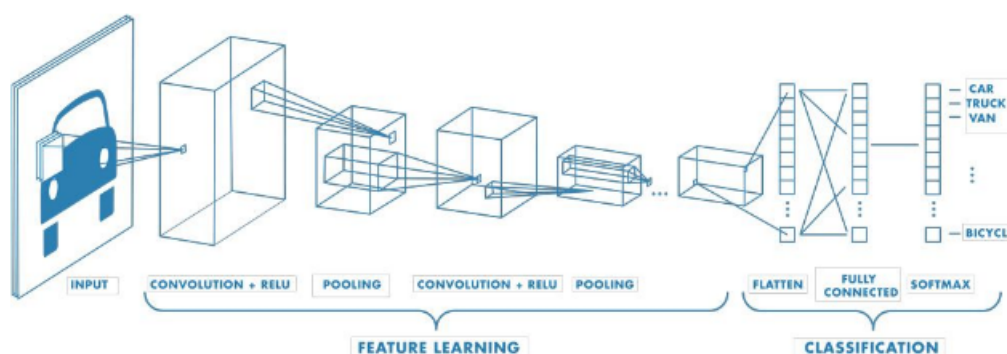
Gambar 2.8 Proses Pembagian Data Menggunakan *K-Fold Cross Validation*

2.6. *Convolutional Neural Network (CNN)*

CNN atau yang juga sering dikenal dengan ConvNet merupakan bagian dari *Deep Neural Networks* yang biasanya digunakan untuk menganalisis data citra (El-Kahlout and Abu-Naser, 2020). CNN merupakan pengembangan dari algoritma

Multi Layer Percepton (MLP) yang digunakan untuk pengolahan data pada bidang dua dimensi. Selain itu, CNN ini juga merupakan bagian dari algoritma *deep learning* *inputnya* dalam bentuk gambar yang mampu mempelajari gambar tersebut untuk memperoleh *feature learning*, dan juga dapat membedakan antara gambar yang satu dengan yang lainnya (Chauhan et al., 2018) (Jogin et al., 2018).

Arsitektur CNN dirancang menyerupai cara kerja sel saraf pada otak manusia. CNN ini terinspirasi oleh salah satu bagian otak yaitu *Visual Cortex* yang bertanggung jawab dalam pemrosesan informasi dalam bentuk visual sehingga CNN dapat mempelajari detail gambar dengan baik (Bhatt et al., 2021). Arsitektur CNN ditampilkan pada Gambar 2.9.



Sumber: (MathWorks, 2021)

Gambar 2.9 Arsitektur CNN

Berdasarkan 2.9 ditunjukkan bahwa arsitektur CNN memiliki beberapa lapisan untuk menjalankan algoritmanya. Lapisan-lapisan tersebut terdiri dari *convolution layer*, fungsi aktivasi ReLU, *normalization layer*, *pooling layer*, dan *fully connected layer*. Masing-masing lapisan tersebut akan dijelaskan sebagai berikut:

2.6.1. Convolution Layer

Convolution layer merupakan *layer* pertama pada proses *feature learning* pada CNN (Wang et al., 2018). Pada *convolution layer* ini diterapkan beberapa operasi konvolusi untuk mempelajari fitur-fitur dari data citra yang diinputkan (Mostafa et al., 2021). Operasi konvolusi berisi perkalian *dot product* antara nilai piksel citra dan matriks filter atau yang juga disebut dengan kernel (Haq, 2020). Kernel berisi nilai random antara -1 hingga 1 yang memiliki ukuran berbeda-beda sesuai dengan jenis arsitekturnya. Rumus convolutional layer ditunjukkan pada Persamaan 2.4.

$$Q_j = \left(\sum_{i=1}^N I_{i,j} \times F_{i,j} \right) + B_j \quad (2.4)$$

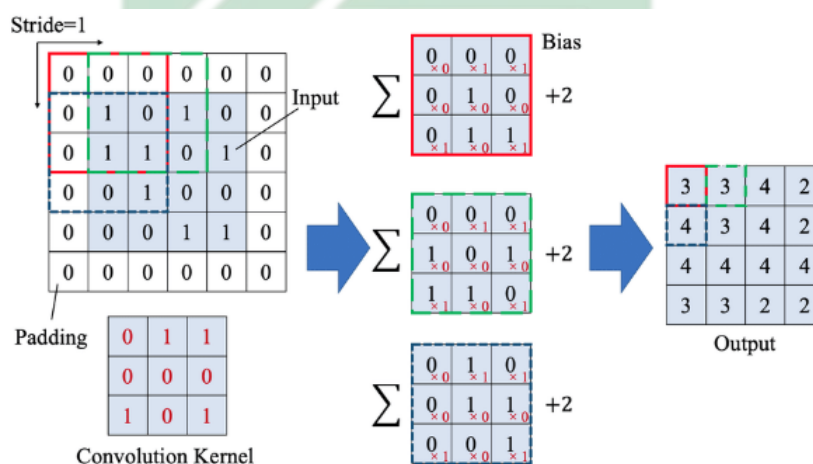
Dengan Q_j merupakan nilai dari proses convolution, sedangkan N menunjukkan banyaknya channel. Pada citra RGB, biasanya terdapat 3 channel yaitu red, green, dan blue. $I_{i,j}$ merupakan matriks yang berisi nilai piksel citra, $F_{i,j}$ merupakan matriks kernel, dan B_j sebagai bias. Sehingga, berdasarkan Persamaan 2.4 tersebut, nilai convolution didapatkan dari perkalian matriks yang berisi nilai piksel citra dengan matriks kernel yang kemudian ditambah dengan bias.

Pada algoritma CNN, kernel akan bergeser sejumlah dengan stride pada nilai piksel citra. Stride merupakan parameter yang menentukan langkah pergeseran matriks kernel baik searah sumbu x maupun sumbu y. Selain itu, terdapat pula padding yang merupakan parameter untuk menambah jumlah piksel dari setiap sisi. Pada setiap sisinya, ditambahkan dengan nilai 0 yang bertujuan agar tidak terjadi pengurangan fitur citra dan menyamakan ukuran matriks piksel dan kernel saat memiliki sisa (Rakamawati, 2021).

Proses *convolutional layer* menghasilkan *output* yang disebut dengan *feature map*. Ukuran dari *output* tersebut dapat dicari menggunakan Persamaan 2.5.

$$h = \frac{i - k + 2p}{s + 1} \quad (2.5)$$

Dimana h merupakan ukuran *feature map*, i adalah ukuran matriks input, k merupakan ukuran matriks kernel, p adalah *padding*, dan s adalah *stride*. Contoh proses *convolution* ditunjukkan pada Gambar 2.10.



Sumber: (Shi et al., 2021)

Gambar 2.10 Proses Convolution Layer

2.6.2. Batch Normalization

Batch Normalization merupakan teknik yang digunakan pada metode CNN untuk mempercepat waktu komputasi dan meningkatkan stabilitas sistem (Bilal et al., 2022). Pada *batch normalization* ini, dilakukan proses normalisasi yang bertujuan untuk mengurangi *overfitting* di setiap *mini-batch* (Zhu et al., 2020). *Overfitting* adalah keadaan dimana data terbaik digunakan untuk proses *training*, sehingga dapat mengurangi akurasi pada proses *testing*. Hasil dari *batch normalization* ini

dapat dicari menggunakan nilai rata-rata (μ), nilai varian (σ^2) pada matriks input H , parameter pembelajaran γ, β , serta nilai konstan untuk meningkatkan stabilitas numerik (ϵ). Dari nilai-nilai tersebut, dibangun persamaan untuk mencari output dari *batch normalization* seperti yang ditunjukkan pada Persamaan 2.6.

$$BN = \gamma \frac{H - \mu}{\sqrt{\sigma^2 + \epsilon}} + \beta \quad (2.6)$$

2.6.3. Rectified Linear Unit (ReLU)

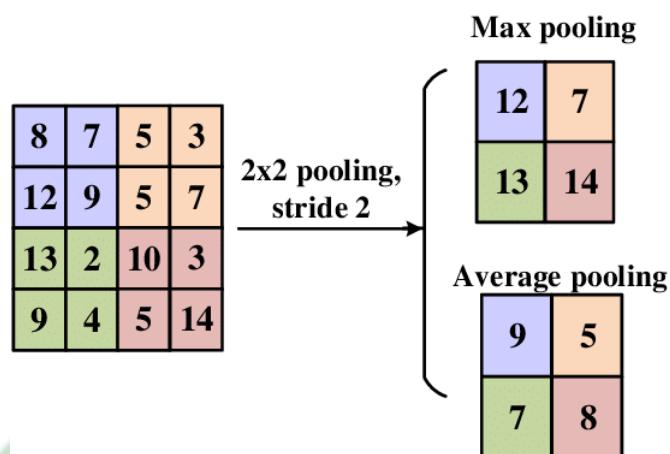
ReLU merupakan fungsi aktivasi yang paling sering digunakan pada metode CNN. Fungsi aktivasi merupakan sebuah fungsi yang memiliki tujuan untuk mengubah sinyal *input* dari sebuah node menjadi sinyal *output*. Fungsi aktivasi ReLU memiliki kinerja yang efisien karena waktu komputasi yang dibutuhkan jauh lebih cepat dibandingkan fungsi aktivasi lainnya (Ding et al., 2018). Secara lebih jelas, fungsi aktivasi ReLU digunakan untuk mengubah nilai x menjadi 0 jika nilai x tersebut kurang dari 0. Apabila nilai x tersebut tidak kurang dari 0, maka nilai x akan tetap dipertahankan. Dengan x adalah *feature map*. Secara matematis, fungsi tersebut dituliskan pada Persamaan 2.7.

$$f(x) = \max(0, x) \quad (2.7)$$

2.6.4. Pooling Layer

Pooling layer merupakan *layer* yang biasanya berada setelah *convolution layer* dan berfungsi untuk mengurangi jumlah parameter yang digunakan dengan operasi *down sampling* (Xiao et al., 2020). Umumnya, *max pooling* dan *average pooling* adalah jenis metode *pooling* yang paling sering digunakan. *Max pooling* merupakan jenis *pooling* yang mengambil nilai terbesar pada *feature map* (Yani,

(2019). Sedangkan *average pooling* adalah pengambilan nilai rata-rata pada *feature map* (Wang et al., 2018). Contoh operasi *pooling layer* ditunjukkan pada Gambar 2.11.



Sumber: (Yingge et al., 2020)

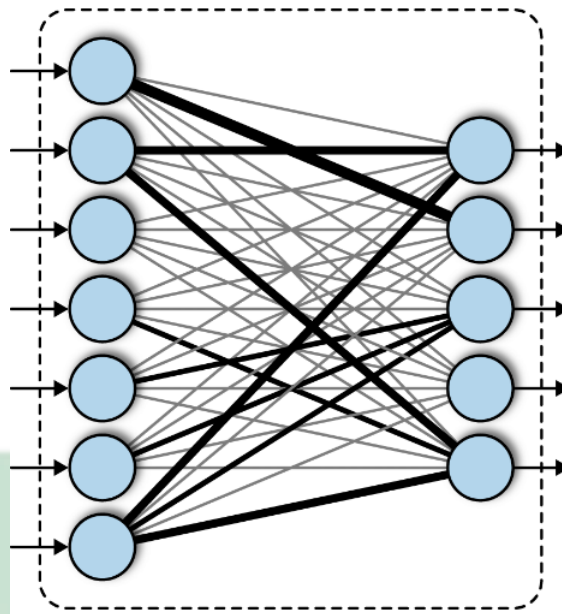
Gambar 2.11 Contoh *Pooling Layer*

2.6.5. Fully Connected Layer

Fully Connected Layer merupakan *layer* yang menghubungkan neuron aktivasi pada *layer* sebelumnya menuju *layer* selanjutnya (Alqumboz and Abu-Naser, 2020). Pada *fully connected layer* ini matriks dengan dimensi x akan diubah menjadi matriks berdimensi satu yang kemudian dapat diklasifikasikan (Lorentius et al., 2019). *Fully connected layer* dapat dicari menggunakan Persamaan 2.8.

$$y_j = b_j + \sum w_{i,j}x_i \quad (2.8)$$

Dimana y_j merupakan hasil output dari *fully connected layer* ke- j , b_j merupakan bias, $w_{(i,j)}$ sebagai bobot jaringan dan x_i adalah input yang diperoleh dari proses *learning feature*. Gambar ilustrasi *fully connected layer* ditunjukkan pada Gambar 2.12.



Sumber: (Kost et al., 2019)

Gambar 2.12 Proses *Fully Connected Layer*

2.6.6. *Softmax Layer*

Softmax layer merupakan jenis fungsi aktivasi yang digunakan pada *output layer* untuk normalisasi jaringan (Alhassan and Zainon, 2021). Hal tersebut dilakukan dengan tujuan mendapatkan nilai probabilitas pada setiap data yang berada diantara interval 0 hingga 1. Softmax dapat dicari menggunakan Persamaan 2.9.

$$p_j(x) = \frac{e^{x_j}}{\sum_{i=1}^k e^{x_j}} \quad (2.9)$$

Dimana $p_j(x)$ menyatakan nilai probabilitas dari x_j yang didapatkan dengan melakukan pembagian antara eksponensial x_j dengan total dari penjumlahan eksponensial x_i pada k (banyak kelas).

2.6.7. Cross Entropy

Cross entropy adalah salah satu bentuk dari *loss function* yang digunakan pada masalah klasifikasi (Wang et al., 2022). *Cross entropy* ini berfungsi untuk mengetahui besar perbedaan antara dua distribusi probabilitas (Li et al., 2020). Lebih mudahnya, *cross entropy* merupakan metode yang dapat mengetahui seberapa jauh perbedaan antara hasil prediksi dengan nilai sesungguhnya. Persamaan 2.10 merupakan persamaan untuk mencari *cross entropy* dimana p merupakan nilai *softmax*, dan t adalah target.

$$D(p, t) = - \sum_j t_j \log p_j \quad (2.10)$$

2.7. Deep Residual Network (ResNet)

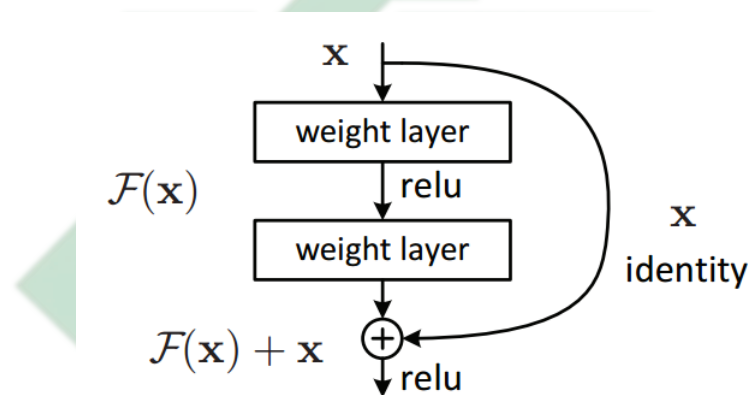
ResNet merupakan arsitektur CNN yang sangat terkenal pada tahun 2015. Pada tahun tersebut, ResNet menjadi pemenang dalam ILSVRC 2015 dan beberapa kompetisi lain. Dalam tahun 2010 hingga 2015 model ResNet memiliki tingkat error terendah jika dibandingkan dengan arsitektur lainnya. Arsitektur ResNet ini dibangun pada tahun 2015 untuk mengatasi masalah *deep learning* yang terbatas pada lapisan tertentu dengan adanya *skip connection* atau *shortcut* (Raman et al., 2021).

ResNet memiliki keunggulan yang tidak dimiliki oleh arsitektur CNN yang lain, yaitu memiliki kinerja yang tetap stabil meskipun kedalaman arsitekturnya semakin rumit. Selain itu, ResNet juga mampu mengatasi masalah mengenai *vanishing gradient problem*, dimana pada masalah tersebut gradien yang terdapat pada layer akhir semakin kecil dan menyebabkan layer selanjutnya tidak terjadi perubahan. Dengan adanya masalah tersebut, ResNet mengusulkan residual block

seperti yang ditunjukkan pada Gambar 2.13 (Basodi et al., 2020). *Residual block* dapat dicari menggunakan persamaan 2.11.

$$y = F(x, W_i + x) \quad (2.11)$$

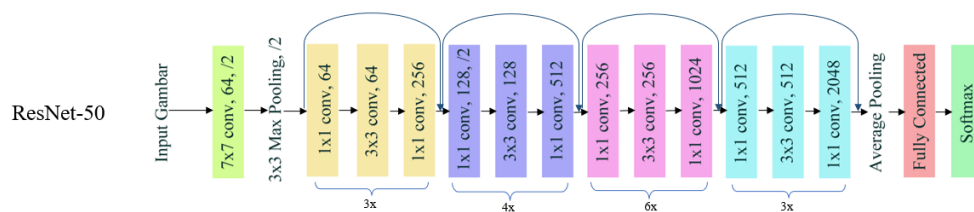
Dimana y merupakan *output layer*, dan x adalah *input layer*. Selain itu, F merupakan fungsi sisa (residual), dan W_i adalah *weight layer*.



Sumber: (Sahoo, 2018)

Gambar 2.13 Residual Block

Residual Block pada ResNet dapat digunakan jika dimensi data *input* dan *output* sama (Sarwinda et al., 2021). Pada jaringan ResNet-50 masing-masing blok terdiri dari 3 *layer*. Lebih jelasnya, arsitektur ResNet-50 divisualisasikan pada 2.14.



Gambar 2.14 Arsitektur ResNet-50

2.8. Confusion Matrix

Confusion matrix adalah metode evaluasi yang digunakan pada sistem klasifikasi yang dapat mengukur tingkat keberhasilan model klasifikasi yang telah dibuat (Koklu and Ozkan, 2020) (Elhoseny et al., 2019). *Confusion matrix* ini digunakan untuk mengetahui akurasi model dengan membandingkan nilai aktual dan nilai prediksi (Zeng, 2020). *Confusion matrix* memiliki ukuran $n \times n$, dimana n menunjukkan jumlah kelas. Tabel *confusion matrix multi class* ditampilkan pada Tabel 2.1 dengan asumsi bahwa C1 = grade 0, C2 = grade 1, C3 = grade 2, C4 = grade 3, dan C5 = grade 4.

Tabel 2.1 *Confusion Matrix* sebanyak 5 kelas

		Predicted Class				
		C1	C2	C3	C4	C5
Actual Class	C1	$x_{1,1}$	$x_{1,2}$	$x_{1,3}$	$x_{1,4}$	$x_{1,5}$
	C2	$x_{2,1}$	$x_{2,2}$	$x_{2,3}$	$x_{2,4}$	$x_{2,5}$
	C3	$x_{3,1}$	$x_{3,2}$	$x_{3,3}$	$x_{3,4}$	$x_{3,5}$
	C4	$x_{4,1}$	$x_{4,2}$	$x_{4,3}$	$x_{4,4}$	$x_{4,5}$
	C5	$x_{5,1}$	$x_{5,2}$	$x_{5,3}$	$x_{5,4}$	$x_{5,5}$

Terdapat beberapa istilah yang digunakan pada *confusion matrix*, yaitu (Chicco et al., 2021):

True Positive (TP) adalah data pada kelas x yang diprediksi benar pada kelas x . TP dapat dicari menggunakan Persamaan 2.12.

$$TP_{all} = \sum_{j=1}^n x_{jj} \quad (2.12)$$

True Negative (TN) adalah data pada kelas y yang diprediksi benar pada kelas

y . TN dapat dicari menggunakan Persamaan 2.13.

$$TN_i = \sum_{j=1, j \neq i}^n \sum_{k=1, k \neq i}^n x_{jk} \quad (2.13)$$

False Positive (FP) adalah data pada kelas y yang diprediksi salah sebagai data pada kelas x . FP dapat dicari menggunakan Persamaan 2.14.

$$FP_i = \sum_{j=1, j \neq i}^n x_{ji} \quad (2.14)$$

False Negative (FN) adalah data pada kelas x yang diprediksi salah sebagai data pada kelas y . FN dapat dicari menggunakan Persamaan 2.15.

$$FN_i = \sum_{j=1, j \neq i}^n x_{ij} \quad (2.15)$$

Berdasarkan nilai yang diberikan oleh TP, TN, FP, dan FN, evaluasi dilakukan menggunakan tiga parameter. Parameter yang digunakan untuk melakukan uji evaluasi menggunakan *confusion matrix* yaitu akurasi, sensitivitas dan spesifisitas. Nilai parameter tersebut dapat dicari menggunakan Persamaan 2.16 sampai Persamaan 2.18 (Muhammad et al., 2020).

$$Akurasi = \frac{TP_{all}}{n_{all}} \times 100\% \quad (2.16)$$

$$Sensitivitas = \sum \frac{TP_{all}}{TP_{all} + FN_i} \times 100\% \quad (2.17)$$

$$\text{Spesifisitas} = \sum \frac{TN_i}{TN_i + FN_i} \times 100\% \quad (2.18)$$

2.9. Penyakit dalam Perspektif Islam

Ditinjau dari segi kedokteran, sakit merupakan keadaan dimana sistem metabolisme tubuh tidak berjalan sebagaimana fungsi awalnya. Berdasarkan perspektif Islam, sakit merupakan bagian pemberian dari Allah yang dapat menguji tingkat keimanan seorang hamba. Segala sesuatu yang diberikan Allah SWT kepada hambanya tentu tidak datang begitu saja tanpa dibarengi adanya maksud dan tujuan. Sejalan dengan hal itu, maka sakit diturunkan oleh Allah juga memiliki makna lain didalamnya, yang salah satunya adalah sebagai ujian untuk hambanya.

Ujian yang Allah turunkan tentu tidak semata-mata untuk menyengsarakan seorang hamba, melainkan juga untuk menguji kesabaran karena setiap manusia yang bernyawa tidak dapat lepas dari ujian yang Allah berikan. Hal tersebut telah dijelaskan dalam Al-Quran surah Al-Baqarah ayat 155 yang berbunyi:

وَلَنَبْلُوَنَّكُمْ بِشَيْءٍ مِّنَ الْخَوْفِ وَالْجُوعِ وَنَقْصٍ مِّنَ الْأَمْوَالِ وَالْأَنْفُسِ
وَالثَّمَرَاتِ ۗ وَبَشِّرِ الصَّابِرِينَ

S U R A B A Y A

Artinya: Dan Kami pasti akan menguji kamu dengan sedikit ketakutan, kelaparan, kekurangan harta, jiwa, dan buah-buahan. Dan sampaikanlah kabar gembira kepada orang-orang yang sabar. Pada ayat tersebut menjelaskan bahwa Allah akan memberikan ujian kepada hambanya dimana salah satunya adalah berupa penyakit. Ayat tersebut juga menjelaskan bahwa akan ada ‘kabar gembira’ bagi orang-orang yang melewati ujian yang Allah berikan dengan sabar. Sehingga, adanya sakit yang menimpa diri seorang hamba dapat diyakini sebagai bentuk ujian dari Allah yang

harus disikapi dengan sabar.

Ujian sakit yang Allah berikan juga dapat mendatangkan kebaikan, seperti sabda Rasulullah SAW pada Hadits yang diriwayatkan oleh Imam Bukhari nomor 5213 yang berbunyi:

حَدَّثَنَا عَبْدُ اللَّهِ بْنُ يُوسُفَ أَخْبَرَنَا مَالِكٌ عَنْ مُحَمَّدِ بْنِ عَبْدِ اللَّهِ بْنِ عَبْدِ الرَّحْمَنِ بْنِ أَبِي صَعْصَعَةَ أَنَّهُ قَالَ سَمِعْتُ سَعِيدَ بْنَ يَسَارٍ أَبَا الْحُبَابِ يَقُولُ سَمِعْتُ أَبَا هُرَيْرَةَ يَقُولُ قَالَ رَسُولُ اللَّهِ صَلَّى اللَّهُ عَلَيْهِ وَسَلَّمَ مَنْ يُرِدُ اللَّهُ بِهِ خَيْرًا يُصِبْ مِنْهُ

Artinya: Telah menceritakan kepada kami [Abdullah bin Yusuf] telah mengabarkan kepada kami [Malik] dari [Muhammad bin Abdullah bin Abdurrahman bin Abu Sha'sha'ah] bahwa dia berkata, saya mendengar [Sa'id bin Yasar Abu Al Hubbab] berkata: saya mendengar [Abu Hurairah] berkata: Rasulullah shallallahu 'alaihi wasallam bersabda: "Barangsiapa dikehendaki Allah kebaikan, maka Dia akan mengujinya" (H.R. Bukhari: 5213). Hadits tersebut menjelaskan bahwasanya kebaikan bisa Allah datangkan kepada seorang hamba yang sedang ditimpa ujian. Oleh karena itu, berbaik sangka pada Allah saat ditimpa sakit merupakan hal yang sangat dianjurkan.

Berbekal kesabaran dan berbaik sangka saat ditimpa rasa sakit juga bukanlah hal yang dibenarkan. Hal tersebut terjadi karena sebagai manusia harus senantiasa berikhtiar dalam segala situasi. Salah satu bentuk ikhtiar dalam menghadapi rasa sakit adalah dengan mencari pengobatan. Sejalan dengan hal tersebut, OA lutut juga harus diusahakan pengobatannya. Sebab setiap kemudharatan yang didatangkan pasti dapat dihilangkan. Hal tersebut sesuai dengan kaidah fikih yang berbunyi:

الضَّرْرُ يُزَالُ

Artinya: "Kemudharatan hendaknya ditolak (dihilangkan) semampunya". Kaidah fikih tersebut menjelaskan bahwa setiap kemudharatan yang datang harus dihilangkan semampunya, baik sebelum atau sesudah terjadi.

Berdasarkan kaidah fikih diatas, maka diagnosa osteoarthritis lutut merupakan salah satu bentuk ikhtiar manusia dalam menghilangkan kemudharatan yang ada. Diagnosa secara dini pada OA lutut bertujuan untuk mengetahui lebih awal kondisi sendi pada lutut yang kemudian berguna untuk meminimalisir keparahan dan dampak yang diberikan.



UIN SUNAN AMPEL
S U R A B A Y A

BAB III

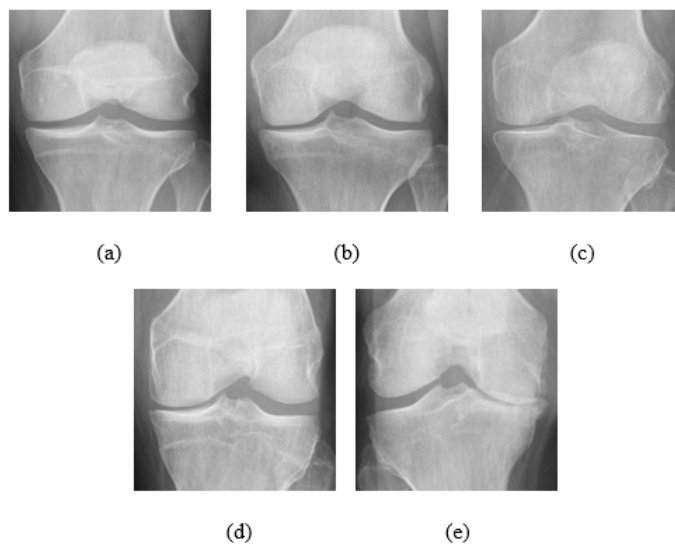
METODE PENELITIAN

3.1. Jenis Penelitian

Penelitian mengenai diagnosis OA lutut ini dapat dikategorikan sebagai penelitian kuantitatif. Penelitian ini menggunakan data numerik yang berupa nilai piksel dari data citra OA lutut. Citra OA lutut tersebut akan dibaca dan diproses oleh komputer sehingga akan berubah menjadi sebuah matriks yang berukuran $M \times N$. Matriks tersebut kemudian dilakukan proses perhitungan menggunakan arsitektur CNN jenis ResNet hingga mendapatkan hasil diagnosa pada masing-masing data citra.

3.2. Sumber Data

Data citra x-ray osteoarthritis lutut diambil melalui Mendeley Dataset (Chen, 2018). Data tersebut terbagi menjadi 5 kelas yang terdiri dari *grade 0*, *grade 1*, *grade 2*, *grade 3*, dan *grade 4*. Total data citra yang diperoleh adalah sebanyak 9458 dengan pembagian 3529 data *grade 0*, 1770 data *grade 1*, 2578 data *grade 2*, 1286 data *grade 3*, dan 295 data *grade 4*. Sampel citra dari masing-masing kelas ditunjukkan pada Gambar 3.1.



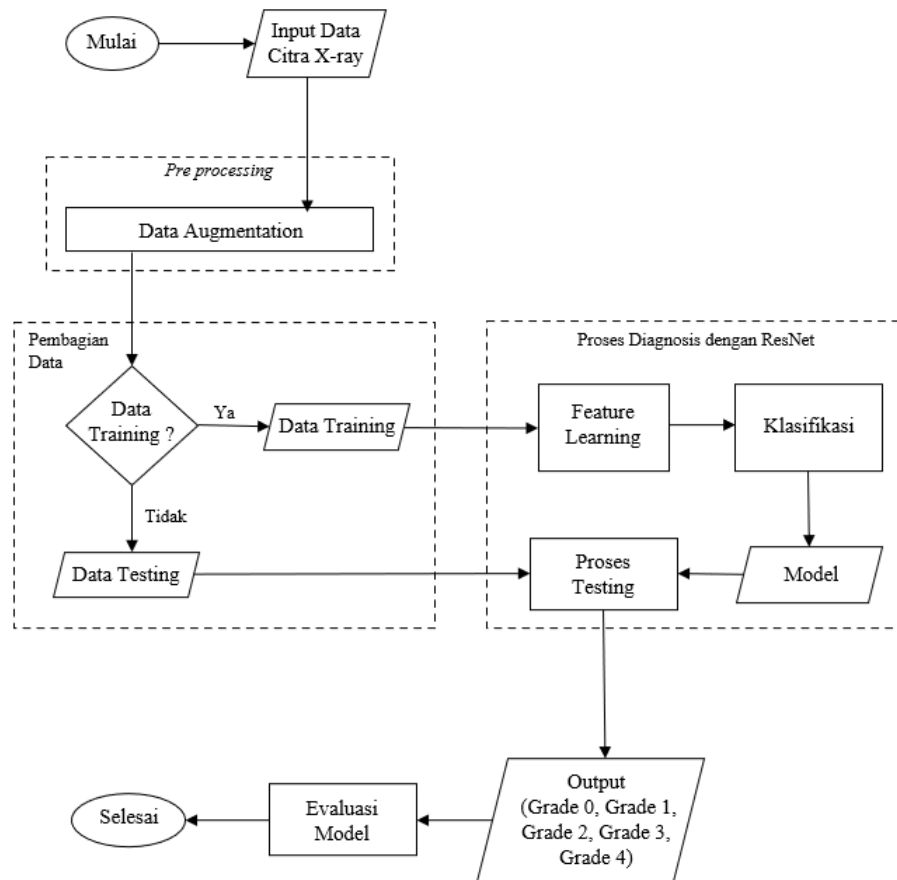
(a) *Grade 0*; (b) *Grade 1*; (c) *Grade 2*; (d) *Grade 3*; (e) *Grade 4*

Gambar 3.1 Citra X-ray Osteoarthritis Lutut

3.3. Kerangka Penelitian

Proses yang dilakukan untuk diagnosis osteoarthritis lutut menggunakan metode *Convolutional Neural Network* (CNN) jenis ResNet terdiri dari beberapa tahapan seperti yang ditunjukkan pada Gambar [3.2](#).

UIN SUNAN AMPEL
S U R A B A Y A



Gambar 3.2 Diagram Alir Penelitian

1. Tahap *preprocessing*, semua data citra osteoarthritis lutut yang berukuran 224×224 telah memenuhi syarat metode ResNet sehingga tidak perlu dilakukan *resizing*. Pada tahap ini, akan dilakukan augmentasi data pada osteoarthritis lutut *grade 4*. Hal tersebut dilakukan karena data pada *grade 4* tidak seimbang dengan data pada *grade* lainnya, sehingga perlu adanya augmentasi dengan melakukan modifikasi kontras dan menghasilkan gambar sejumlah 1475.
2. Tahap pembagian data dilakukan secara *random* pada perbandingan 60% : 40%, 70% : 30%, 80% : 20%, dan 5-fold dengan uji coba *batchsize* (jumlah sampel data pada satu iterasi) dengan nilai 8, 16, 32, dan 64.
3. Proses *feature learning* dilakukan menggunakan algoritma CNN dengan

arsitektur jaringan ResNet 50.

4. Tahap klasifikasi pada *fully connected layer* dengan menggunakan *softmax layer* dapat dilakukan dengan menggunakan persamaan [2.8](#) dan [2.9](#).
5. Tahap terakhir yaitu proses evaluasi menggunakan *confusion matrix multi-class* untuk mendapatkan nilai akurasi, sensitivitas, dan spesifisitas yang masing-masing dapat dicari menggunakan Persamaan [2.16](#), [2.17](#), dan [2.18](#).



UIN SUNAN AMPEL
S U R A B A Y A

BAB IV

HASIL DAN PEMBAHASAN

4.1. *Preprocessing*

Pada penelitian ini, data citra x-ray pada kasus osteoarthritis lutut yang diperoleh dari Mendeley Dataset berjumlah 9458 citra. Dataset tersebut berukuran 224×224 dan akan dilakukan tahap *preprocessing* berupa augmentasi data. Metode augmentasi yang digunakan untuk memperbanyak data osteoarthritis lutut adalah augmentasi kontras. Proses augmentasi ini dilakukan untuk data pada *grade* 4 sebanyak lima kali. Augmentasi dilakukan karena data pada *grade* 4 mengalami *imbalanced* atau ketidakseimbangan. Hal tersebut ditunjukkan dengan data yang hanya berjumlah 295 citra. Oleh karena itu, perlu adanya perbanyak data hingga seimbang dengan data pada *grade* lainnya. Proses augmentasi dilakukan sebanyak 5 kali sehingga menghasilkan data sejumlah 1475. Berdasarkan proses augmentasi ini, data yang akan digunakan untuk diagnosis osteoarthritis bertambah menjadi 10638 citra.

Augmentasi kontras dapat dilakukan dengan mengubah citra menjadi sebuah matriks yang berisi nilai piksel. Sampel citra pada *grade* 4 yang akan diaugmentasi ditunjukkan pada Gambar [4.1](#).



Gambar 4.1 Sampel Cita pada *Grade 4* yang akan diaugmentasi

Citra yang ditunjukkan pada 4.1 tersebut akan diubah menjadi matriks yang berukuran 224×224 seperti berikut:

$$f(x, y) = \begin{bmatrix} 130 & 136 & 131 & 138 & 137 & \dots & 73 & 71 \\ 138 & 140 & 133 & 133 & 138 & \dots & 71 & 70 \\ 140 & 139 & 132 & 129 & 132 & \dots & 72 & 70 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 104 & 105 & 105 & 106 & 106 & \dots & 115 & 112 \\ 104 & 106 & 106 & 105 & 105 & \dots & 113 & 113 \end{bmatrix}$$

Matriks yang telah diperoleh tersebut akan dioperasikan menggunakan Persamaan 2.3 dimana nilai α akan diperoleh secara *random* pada rentang -0.7 hingga 0.7 . Nilai tersebut menunjukkan bahwa kontras citra berada pada *range* -0.7 hingga 0.7 . Sedangkan nilai β yang digunakan pada proses augmentasi ini adalah 1 . Sehingga, apabila didapatkan nilai *random* α sebesar 0.5 dan nilai matriks inputan seperti yang

disebutkan sebelumnya, maka Persamaan 2.3 akan menjadi seperti dibawah ini:

$$g(i, j) = 0.5 \begin{bmatrix} 130 & 136 & 131 & 138 & 137 & \dots & 73 & 71 \\ 138 & 140 & 133 & 133 & 138 & \dots & 71 & 70 \\ 140 & 139 & 132 & 129 & 132 & \dots & 72 & 70 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 104 & 105 & 105 & 106 & 106 & \dots & 115 & 112 \\ 104 & 106 & 106 & 105 & 105 & \dots & 113 & 113 \end{bmatrix} + 1$$

Jika persamaan diatas dilakukan perhitungan, maka akan menghasilkan nilai piksel seperti pada matriks dibawah ini:

$$g(x, y) = \begin{bmatrix} 93 & 94 & 94 & 93 & 91 & \dots & 38 & 37 \\ 94 & 93 & 93 & 93 & 93 & \dots & 37 & 36 \\ 94 & 92 & 94 & 94 & 93 & \dots & 37 & 36 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 39 & 40 & 41 & 42 & 43 & \dots & 59 & 57 \\ 38 & 39 & 41 & 41 & 42 & \dots & 58 & 58 \end{bmatrix}$$

Matriks diatas jika direpresentasikan dalam bentuk gambar akan menjadi gambar seperti yang ditunjukkan pada [4.2](#).



Gambar 4.2 Hasil Augmentasi

Dengan menggunakan langkah kerja yang sama, proses augmentasi dilakukan sebanyak lima kali pada masing-masing data nya.

4.2. Diagnosis OA Lutut Menggunakan Metode CNN Jenis ResNet

Pada Penelitian ini, model ResNet yang digunakan untuk diagnosis OA lutut adalah ResNet-50. Metode ResNet-50 memiliki susunan *layer* seperti yang ditampilkan pada Gambar 2.13. Penjelasan dan contoh perhitungan pada masing-masing *layer* akan dijelaskan sebagai berikut:

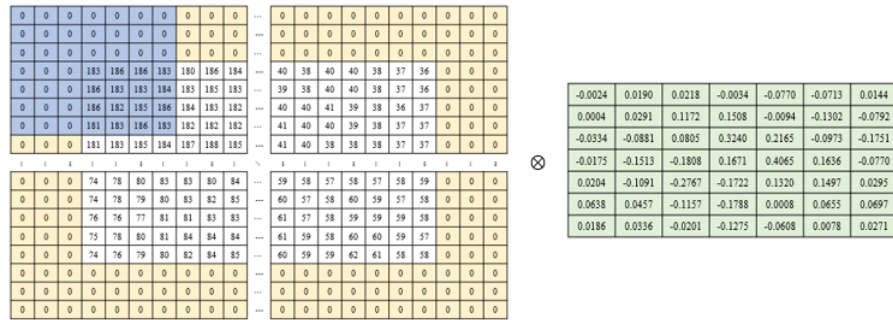
1. Convolution Layer

Proses *convolution layer* dapat dilakukan menggunakan Persamaan 2.4, dimana matriks input (I) berukuran $224 \times 224 \times 3$ dengan nilai *padding* = 3 dan *stride* = 2. Sedangkan matriks kernel (F) yang digunakan berukuran 7×7 sejumlah 64. Matriks tersebut bertujuan untuk melakukan ekstraksi fitur pada bagian deteksi tepi. Matriks kernel ini dibangun secara *random* menggunakan metode *glorot*. Representasi dari matrik *input* (I) dengan *padding* ditampilkan pada Gambar 4.3.

0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0	...
0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0	...
0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0	...
0	0	0	183	186	186	183	180	186	184	...	40	38	40	40	38	37	36	0	0	0	...
0	0	0	186	183	183	184	183	185	183	...	39	38	40	40	38	37	36	0	0	0	...
0	0	0	186	182	185	186	184	183	182	...	40	40	41	39	38	36	37	0	0	0	...
0	0	0	181	183	186	183	182	182	182	...	41	40	40	39	38	37	37	0	0	0	...
0	0	0	181	183	185	184	187	188	185	...	41	40	38	38	38	37	37	0	0	0	...
:	:	:	:	:	:	:	:	:	:	...	:	:	:	:	:	:	:	:	:	:	...
0	0	0	74	78	80	83	83	80	84	...	59	58	57	58	57	58	59	0	0	0	...
0	0	0	74	78	79	80	83	82	85	...	60	57	58	60	59	57	58	0	0	0	...
0	0	0	76	76	77	81	81	83	83	...	61	57	58	59	59	59	58	0	0	0	...
0	0	0	75	78	80	81	84	84	84	...	61	59	58	60	60	59	57	0	0	0	...
0	0	0	74	76	79	80	82	84	85	...	60	59	59	62	61	58	58	0	0	0	...
0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0	...
0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0	...
0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0	...

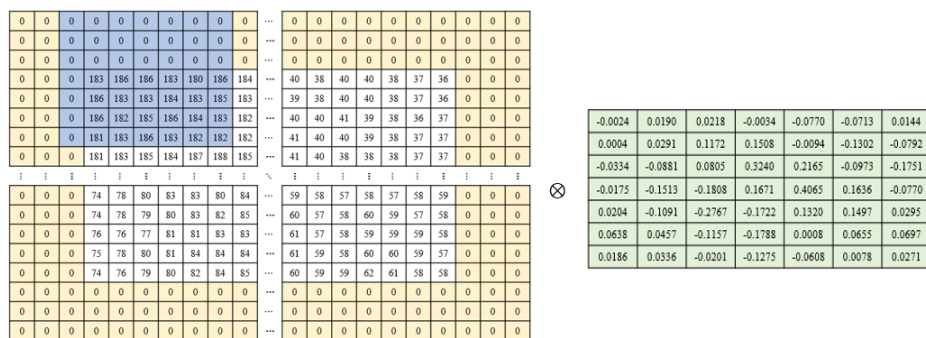
Gambar 4.3 Matriks *Input* dengan *Padding* = 3

Selanjutnya, matriks *input* yang ditunjukkan pada Gambar 4.3 akan dikalikan dengan matriks kernel yang berukuran 7×7 dengan *stride* = 2 menggunakan Persamaan 2.4 seperti yang ditunjukkan pada Gambar 4.4.



Gambar 4.4 Perhitungan *Convolution Layer* dengan *Stride* = 2

$$\begin{aligned}
 Q_j &= \left(\sum_{i=1}^N I_{i,j} \times F_{i,j} \right) \\
 &= (0 \times -0.0024) + (0 \times 0.0190) + (0 \times 0.0218) + (0 \times -0.0034) + \dots \\
 &\quad + (181 \times -0.1275) + (183 \times -0.0608) + (186 \times 0.0078) + (183 \times 0.0271) \\
 &= 111.6535
 \end{aligned}$$



Gambar 4.5 Perhitungan *Convolution Layer* dengan *Stride* = 2

$$\begin{aligned}
 Q_j &= \left(\sum_{i=1}^N I_{i,j} \times F_{i,j} \right) \\
 &= (0 \times -0.0024) + (0 \times 0.0190) + (0 \times 0.0218) + (0 \times -0.0034) + \dots \\
 &\quad + (186 \times -0.1275) + (183 \times -0.0608) + (182 \times 0.0078) + (182 \times 0.0271) \\
 &= -32.7869
 \end{aligned}$$

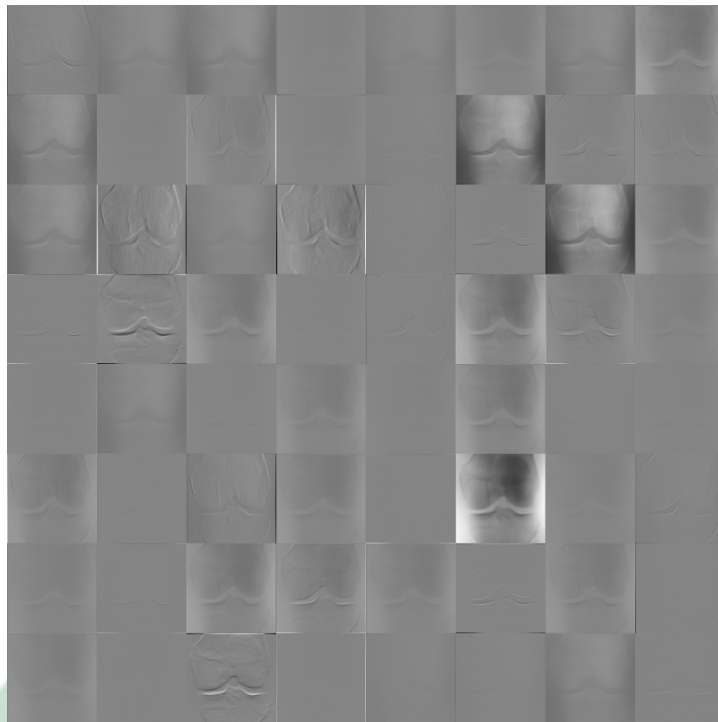
Ukuran dari *feature map* yang dihasilkan pada *convolution layer* dapat dicari menggunakan Persamaan 2.5.

$$\begin{aligned}
 k &= \frac{i - k + 2p}{s} + 1 \\
 &= \frac{224 - 7 + 2(3)}{2} + 1 \\
 &= 112
 \end{aligned}$$

Sehingga, ukuran *feature map* yang dihasilkan adalah 112×112 seperti matriks dibawah ini.

$$g(x, y) = \begin{bmatrix}
 111.6535 & -32.7869 & 33.40 & 28.25 & 26.62 & \dots & 10.32 & 122.80 \\
 -154.92 & -17.13 & -25.07 & -17.77 & -24.00 & \dots & -39.67 & -39.53 \\
 -128.31 & 3.00 & -11.42 & -7.98 & -8.18 & \dots & -22.85 & -38.63 \\
 \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
 -237.35 & -5.68 & -22.49 & -26.97 & -22.39 & \dots & -21.13 & -33.84 \\
 -330.99 & -113.02 & -97.95 & -94.10 & -94.52 & \dots & -61.82 & -112.32
 \end{bmatrix}$$

Perhitungan diatas berulang sejumlah kernel yang digunakan. Pada *convolution layer* pertama pada ResNet jumlah kernel yang digunakan adalah 64, sehingga perhitungan akan dilakukan sebanyak 64 kali dan menghasilkan *feature map* yang berjumlah 64 seperti yang ditampilkan pada Gambar [4.6](#)



Gambar 4.6 Visualisasi *feature map* pada Proses *Convolution Layer*

2. *Batch Normalization*

Berdasarkan hasil *feature map* yang diperoleh pada proses *convolution layer*, akan dinormalisasikan menggunakan metode *Batch Normalization* (BN). Perhitungan normalisasi menggunakan BN dilakukan berdasarkan Persamaan 2.6 dengan nilai $\mu = -0.1512$, $\sigma^2 = 5458.7$ serta nilai parameter $\gamma = 0.1$, $\beta = 0.1$, dan nilai $\epsilon = 0.001$. Sehingga, berdasarkan substitusi nilai-nilai tersebut pada Persamaan 2.6, diperoleh persamaan BN nya adalah sebagai berikut:

$$BN = 0.1 \frac{Q - 0.1512}{\sqrt{(5458.7)^2 + 0.001}} + 0.1$$

Dengan mensubstitusikan nilai Q , maka diperoleh perhitungan sebagai berikut:

$$BN(1, 1, 1) = 0.1 \frac{111.6535 - 0.1512}{\sqrt{(5458.7)^2 + 0.001}} + 0.1$$

$$= -3.21$$

$$BN(1, 2, 1) = 0.1 \frac{-32.7869 - 0.1512}{\sqrt{(5458.7)^2 + 0.001}} + 0.1$$

$$= 3.36$$

$$BN(1, 3, 1) = 0.1 \frac{33.40 - 0.1512}{\sqrt{(5458.7)^2 + 0.001}} + 0.1$$

$$= 2.64$$

hingga

$$BN(112, 112, 1) = 0.1 \frac{-112.32 - 0.1512}{\sqrt{(5458.7)^2 + 0.001}} + 0.1$$

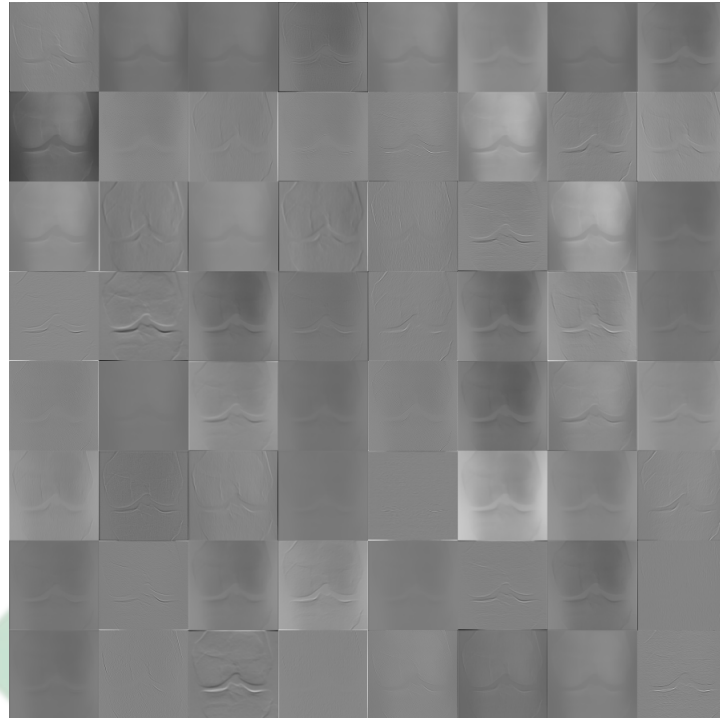
$$= -3.48$$

Sehingga, hasil perhitungan BN pada masing-masing *pixel* adalah:

$$BN(:, :, 1) = \begin{bmatrix} -3.21 & 3.36 & 2.64 & 2.42 & 2.35 & \dots & 1.67 & 6.39 \\ -5.27 & 0.52 & 0.18 & 0.49 & 0.23 & \dots & -0.43 & -0.42 \\ -4.15 & 1.36 & 0.76 & 0.90 & 0.89 & \dots & 0.28 & -0.38 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ -8.73 & 0.99 & 0.29 & 0.10 & 0.30 & \dots & 0.35 & -0.18 \\ -12.66 & -3.51 & -2.88 & -92.71 & -2.73 & \dots & -1.36 & -3.48 \end{bmatrix}$$

Berdasarkan hasil yang diperoleh pada proses normalisasi ini, nilai *feature map* antar *pixel* memiliki jarak yang lebih dekat jika dibandingkan *feature map* pada proses sebelumnya. Jumlah *feature map* yang dihasilkan pada proses BN

ini adalah 64 kali seperti yang ditampilkan pada Gambar 4.7



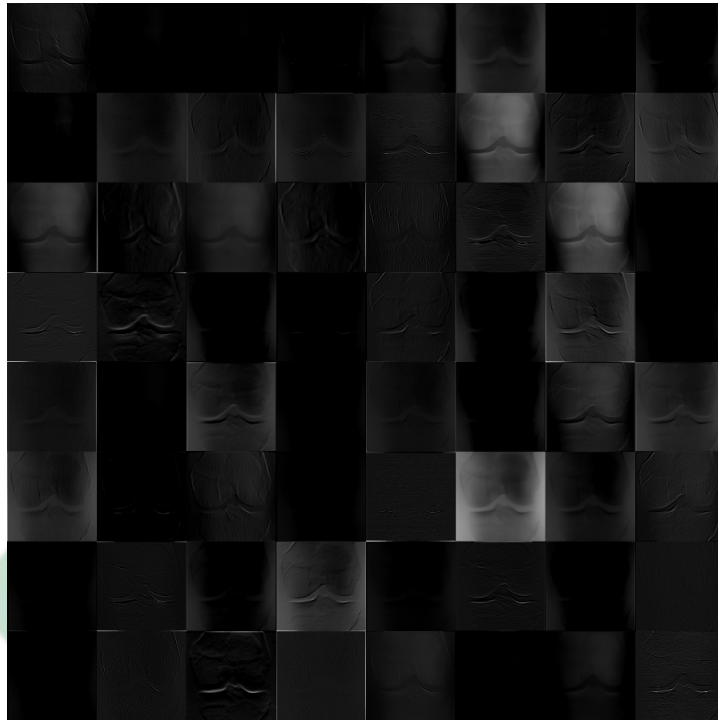
Gambar 4.7 Visualisasi *feature map* pada Proses *Batch Normalization*

3. ReLU

ReLU merupakan fungsi aktivasi yang merubah nilai *feature map* yang negatif menjadi 0. Hasil *feature map* yang diperoleh dari proses BN akan diproses pada *layer* ReLU menggunakan Persamaan 2.7 sehingga menghasilkan hasil sebagai berikut:

$$ReLU(:, :, 1) = \begin{bmatrix} 0 & 3.36 & 2.64 & 2.42 & 2.35 & \dots & 1.67 & 6.39 \\ 0 & 0.52 & 0.18 & 0.49 & 0.23 & \dots & 0 & 0 \\ 0 & 1.36 & 0.76 & 0.90 & 0.89 & \dots & 0.28 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0.99 & 0.29 & 0.10 & 0.30 & \dots & 0.35 & 0 \\ 0 & 0 & 0 & 0 & 0 & \dots & 0 & 0 \end{bmatrix}$$

Hasil yang diperoleh pada proses ReLU adalah *feature map* sebanyak 64 seperti yang ditampilkan pada Gambar 4.8



Gambar 4.8 Visualisasi *feature map* pada Proses ReLU

4. *Max Pooling*

Max pooling merupakan *layer* yang berfungsi untuk mengurangi dimensi data secara spasial. Operasi *max pooling* dilakukan menggunakan jumlah *stride* = 2. ukuran hasil keluaran dari operasi *max pooling* dapat dicari menggunakan Persamaan 2.5.

$$\begin{aligned}
 k &= \frac{i - k + 2p}{s + 1} \\
 &= \frac{112 - 7 + 2(3)}{2} + 1 \\
 &= 56
 \end{aligned}$$

Sehingga, *feature map* yang dihasilkan dari layer ReLU akan dilakukan operasi *max pooling* dengan *stride* = 2 seperti yang ditunjukkan pada Gambar 4.9.

0	3.36	2.64	2.42	2.35	2.36	1.99	2.25	2.63	2.38
0	0.52	0.18	0.49	0.23	0.40	0.27	0.10	0.56	0.81
0	1.36	0.76	0.90	0.89	0.89	0.82	0.79	0.79	1.01
0	1.36	0.95	0.81	0.85	0.79	0.79	0.79	0.74	0.79
0	1.23	0.98	0.97	0.83	0.96	0.82	0.74	0.78	0.69
0	1.23	0.64	1.04	0.82	0.82	0.83	0.78	0.78	0.99
0	1.31	0.72	0.90	0.89	0.95	0.65	0.80	0.79	0.88
0	1.25	0.68	0.84	0.77	0.86	0.86	0.76	0.76	0.58
0	1.30	0.70	0.74	0.86	0.81	0.60	0.71	0.71	0.83

Gambar 4.9 Operasi Max Pooling dengan Stride = 2

Berdasarkan Gambar 4.9, maka hasil operasi *max pooling* yang didapatkan adalah:

$$\begin{aligned}
 Mp(1, 1, 1) &= MAX(0, 3.36, 2.64, 0, 0.52, 0.18, 0, 1.36, 0.76) \\
 &= 3.36
 \end{aligned}$$

Selanjutnya, dengan menggunakan *stride* = 2, maka proses *max pooling* kedua dapat dilihat pada Gambar 4.10.

0	3.36	2.64	2.42	2.35	2.36	1.99	2.25	2.63	2.38
0	0.52	0.18	0.49	0.23	0.40	0.27	0.10	0.56	0.81
0	1.36	0.76	0.90	0.89	0.89	0.82	0.79	0.79	1.01
0	1.36	0.95	0.81	0.85	0.79	0.79	0.79	0.74	0.79
0	1.23	0.98	0.97	0.83	0.96	0.82	0.74	0.78	0.69
0	1.23	0.64	1.04	0.82	0.82	0.83	0.78	0.78	0.99
0	1.31	0.72	0.90	0.89	0.95	0.65	0.80	0.79	0.88
0	1.25	0.68	0.84	0.77	0.86	0.86	0.76	0.76	0.58
0	1.30	0.70	0.74	0.86	0.81	0.60	0.71	0.71	0.83

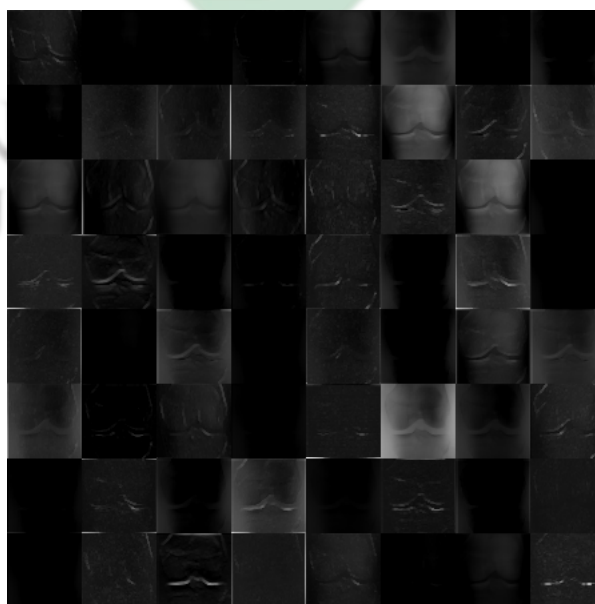
Gambar 4.10 Operasi Max Pooling dengan Stride = 2

$$\begin{aligned}
 Mp(1, 2, 1) &= MAX(2.64, 2.42, 2.35, 0.18, 0.49, 0.23, 0.76, 0.90, 0.89) \\
 &= 2.64
 \end{aligned}$$

Proses tersebut berlanjut hingga seluruh *pixel* dan menghasilkan *output* matriks sebagai berikut:

$$Mp(:, :, 1) = \begin{bmatrix} 3.36 & 2.64 & 2.42 & 2.36 & 2.63 & \dots & 1.84 & 6.39 \\ 1.37 & 1.37 & 0.90 & 0.89 & 1.01 & \dots & 0.41 & 0.58 \\ 1.37 & 1.37 & 1.04 & 0.96 & 1.02 & \dots & 0.51 & 0.58 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 1.09 & 1.09 & 0.43 & 0.41 & 0.45 & \dots & 0.57 & 0.68 \\ 1.09 & 1.09 & 0.43 & 0.36 & 0.32 & \dots & 0.57 & 0.57 \end{bmatrix}$$

Proses *max pooling* akan menghasilkan 64 *feature map* sebagaimana yang ditampilkan pada Gambar 4.11.



Gambar 4.11 Operasi *Max Pooling* dengan *Stride = 2*

5. Addition Layer

Addition layer merupakan *layer* yang menjadi pembeda antara ResNet dengan arsitektur CNN yang lainnya. Pada *addition layer* ini, terjadi proses penjumlahan antara *feature map inputan* dan *feature map* yang sudah diproses.

Contoh proses pada *addition layer* ditunjukkan pada Gambar 4.12.

The diagram illustrates the addition of two feature maps. The first input map (green) has values: 0.41, -0.45, 0.11, 0.34, ..., 0.24, 0.09, 0.36, 0.22, ..., 0.39, 0.20, 0.41, 0.44, ..., 0.15, 0.22, 0.52, 0.42, ..., 0.34, 0.27, 0.07, 0.13, 0.40, 0.34, ... The second input map (orange) has values: -0.04, -0.13, -0.15, -0.15, ..., 0.22, 0.09, 0.04, 0.04, ..., 0.23, 0.09, 0.04, 0.04, ..., 0.23, 0.09, 0.06, 0.05, ..., 0.24, 0.10, 0.06, 0.06, ..., -0.11, 0.02, 0.12, 0.28, ..., 0.13, 0.26, 0.14, 0.27, ..., 0.15, 0.30, ... The output map (blue) has values: 0.36, -0.58, -0.04, 0.18, ..., 0.12, 0.12, 0.19, -0.23, 0.08, 0.16, ..., 0.48, 0.49, 0.20, 0.19, 0.52, 0.47, ..., 0.53, 0.46, 0.38, 0.31, 0.58, 0.47, ..., 0.55, 0.71, 0.30, 0.24, 0.47, 0.40, ..., 0.49, 0.57, ...

Gambar 4.12 Operasi *Addition Layer*

6. Average Pooling

Average pooling merupakan salah satu bagian dari *pooling layer*. *Average pooling* ini berada pada *layer* akhir *feature learning*. Proses ini akan mengubah matriks menjadi suatu vektor, hal tersebut menyebabkan pada metode ResNet tidak memerlukan lagi proses *flatten*. Operasi *average pooling* dilakukan dengan *stride* = 2 seperti yang ditunjukkan pada Gambar 4.13.

0	3.36	2.64	2.42	2.35	2.36	1.99	2.25	2.63	2.38
0	0.52	0.18	0.49	0.23	0.40	0.27	0.10	0.56	0.81
0	1.36	0.76	0.90	0.89	0.89	0.82	0.79	0.79	1.01
0	1.36	0.95	0.81	0.85	0.79	0.79	0.79	0.74	0.79
0	1.23	0.98	0.97	0.83	0.96	0.82	0.74	0.78	0.69
0	1.23	0.64	1.04	0.82	0.82	0.83	0.78	0.78	0.99
0	1.31	0.72	0.90	0.89	0.95	0.65	0.80	0.79	0.88
0	1.25	0.68	0.84	0.77	0.86	0.86	0.76	0.76	0.58
0	1.30	0.70	0.74	0.86	0.81	0.60	0.71	0.71	0.83

Gambar 4.13 Operasi *Average Pooling* dengan *Stride* = 2

Berdasarkan Gambar 4.13, maka hasil operasi *average pooling* yang

didapatkan adalah:

$$\begin{aligned} Av(1, 1, 1) &= AVERAGE(0, 3.36, 2.64, 0, 0.52, 0.18, 0, 1.36, 0.76) \\ &= 0.98 \end{aligned}$$

Selanjutnya, dengan menggunakan *stride* = 2, maka proses *average pooling* kedua dapat dilihat pada Gambar 4.14.

0	3.36	2.64	2.42	2.35	2.36	1.99	2.25	2.63	2.38
0	0.52	0.18	0.49	0.23	0.40	0.27	0.10	0.56	0.81
0	1.36	0.76	0.90	0.89	0.89	0.82	0.79	0.79	1.01
0	1.36	0.95	0.81	0.85	0.79	0.79	0.79	0.74	0.79
0	1.23	0.98	0.97	0.83	0.96	0.82	0.74	0.78	0.69
0	1.23	0.64	1.04	0.82	0.82	0.83	0.78	0.78	0.99
0	1.31	0.72	0.90	0.89	0.95	0.65	0.80	0.79	0.88
0	1.25	0.68	0.84	0.77	0.86	0.86	0.76	0.76	0.58
0	1.30	0.70	0.74	0.86	0.81	0.60	0.71	0.71	0.83

Gambar 4.14 Operasi *Average Pooling* dengan *Stride* = 2

$$\begin{aligned} Av(1, 2, 1) &= AVERAGE(2.64, 2.42, 2.35, 0.18, 0.49, 0.23, 0.76, 0.90, 0.89) \\ &= 1.21 \end{aligned}$$

Hasil dari operasi *average pooling* menghasilkan vektor dengan ukuran 1×2048 seperti yang ditunjukkan pada Gambar 4.15.

0.04
0
0
0
0.01
0
0.19
⋮
⋮
0
0

Gambar 4.15 Hasil Average Pooling

7. *Fully Connected Layer*

Pada *fully connected layer*, akan dilakukan proses klasifikasi. Proses ini dapat dilakukan menggunakan Persamaan 2.8 yaitu:

$$y_j = b_j + \sum w_{i,j}x_i$$

UIN SUNAN AMPEL
S U R A B A Y A

dimana:

$$b = \begin{bmatrix} 0.0003 \\ -0.0024 \\ -0.0234 \\ -0.0317 \\ -0.0817 \end{bmatrix}$$

$$w = \begin{bmatrix} 0.0580 & -0.0314 & 0.0675 & 0.0128 & -0.1315 & \dots & -0.0619 \\ 0.0269 & -0.0055 & -0.0912 & 0.0585 & -0.0729 & \dots & -0.0316 \\ -0.0819 & 0.0276 & -0.0678 & -0.0907 & 0.0071 & \dots & 0.0568 \\ -0.1782 & -0.0910 & 0.0307 & 0.0117 & 0.0053 & \dots & -0.0485 \\ -0.0811 & 0.0606 & -0.0341 & -0.0089 & -0.0426 & \dots & -0.0304 \end{bmatrix}$$

$$x = \begin{bmatrix} 0.6402 \\ 0.0348 \\ 0.1374 \\ 0 \\ 0 \\ 0.1257 \\ \dots \\ 0.1010 \end{bmatrix}$$

Berdasarkan nilai yang didapatkan diatas, akan dimasukkan ke Persamaan 2.8

dan didapatkan nilai sebagai berikut:

$$\begin{aligned}
 y_1 &= b_1 + \sum_{i=1}^{2048} w_{(i,1)}x_i \\
 &= 0.0003 + (0.0580 \times 0.6402) + (-0.0314 \times 0.0348) + (0.0675 \times 0.1374) \\
 &\quad + (0.0128 \times 0) + (-0.1315 \times 0) + \dots + (-0.0619 \times 0.1010) \\
 &= 5.2395
 \end{aligned}$$

$$\begin{aligned}
 y_2 &= b_2 + \sum_{i=1}^{2048} w_{(i,2)}x_i \\
 &= -0.0024 + (0.0269 \times 0.6402) + (-0.0055 \times 0.0348) + (-0.0912 \times 0.1374) \\
 &\quad + (0.0585 \times 0) + (-0.0729 \times 0) + \dots + (-0.0316 \times 0.1010) \\
 &= -9.3254
 \end{aligned}$$

$$\begin{aligned}
 y_3 &= b_3 + \sum_{i=1}^{2048} w_{(i,3)}x_i \\
 &= -0.0234 + (-0.0819 \times 0.6402) + (0.0276 \times 0.0348) + (-0.0678 \times 0.1374) \\
 &\quad + (-0.0907 \times 0) + (0.0071 \times 0) + \dots + (0.0568 \times 0.1010) \\
 &= -6.4322
 \end{aligned}$$

$$\begin{aligned}
 y_4 &= b_4 + \sum_{i=1}^{2048} w_{(i,4)}x_i \\
 &= -0.0317 + (-0.1782 \times 0.6402) + (-0.0910 \times 0.0348) + (0.0307 \times 0.1374) \\
 &\quad + (0.0117 \times 0) + (0.0053 \times 0) + \dots + (-0.0485 \times 0.1010) \\
 &= -15.5778
 \end{aligned}$$

$$\begin{aligned}
 y_5 &= b_5 + \sum_{i=1}^{2048} w_{(i,5)}x_i \\
 &= -0.0817 + (-0.0811 \times 0.6402) + (0.0606 \times 0.0348) + (-0.0341 \times 0.1374) \\
 &\quad + (-0.0089 \times 0) + (-0.0426 \times 0) + \dots + (-0.0304 \times 0.1010) \\
 &= -13.1601
 \end{aligned}$$

8. *Softmax Layer*

Softmax layer merupakan fungsi aktivasi yang digunakan untuk mencari nilai probabilitas pada masing-masing *inputan*. *Softmax layer* ini akan menentukan kelas klasifikasi menggunakan hasil yang diperoleh pada *fully connected layer*.



Probabilitas tersebut dapat dicari menggunakan Persamaan 2.9.

$$\begin{aligned}
 p_1(x) &= \frac{e^{x_1}}{\sum_{i=1}^5 e^{x_i}} \\
 &= \frac{e^{5.2395}}{e^{5.2395} + e^{-9.3524} + e^{-6.4322} + e^{-15.5778} + e^{-13.1601}} \\
 &= \frac{188.58}{188.58 + 0.00008 + 0.001609 + 0.0000002 + 0.0000002} \\
 &= 0.999991032
 \end{aligned}$$

$$\begin{aligned}
 p_2(x) &= \frac{e^{x_2}}{\sum_{i=1}^5 e^{x_i}} \\
 &= \frac{e^{-9.3524}}{e^{5.2395} + e^{-9.3524} + e^{-6.4322} + e^{-15.5778} + e^{-13.1601}} \\
 &= \frac{0.00008}{188.58 + 0.00008 + 0.001609 + 0.0000002 + 0.0000002} \\
 &= 0.00000047
 \end{aligned}$$

$$\begin{aligned}
 p_3(x) &= \frac{e^{x_3}}{\sum_{i=1}^5 e^{x_i}} \\
 &= \frac{e^{-6.4322}}{e^{5.2395} + e^{-9.3524} + e^{-6.4322} + e^{-15.5778} + e^{-13.1601}} \\
 &= \frac{0.001609}{188.58 + 0.00008 + 0.001609 + 0.0000002 + 0.0000002} \\
 &= 0.00000853
 \end{aligned}$$

$$\begin{aligned}
 p_4(x) &= \frac{e^{x_4}}{\sum_{i=1}^5 e^{x_i}} \\
 &= \frac{e^{-15.5778}}{e^{5.2395} + e^{-9.3524} + e^{-6.4322} + e^{-15.5778} + e^{-13.1601}} \\
 &= \frac{0.0000002}{188.58 + 0.00008 + 0.001609 + 0.0000002 + 0.0000002} \\
 &= 0.0000000009
 \end{aligned}$$

$$\begin{aligned}
 p_5(x) &= \frac{e^{x_5}}{\sum_{i=1}^5 e^{x_i}} \\
 &= \frac{e^{-13.1601}}{e^{5.2395} + e^{-9.3524} + e^{-6.4322} + e^{-15.5778} + e^{-13.1601}} \\
 &= \frac{0.0000002}{188.58 + 0.00008 + 0.001609 + 0.0000002 + 0.0000002} \\
 &= 0.0000000102
 \end{aligned}$$

9. *Cross Entropy*

Cross entropy merupakan *layer* yang digunakan untuk menghitung nilai *error* model berdasarkan data keluaran dari *softmax layer*. Data yang diperoleh dari *softmax layer* akan dibandingkan dengan data aktual menggunakan Persamaan 2.10 sehingga untuk mendapatkan nilai *error* nya.

$$\begin{aligned}
 D(p, t) &= - \sum_{j=1}^5 t_j \log p_j \\
 &= -((1 \times \log(0.999991032)) + (0 \times \log(0.00000047)) + (0 \times \log(0.00000853)) \\
 &\quad + (0 \times \log(0.0000000009)) + (0 \times \log(0.0000000102))) \\
 &= 0.0000089680
 \end{aligned}$$

4.3. Analisis Hasil Diagnosis OA Lutut Sebelum Augmentasi

Dengan model ResNet-50 tersebut, akan dilakukan beberapa uji coba untuk mencari model yang optimal pada data sebelum augmentasi, yaitu uji coba *batch size* dan pembagian data. Uji coba *batch size* yang diterapkan pada penelitian ini yaitu 8, 16, 32, dan 64. Sedangkan uji coba pembagian data yang digunakan adalah 70% : 30%, 80% : 20%, 90% : 10% dan *k-fold* = 5. Hasil dari berbagai uji coba tanpa menggunakan *k-fold* ditunjukkan pada Tabel 4.5.

Tabel 4.1 Hasil Uji Coba Sebelum Augmentasi

TABEL UJI COBA					
Pembagian Data	Batch	Sens	Akurasi	Spes	Waktu
70% : 30%	8	66.02	67.11	90.83	409m 12s
	16	65.97	65.60	90.68	351m 57s
	32	67.90	64.61	90.36	315m 24s
	64	66.63	67.79	91.01	308m 40s
80% : 20%	8	67.00	67.71	89.91	567m 09s
	16	68.32	68.09	91.11	523m 46s
	32	70.61	71.09	92.75	481m 16s
	64	70.58	70.42	92.10	458m 07s
90% : 10%	8	66.32	69.77	89.08	608m 06s
	16	65.43	68.46	90.33	597m 39s
	32	68.75	70.92	92.61	532m 13s
	64	68.01	69.81	91.54	499m 17s

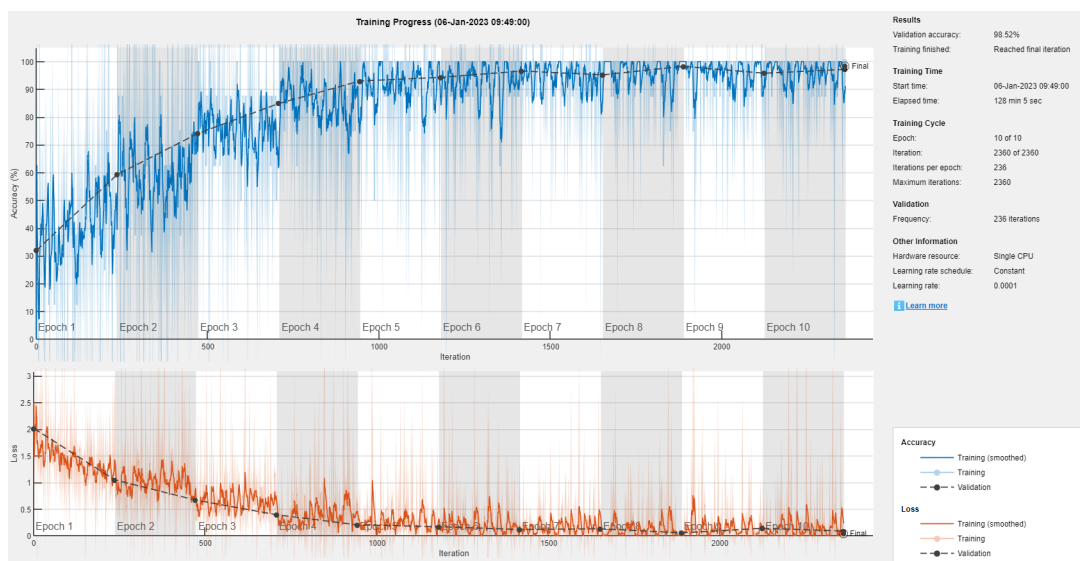
Selain uji coba yang disajikan pada Tabel 4.5, uji coba lainnya yang dilakukan untuk mendapatkan hasil terbaik dilakukan menggunakan *k-fold cross validation*. Hasil uji coba tersebut disajikan pada Tabel 4.6.

UIN SUNAN AMPEL
S U R A B A Y A

Tabel 4.2 Hasil Uji Coba Sebelum Augmentasi

TABEL UJI COBA					
K	Batch	Sens	Akurasi	Spes	Waktu
K=1	8	88.47	89.26	90.43	134m 32s
	16	89.51	88.09	91.78	114m 54s
	32	93.89	94.73	95.45	111m 31s
	64	89.06	90.28	90.78	104m 07s
K=2	8	89.84	93.68	91.24	139m 55s
	16	90.51	91.62	92.45	120m 12s
	32	95.83	93.04	97.64	120m 57s
	64	90.54	91.17	92.76	104m 38s
K=3	8	90.02	90.88	92.32	136m 12s
	16	92.11	92.45	94.57	120m 58s
	32	95.79	94.86	96.79	120m 01s
	64	93.39	94.32	92.04	100m 49s
K=4	8	91.71	93.86	93.74	139m 22s
	16	94.42	94.55	96.56	136m 15s
	32	98.12	98.52	99.60	128m 05s
	64	93.00	92.68	93.32	106m 42s
K=5	8	95.62	90.52	94.47	140m 56s
	16	96.56	89.80	90.55	121m 28s
	32	92.07	94.07	92.32	105m 30s
	64	96.91	93.32	95.68	101m 44s

Berdasarkan hasil uji coba yang telah ditunjukkan pada Tabel 4.5 dan Tabel 4.6, didapatkan model yang paling optimal adalah pada uji coba pembagian data menggunakan *k-fold cross validation* dengan jumlah *batch size* 64. Hasil tersebut diperoleh dari proses *training* seperti pada grafik yang ditunjukkan pada Gambar 4.7.

Tabel 4.3 Grafik Proses *Training* pada Model Optimal

Pada uji coba tersebut, nilai akurasi yang didapatkan yaitu 98.52%, dan sensitivitas yang diperoleh adalah 98.12%. Hasil tersebut diperoleh berdasarkan perhitungan pada *confusion matrix* menggunakan Persamaan 2.16 hingga 2.18. Tabel *confusion matrix* yang didapatkan pada uji coba tersebut ditampilkan pada Tabel 4.8

Tabel 4.4 Hasil *Confusion Matrix* pada Model Optimal

		Predicted Class				
		C1	C2	C3	C4	C5
Actual Class	C1	701	3	2	0	0
	C2	4	347	9	0	0
	C3	1	4	505	3	0
	C4	0	0	0	254	2
	C5	0	0	0	0	57

Berdasarkan hasil *confusion matrix* yang ditunjukkan pada Tabel 4.16, nilai akurasi, sensitivitas, dan spesifisitas secara berturut-turut akan dihitung

menggunakan Persamaan 2.16, 2.17, dan 2.18.

$$Akurasi = \frac{701+347+505+254+57}{5} \times 100\%$$

$$= 98.52\%$$

$$Sensitivitas = \frac{\frac{701}{701+4+1+0+0} + \frac{347}{3+347+4+0+0} + \frac{505}{2+9+505+0+0} + \frac{254}{0+0+3+254+0} + \frac{57}{0+0+0+2+57}}{5} \times 100\%$$

$$= 98.12\%$$

$$Spesifisitas = \frac{\frac{1181}{1181+5} + \frac{1525}{1525+13} + \frac{1368}{1368+8} + \frac{1633}{1633+2} + \frac{1833}{1833}}{5} \times 100\%$$

$$= 99.60\%$$

4.4. Analisis Hasil Diagnosis OA Lutut Setelah Augmentasi

Data citra x-ray OA lutut yang diperoleh akan dilakukan augmentasi untuk mengetahui pengaruh augmentasi pada kinerja sistem. Uji coba yang dilakukan untuk mencari hasil terbaik dilakukan sama dengan percobaan pada data sebelum augmentasi yaitu uji coba *batch size* dan pembagian data. Uji coba *batch size* yang diterapkan pada penelitian ini yaitu 8, 16, 32, dan 64. Sedangkan uji coba pembagian data yang digunakan adalah 70% : 30%, 80% : 20%, 90% : 10% dan *k-fold* = 5. Hasil dari berbagai uji coba tanpa menggunakan *k-fold* pada data setelah augmentasi ditunjukkan pada Tabel 4.5.

Tabel 4.5 Hasil Uji Coba Sesudah Augmentasi

TABEL UJI COBA					
Pembagian Data	Batch	Sens	Akurasi	Spes	Waktu
70% : 30%	8	70,91	69,76	92,19	515m 41s
	16	70,74	72,33	92,58	390m 22s
	32	69,05	70,58	92,23	340m 39s
	64	69,54	69,09	91,95	678m 11s
80% : 20%	8	59,90	64,98	91,18	547m 41s
	16	69,38	68,82	92,00	463m 6s
	32	69,09	70,96	92,24	411m 3s
	64	70,33	71,51	92,44	568m 56s
90% : 10%	8	70,30	68,71	92,03	622m 42s
	16	70,42	72,67	92,63	496m 37s
	32	72,06	71,57	92,68	461m 32s
	64	69,18	71,66	92,44	497m 15s

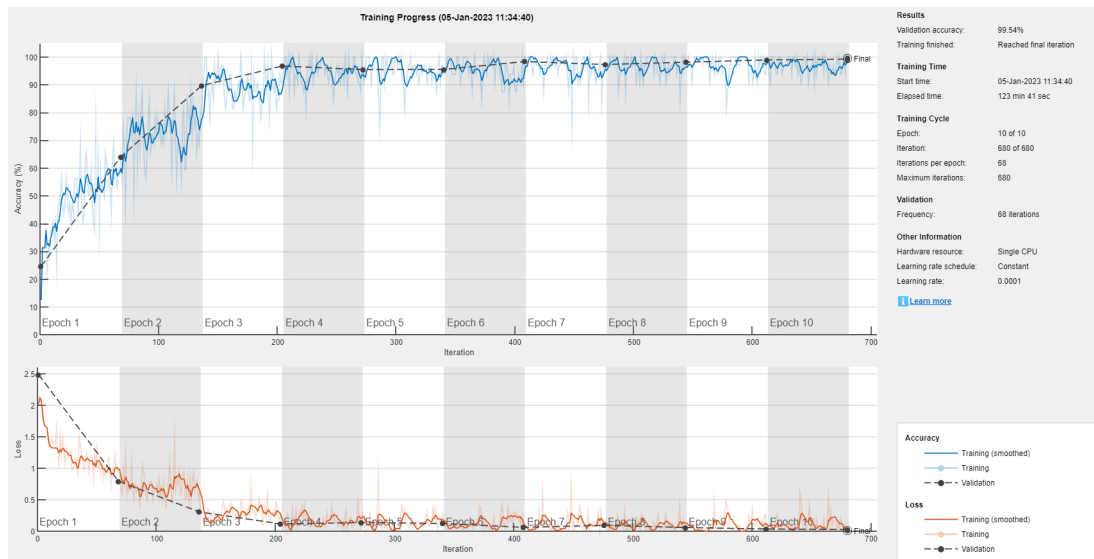
Selain uji coba yang disajikan pada Tabel 4.5, uji coba lainnya yang dilakukan untuk mendapatkan hasil terbaik dilakukan menggunakan *k-fold cross validation*. Hasil uji coba tersebut disajikan pada Tabel 4.6.

UIN SUNAN AMPEL
S U R A B A Y A

Tabel 4.6 Hasil Uji Coba Sesudah Augmentasi

TABEL UJI COBA					
K	Batch	Sens	Akurasi	Spes	Waktu
K=1	8	98.94	99.13	99.76	146m 56s
	16	97.80	97.94	99.48	126m 08s
	32	99.61	99.54	99.89	123m 41s
	64	99.00	98.99	99.75	116m 02s
K=2	8	98.97	98.99	99.74	151m 33s
	16	99.44	99.41	99.85	132m 40s
	32	99.31	98.99	99.75	132m 23s
	64	98.51	98.31	99.59	116m 28s
K=3	8	98.84	98.72	99.69	148m 27s
	16	99.40	99.36	99.84	132m 57s
	32	97.02	97.16	99.16	132m 26s
	64	96.79	96.98	99.12	112m 56s
K=4	8	98.57	98.76	99.69	151m 48s
	16	97.33	97.71	99.41	138m 57s
	32	99.09	99.13	99.77	137m 05s
	64	98.93	98.94	99.72	118m 43s
K=5	8	97.33	97.89	99.46	152m 17s
	16	98.52	98.44	99.59	133m 17s
	32	98.94	99.08	99.75	127m 50s
	64	97.02	97.16	99.16	109m 07s

Berdasarkan hasil uji coba yang telah ditunjukkan pada Tabel 4.5 dan Tabel 4.6, didapatkan model yang paling optimal adalah pada uji coba pembagian data menggunakan *k-fold cross validation* dengan jumlah *batch size* 32. Hasil tersebut diperoleh dari proses *training* seperti pada grafik yang ditunjukkan pada Gambar 4.7.

Tabel 4.7 Grafik Proses *Training* pada Model Optimal

Pada uji coba tersebut, nilai akurasi yang didapatkan yaitu 99.54%, dan sensitivitas yang diperoleh adalah 99.61%. Hasil tersebut diperoleh berdasarkan perhitungan pada *confusion matrix* menggunakan Persamaan 2.16 hingga 2.18. Tabel *confusion matrix* yang didapatkan pada uji coba tersebut ditampilkan pada Tabel 4.8

Tabel 4.8 Hasil *Confusion Matrix* pada Model Optimal

		Predicted Class				
		C1	C2	C3	C4	C5
Actual Class	C1	704	1	1	0	0
	C2	2	353	2	0	0
	C3	0	0	508	0	0
	C4	0	0	4	258	0
	C5	0	0	0	0	354

Berdasarkan hasil *confusion matrix* yang ditunjukkan pada Tabel 4.16, nilai akurasi, sensitivitas, dan spesifisitas secara berturut-turut akan dihitung

menggunakan Persamaan 2.16, 2.17, dan 2.18.

$$Akurasi = \frac{704+353+508+258+354}{2187} \times 100\%$$

$$= 99.54\%$$

$$Sensitivitas = \frac{\frac{704}{704+2+0+0+0} + \frac{353}{1+353+0+0+0} + \frac{508}{1+2+508+4+0} + \frac{258}{0+0+0+258+0} + \frac{354}{0+0+0+0+354}}{5} \times 100\%$$

$$= 99.61\%$$

$$Spesifisitas = \frac{\frac{1479}{1479+2} + \frac{1829}{1829+4} + \frac{1672}{1672+0} + \frac{1925}{1925+4} + \frac{706}{706}}{5} \times 100\%$$

$$= 99.89\%$$

Berdasarkan hasil yang diperoleh pada data sebelum dan sesudah augmentasi, bahwa hasil terbaik didapatkan pada percobaan data setelah adanya proses augmentasi. Hal tersebut menunjukkan bahwa augmentasi data merupakan proses penting yang dapat mengatasi masalah *inbalanced data* dan mampu meningkatkan hasil akurasi.

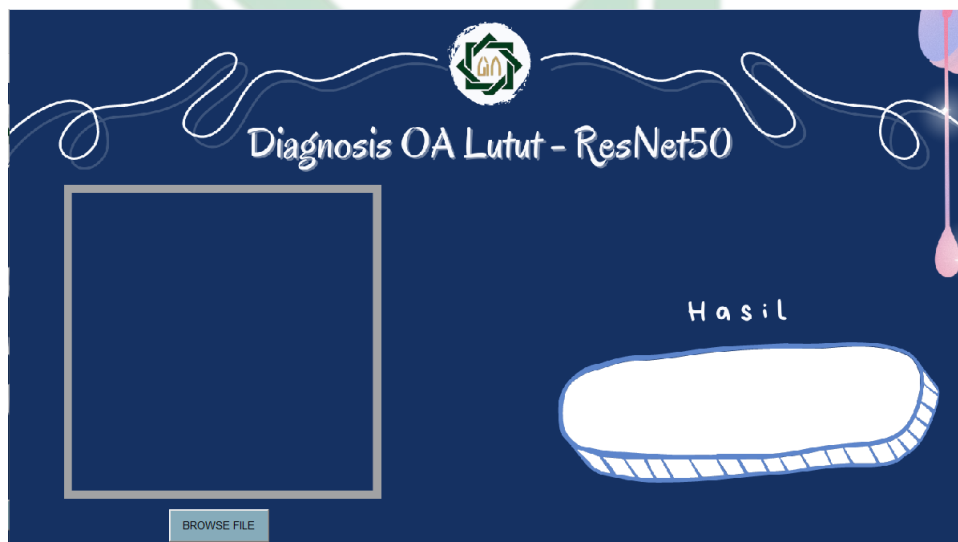
4.5. Aplikasi Diagnosis OA Lutut

Model terbaik yang diperoleh dari hasil uji coba sebelumnya akan digunakan sebagai model utama untuk diagnosis OA lutut. Model tersebut akan diterapkan pada aplikasi yang dibuat untuk memudahkan dalam diagnosis OA lutut. Aplikasi yang dibangun dapat mendiagnosis penderita OA lutut maupun tidak. Penderita OA lutut juga lebih detail dibagi lagi menjadi empat tingkatan berbeda, yaitu *grade 1*, *grade 2*, *grade 3*, dan *grade 4*. Tampilan awal pada aplikasi tersebut ditampilkan pada Gambar [4.16](#).



Gambar 4.16 Tampilan Awal Aplikasi Diagnosis OA Lutut

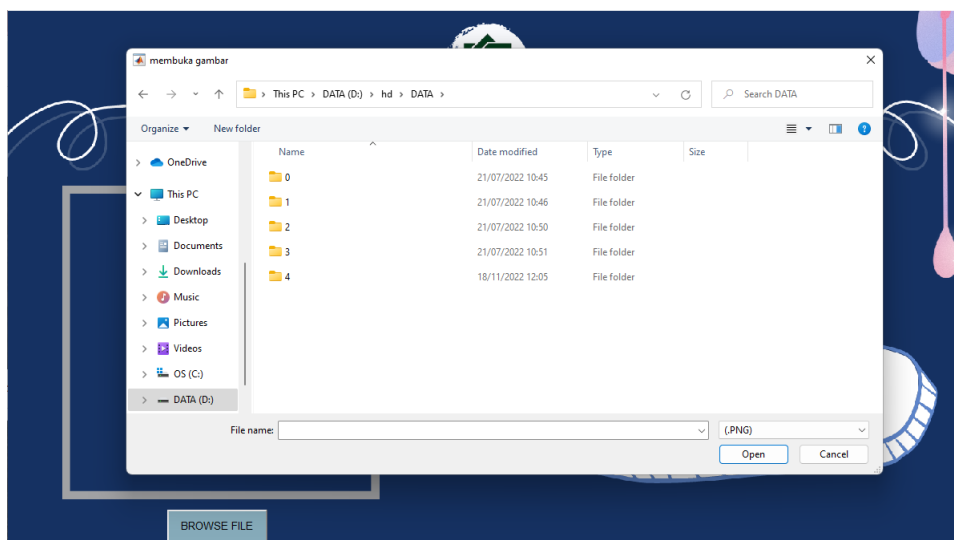
Setelah tampilan awal aplikasi keluar, selanjutnya untuk menjalankan aplikasi dapat dilakukan dengan menekan tombol *start*. Tampilan yang disajikan setelah menekan tombol tersebut ditunjukkan pada Gambar 4.17.



Gambar 4.17 Tampilan Aplikasi Diagnosis OA Lutut

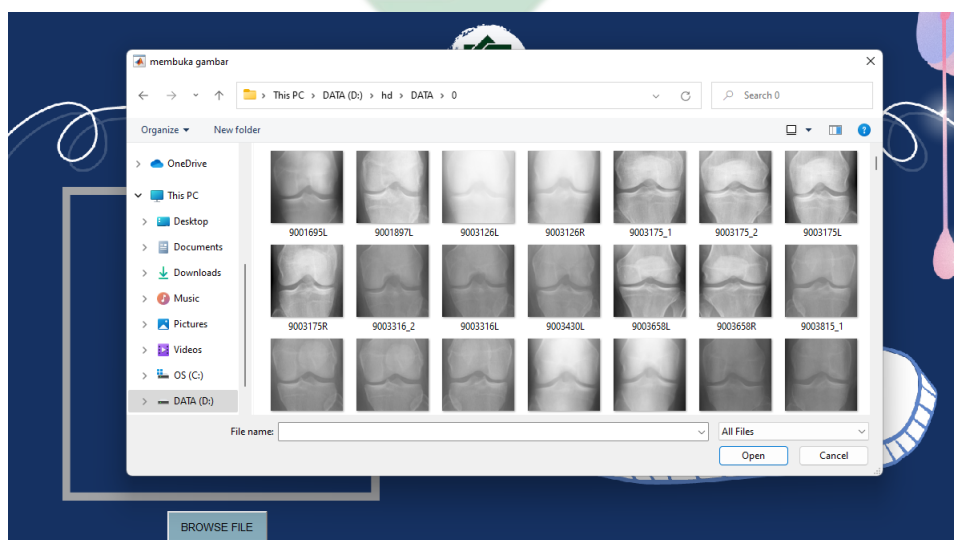
Proses untuk melakukan diagnosis OA lutut diawali dengan memasukkan data yang akan di diagnosis. Proses tersebut dapat dilakukan pada aplikasi dengan menekan tombol *browse file* yang ada di bagian bawah *frame*. Setelah itu, akan

dihadapkan pilihan folder kelas data seperti yang ditunjukkan pada Gambar 4.18.



Gambar 4.18 Tampilan Aplikasi Diagnosis OA Lutut

Nama folder yang ditampilkan pada Gambar 4.18 menunjukkan urutan *grade*. Apabila ingin melakukan diagnosis terhadap pasien normal atau yang tidak terkena OA lutut, maka dapat dilakukan dengan memilih folder dengan nama '0' sebagaimana ditampilkan pada Gambar 4.19.



Gambar 4.19 Tampilan Aplikasi Diagnosis OA Lutut

Selanjutnya, dapat dilakukan dengan memilih salah satu *file* citra pada *grade*

0 untuk dilakukan diagnosis. Apabila data citra sudah dipilih, maka data tersebut akan otomatis masuk ke aplikasi dan berada di tengah *frame* yang telah dibuat. Hasil tersebut ditunjukkan pada Gambar 4.20.



Gambar 4.20 Tampilan Aplikasi Diagnosis OA Lutut

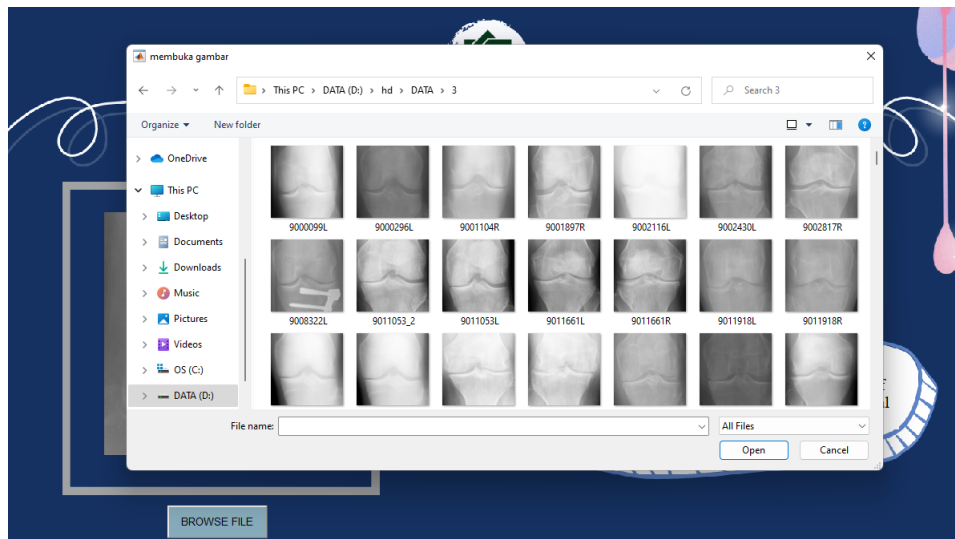
Langkah terakhir yang dilakukan pada aplikasi diagnosis OA lutut ini adalah dengan menekan tombol *check*. Tombol tersebut berfungsi untuk melihat hasil akhir dari sistem diagnosis citra yang telah diinputkan. Setelah tombol tersebut ditekan, maka akan muncul hasil diagnosis yang diletakkan pada *space* putih dalam tampilan sebagaimana ditunjukkan pada Gambar 4.21.



Gambar 4.21 Tampilan Aplikasi Diagnosis OA Lutut

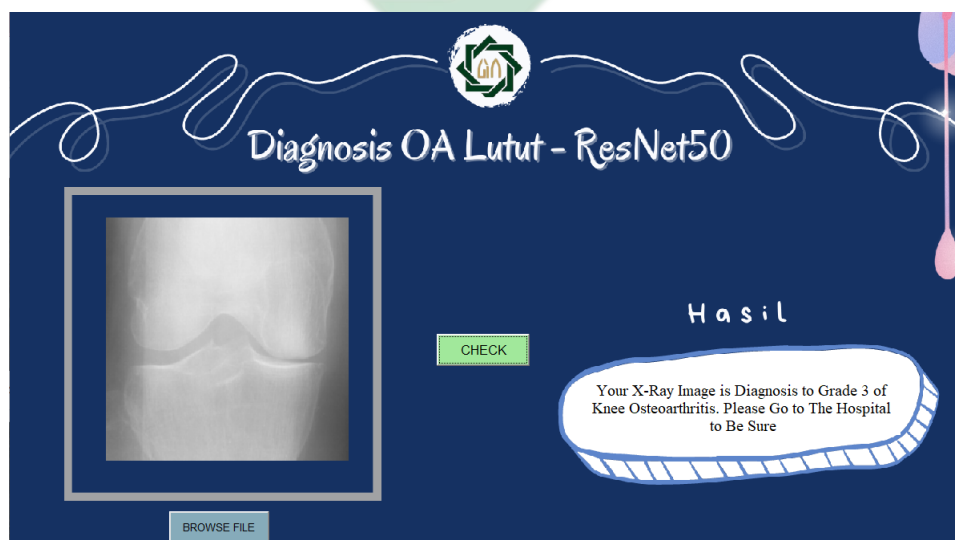
Berdasarkan Gambar 4.21, sistem menunjukkan bahwa data yang diinputkan merupakan data dari non penderita OA lutut atau pasien normal. Pasien tersebut juga disarankan untuk tetap menjaga kesehatan agar tetap terhindar dari OA lutut. Selain itu, aplikasi juga dapat dilakukan untuk diagnosis kembali pada kelas yang berbeda dengan menekan ulang tombol *browse file*. Apabila aplikasi tersebut ingin digunakan untuk melakukan diagnosis pada penderita OA lutut dengan *grade 3*, maka dapat memilih folder '3' seperti Gambar 4.22.

UIN SUNAN AMPEL
S U R A B A Y A



Gambar 4.22 Tampilan Aplikasi Diagnosis OA Lutut

Setelah itu, data pada *grade 3* dimasukkan dengan cara memilih salah satu *file* citra yang akan dimasukkan pada aplikasi. Apabila data yang dipilih sudah dimasukkan pada aplikasi, maka akan dihasilkan *output* dengan menekan tombol *check*. Hasil akhir dari aplikasi untuk diagnosis OA lutut pada *grade 3* ditunjukkan pada Gambar 4.23.



Gambar 4.23 Tampilan Aplikasi Diagnosis OA Lutut

Berdasarkan hasil tersebut, didapatkan diagnosis akhir bahwa pasien

mengalami OA lutut dengan *grade* 3. Selanjutnya, pasien disarankan untuk melakukan pemeriksaan lanjutan di rumah sakit untuk meminimalisir dampak yang lebih parah.

4.6. Sistem Diagnosis dalam Perspektif Islam

Sistem diagnosis osteoarthritis lutut merupakan salah satu bentuk aplikasi yang dibuat untuk memudahkan masyarakat dalam melakukan diagnosis. Proses tersebut dapat dikatakan sebagai alternatif baru yang dapat digunakan semua kalangan. Sebagai manusia yang senantiasa taat kepada Allah SWT., hal tersebut sudah selayaknya dilakukan karena sebagai manusia harus senantiasa bermanfaat bagi manusia lainnya. Hal tersebut telah diajarkan dalam agama Islam sebagaimana dibuktikan dengan adanya Hadits yang diriwayatkan oleh Ath-Thabrani nomor 5787.

عَنْ جَابِرٍ، رَضِيَ اللَّهُ عَنْهُمَا، قَالَ: قَالَ رَسُولُ اللَّهِ صَلَّى اللَّهُ عَلَيْهِ وَسَلَّمَ: خَيْرُ النَّاسِ أَنْفَعُهُمْ لِلنَّاسِ

Artinya: Dari sahabat Jabir radhiyallahu‘anhuma bercerita bahwa Rasulullah shallallahu ‘alaihi wasallam bersabda: "Sebaik-baik manusia adalah yang paling bermanfaat bagi manusia."

Berdasarkan hadits yang diriwayatkan oleh Ath-Thabrani tersebut, dijelaskan bahwa manusia yang baik dihadapan Allah SWT adalah manusia yang bermanfaat bagi sesamanya. Salah satu cara agar dapat bermanfaat bagi sesama adalah dengan membangun sistem diagnosis OA lutut ini, karena sistem ini dapat membantu sesama dalam melakukan pengecekan kondisi lutut. Diagnosis OA lutut ini dapat dilakukan hanya jika manusia mau berusaha. Usaha tersebut tidak selalu memberikan hasil yang sempurna, namun bisa saja mendekati sempurna. Seperti pada penelitian ini,

hasil sensitivitas dari berbagai uji coba yang dilakukan untuk diagnosis OA lutut yakni sebesar 99.61%. Hasil tersebut memang belum dapat dikatakan sempurna, namun sudah mendekati nilai sempurna. Meskipun demikian, usaha yang telah dilakukan untuk membangun sistem ini merupakan usaha yang positif dan bernilai kebaikan. Hal tersebut telah dijelaskan dalam firman Allah pada QS. At-Taubah ayat 105 yang berbunyi:

وَقُلْ اَعْمَلُوا فَسَيَرَى اللّٰهُ عَمَلَكُمْ وَرَسُولُهُ وَالْمُؤْمِنُونَ وَسَتُرَدُّونَ
اِلَىٰ عِلْمِ الْغَيْبِ وَالشَّهَادَةِ فَيُنبِّئُكُمْ بِمَا كُنْتُمْ تَعْمَلُونَ ۝

Artinya: Dan katakanlah, “Bekerjalah kamu, maka Allah akan melihat pekerjaanmu, begitu juga Rasul-Nya dan orang-orang mukmin, dan kamu akan dikembalikan kepada (Allah) Yang Mengetahui yang gaib dan yang nyata, lalu diberitakan-Nya kepada kamu apa yang telah kamu kerjakan.” Ayat tersebut menjelaskan bahwa segala sesuatu yang kita usahakan akan Allah berikan balasan yang setimpal kepada diri kita. Oleh karena itu, usaha dalam membangun sistem diagnosis OA lutut ini juga merupakan suatu pekerjaan yang Allah sukai.

Sistem diagnosis OA lutut ini dibangun dengan ikhtiar mencari jalan keluar bagi para penyintas OA lutut. Jalan keluar berupa kesembuhan sangat diharapkan dengan adanya sistem ini karena dengan diketahuinya penyakit sejak awal, maka akan mengurangi dampak yang lebih parah. Sistem ini dibuat karena penyakit OA lutut ini diyakini memiliki obat penyembuhan sebagaimana sejalan dengan Hadits yang diriwayatkan oleh Imam Muslim nomor 4084 yang berbunyi:

حَدَّثَنَا هَارُونُ بْنُ مَعْرُوفٍ وَأَبُو الطَّاهِرِ وَأَحْمَدُ بْنُ عِيسَى قَالُوا
 حَدَّثَنَا ابْنُ وَهْبٍ أَخْبَرَنِي عَمْرُو وَهُوَ ابْنُ الْحَارِثِ عَنْ عَبْدِ رَبِّهِ
 بْنِ سَعِيدٍ عَنْ أَبِي الزُّبَيْرِ عَنْ جَابِرٍ عَنْ رَسُولِ اللَّهِ صَلَّى اللَّهُ عَلَيْهِ
 وَسَلَّمَ أَنَّهُ قَالَ لِكُلِّ دَاءٍ دَوَاءٌ فَإِذَا أُصِيبَ دَوَاءُ الدَّاءِ بَرَأَ بِإِذْنِ اللَّهِ
 عَزَّ وَجَلَّ

Artinya: Telah menceritakan kepada kami Harun bin Ma'rif dan Abu Ath Thahir serta Ahmad bin 'Isa mereka berkata; Telah menceritakan kepada kami Ibnu Wahb; Telah mengabarkan kepadaku 'Amru, yaitu Ibnu al-Harits dari 'Abdu Rabbih bin Sa'id dari Abu Az Zubair dari Jabir dari Rasulullah shallallahu 'alaihi wasallam, beliau bersabda: "Setiap penyakit ada obatnya. Apabila ditemukan obat yang tepat untuk suatu penyakit, akan sembuhlah penyakit itu dengan izin Allah 'azza wajalla."

Hadits yang diriwayatkan oleh Imam Muslim nomor 4084 menjadi bukti bahwa penyakit OA lutut yang digunakan dalam penelitian ini merupakan suatu penyakit yang pasti memiliki obat. Oleh karena itu, obat tersebut harus dicari yang salah satu langkah pencarian obatnya adalah dengan melakukan diagnosis. Obat atau kesembuhan bagi penderita OA lutut harus dicari karena seorang hamba yang memiliki tubuh sehat dan kuat memiliki keutamaan di hadapan Allah seperti Hadits yang diriwayatkan oleh Imam Muslim nomor 2664 yang berbunyi:

الْمُؤْمِنُ الْقَوِيُّ خَيْرٌ وَأَحَبُّ إِلَى اللَّهِ مِنَ الْمُؤْمِنِ الضَّعِيفِ وَفِي كُلِّ
 خَيْرٍ احْرَصْ عَلَى مَا يَنْفَعُكَ وَاسْتَعِنْ بِاللَّهِ وَلَا تَعْجِزْ وَإِنْ أَصَابَكَ
 شَيْءٌ فَلَا تَقُلْ لَوْ أَنِّي فَعَلْتُ كَذَا وَكَذَا. وَلَكِنْ قُلْ قَدَرُ اللَّهِ
 وَمَا شَاءَ فَعَلَ فَإِنَّ لَوْ تَفْتَحُ عَمَلَ الشَّيْطَانِ

Artinya: "Mukmin yang kuat lebih baik dan lebih dicintai oleh Allah daripada mukmin yang lemah. Namun, keduanya tetap memiliki kebaikan. Bersemangatlal

atas hal-hal yang bermanfaat bagimu. Minta tolonglah pada Allah, jangan engkau lemah. Jika engkau tertimpa suatu musibah, maka janganlah engkau katakan: 'Seandainya aku lakukan demikian dan demikian.' Akan tetapi hendaklah kau katakan: 'Ini sudah jadi takdir Allah. Setiap apa yang telah Dia kehendaki pasti terjadi.' Karena perkataan law (seandainya) dapat membuka pintu setan."

Berdasarkan hadits yang telah disebutkan sebelumnya, didapatkan bahwa orang mukmin yang senantiasa menjaga kesehatan pada dirinya, akan dicintai oleh Allah SWT. Hal tersebut dikarenakan pada diri seorang mukmin yang sehat, mampu melakukan lebih banyak amal sholeh, baik untuk dirinya sendiri maupun sekitar. Oleh karena itu, apabila pada diri seorang hamba terdapat penyakit, maka harus senantiasa berikhtiar dalam mencari kesembuhan. Sebagaimana penyakit OA lutut ini, harus dicari obatnya untuk mencari kesembuhan diri sehingga dapat meningkatkan kedudukan di mata Allah SWT.



UIN SUNAN AMPEL
S U R A B A Y A

BAB V

PENUTUP

5.1. Kesimpulan

Berdasarkan hasil yang diperoleh dari beberapa uji coba untuk diagnosis osteoarthritis lutut menggunakan metode *Convolutional Neural Network* jenis ResNet ini dapat diambil beberapa kesimpulan, diantaranya adalah:

1. Hasil terbaik yang didapatkan pada penelitian ini adalah pada uji coba pembagian data menggunakan *k-fold cross validation* dengan jumlah *batchsize* 32. Pada uji coba tersebut, data yang digunakan adalah data hasil augmentasi dan diperoleh nilai sensitivitas 99.61%. Nilai sensitivitas merupakan parameter utama pada proses evaluasi, karena sensitivitas sangat diperhatikan pada medis.
2. Proses augmentasi merupakan proses yang penting dalam penelitian ini, karena dapat mengatasi masalah *imbalanced data*. Hal tersebut mempengaruhi hasil sensitivitas yang didapatkan. Pada data sebelum augmentasi, nilai sensitivitas terbaik yang dihasilkan adalah 92.79%, sedangkan setelah augmentasi mampu menghasilkan 99.61%.

5.2. Saran

Pada penelitian diagnosis OA lutut menggunakan metode ResNet ini masih memiliki banyak kekurangan yang perlu disempurnakan untuk penelitian selanjutnya. Beberapa hal yang perlu dikembangkan untuk mendapatkan hasil yang lebih baik

dalam penelitian selanjutnya adalah:

1. Melakukan lebih banyak uji coba untuk menghasilkan hasil yang lebih optimal. Uji coba yang dilakukan dapat berupa uji coba arsitektur ResNet, *optimizer*, jumlah *epoch*, *learning rate*, dan lain sebagainya.
2. Metode ResNet mampu memberikan hasil yang baik untuk diagnosis OA lutut. Namun, hal tersebut membutuhkan waktu yang relatif lama. Oleh karena itu, perlu dilakukan adanya modifikasi metode ResNet seperti dengan menggabungkan dengan metode lainnya seperti dengan AlexNet untuk mempersingkat waktu komputasi tanpa mengurangi kinerja sistem.



UIN SUNAN AMPEL
S U R A B A Y A

DAFTAR PUSTAKA

- Agrawal, S., Patel, M., and Sinhal, A. (2021). An enhance security of the color image using asymmetric rsa algorithm. pages 279–286. Springer.
- Alhassan, A. M. and Zainon, W. M. N. W. (2021). Brain tumor classification in magnetic resonance image using hard swish-based relu activation function-convolutional neural network. *Neural Computing and Applications*, 33:9075–9087.
- Allaam, M. R. R. and Wibowo, A. T. (2021). Klasifikasi genus tanaman anggrek menggunakan metode convolutional neural network (cnn). *eProceedings of Engineering*, 8.
- Alqumboz, M. N. A. and Abu-Naser, S. S. (2020). Avocado classification using deep learning. *International Journal of Academic Engineering Research (IJAER)*, 3.
- Basodi, S., Ji, C., Zhang, H., and Pan, Y. (2020). Gradient amplification: An efficient way to train deep neural networks. *Big Data Mining and Analytics*, 3:196–207.
- Bhatt, D., Patel, C., Talsania, H., Patel, J., Vaghela, R., Pandya, S., Modi, K., and Ghayvat, H. (2021). Cnn variants for computer vision: History, architecture, application, challenges and future scope. *Electronics*, 10:2470.
- Bilal, M. A., Ji, Y., Wang, Y., Akhter, M. P., and Yaqub, M. (2022). Early earthquake detection using batch normalization graph convolutional neural network (bngcnn). *Applied Sciences*, 12:7548.

- Bork, F., Stratmann, L., Enssle, S., Eck, U., Navab, N., Waschke, J., and Kugelmann, D. (2019). The benefits of an augmented reality magic mirror system for integrated radiology teaching in gross anatomy. *Anatomical sciences education*, 12:585–598.
- Budhiman, A., Suyanto, S., and Arifianto, A. (2019). Melanoma cancer classification using resnet with data augmentation. pages 17–20. IEEE.
- Chapala, H. and Sujatha, B. (2020). Resnet: detection of invasive ductal carcinoma in breast histopathology images using deep learning. pages 60–67. IEEE.
- Chauhan, R., Ghanshala, K. K., and Joshi, R. C. (2018). Convolutional neural network (cnn) for image detection and recognition. pages 278–282. IEEE.
- Chen, P. (2018). Knee osteoarthritis severity grading dataset.
- Chicco, D., Tötsch, N., and Jurman, G. (2021). The matthews correlation coefficient (mcc) is more reliable than balanced accuracy, bookmaker informedness, and markedness in two-class confusion matrix evaluation. *BioData mining*, 14:1–22.
- Chow, Y. Y. and Chin, K.-Y. (2020). The role of inflammation in the pathogenesis of osteoarthritis. *Mediators of inflammation*, 2020.
- Cui, A., Li, H., Wang, D., Zhong, J., Chen, Y., and Lu, H. (2020). Global, regional prevalence, incidence and risk factors of knee osteoarthritis in population-based studies. *EClinicalMedicine*, 29:100587.
- Damanik, R. (2022). Image detection edge image using canny edge algorithm. *Journal of Artificial Intelligence and Engineering Applications (JAIEA)*, 1:248–255.
- Ding, B., Qian, H., and Zhou, J. (2018). Activation functions and their characteristics in deep neural networks. pages 1836–1841. IEEE.

- Dougherty, E. R. (2020). *Digital image processing methods*. CRC Press.
- El-Kahlout, M. I. and Abu-Naser, S. S. (2020). Peach type classification using deep learning. *International Journal of Academic Engineering Research (IJAER)*, 3.
- Elhoseny, M., Shankar, K., and Uthayakumar, J. (2019). Intelligent diagnostic prediction and classification system for chronic kidney disease. *Scientific reports*, 9:1–14.
- Elswah, D. K., Elnakib, A. A., and din Moustafa, H. E. (2020). Automated diabetic retinopathy grading using resnet. pages 248–254.
- Feng, B., Zhou, H., Li, G., Zhang, Y., Sood, K., and Yu, S. (2021). Enabling machine learning with service function chaining for security enhancement at 5g edges. *IEEE Network*, 35:196–201.
- Haq, D. Z. (2020). Klasifikasi citra kanker kulit menggunakan convolutional neural network model googlenet.
- Hsu, H. and Siwiec, R. M. (2018). Knee osteoarthritis.
- Hussain, A. and Khuntela, A. (2020). Semantic segmentation of brain tumor from mri images and svm classification using glcm features. pages 38–43.
- Jang, S., Lee, K., and Ju, J. H. (2021). Recent updates of diagnosis, pathophysiology, and treatment on osteoarthritis of the knee. *International journal of molecular sciences*, 22:2619.
- Jogin, M., Madhulika, M. S., Divya, G. D., Meghana, R. K., and Apoorva, S. (2018). Feature extraction using convolution neural networks (cnn) and deep learning. pages 2319–2323. IEEE.

- Jones, I. A., Togashi, R., Wilson, M. L., Heckmann, N., and Vangsness, C. T. (2019). Intra-articular treatment options for knee osteoarthritis. *Nature Reviews Rheumatology*, 15:77–90.
- Kan, H. S., Chan, P. K., Chiu, K. Y., Yan, C. H., Yeung, S. S., Ng, Y. L., Shiu, K. W., and Ho, T. (2019). Non-surgical treatment of knee osteoarthritis. *Hong Kong Medical Journal*, 25:127.
- Kellgren, J. and Lawrence, J. S. (1957). Radiological assessment of osteo-arthrosis. *Annals of the Rheumatic Diseases*, 16:494 – 502.
- Khaleghi, M. (2020). New arthritis foundation guidelines on cbd use could be first of many more to come. *Alternative therapies in health and medicine*, 26:8–11.
- Kim, D. H., Kim, S., Yoon, J., and Lee, Y. (2020). Are there harmful effects of preoperative mild lateral or patellofemoral degeneration on the outcomes of open wedge high tibial osteotomy for medial compartmental osteoarthritis? *Orthopaedic Journal of Sports Medicine*, 8:232596712092748.
- Koklu, M., Cinar, I., and Taspinar, Y. S. (2021). Classification of rice varieties with deep learning methods. *Computers and Electronics in Agriculture*, 187:106285.
- Koklu, M. and Ozkan, I. A. (2020). Multiclass classification of dry beans using computer vision and machine learning techniques. *Computers and Electronics in Agriculture*, 174:105507.
- Kost, A., Altabey, W., Noori, M., and Awad, T. (2019). Applying neural networks for tire pressure monitoring systems. *Structural Durability & Health Monitoring*, 13:247–266.
- Kurnianto, D. (2013). Empat tipe dasar citra digital.

- Lai, Y. and Donsig, A. (2018). Using geometric habits of mind to connect geometry from a transformation perspective to graph transformations and abstract algebra.
- Lespasio, M. J., PiuZZi, N. S., Husni, M. E., Muschler, G. F., Guarino, A. J., and Mont, M. A. (2017). Knee osteoarthritis: a primer. *The Permanente Journal*, 21.
- Li, J. (2022). Application of nanotubes combined with ethnic sports rehabilitation therapy in the treatment of patients with knee arthritis. *Journal of Nanomaterials*, 2022.
- Li, Q., Hu, Y., Zhou, X., Liu, S., Han, Q., and Cheng, L. (2020). Role of oral bacteria in the development of oral squamous cell carcinoma. *Cancers*, 12:2797.
- Lorentius, C. A., Gunadi, K., and Tjondrowiguno, A. N. (2019). Pengenalan aksara jawa dengan menggunakan metode convolutional neural network. *Jurnal Infra*, 7:221–227.
- Maheen, U., Malik, K. I., and Ali, G. (2021). Comparative analysis of deep learning algorithms for classification of covid-19 x-ray images. *arXiv preprint arXiv:2110.09294*.
- Maruyama, T., Hayashi, N., Sato, Y., Hyuga, S., Wakayama, Y., Watanabe, H., Ogura, A., and Ogura, T. (2018). Comparison of medical image classification accuracy among three machine learning methods. *Journal of X-ray Science and Technology*, 26:885–893.
- Maskuri, N. L., Bakar, M. H. A., and Ismail, A. K. (2020). The image processing technique of defect detection in metal materials using active infrared thermography.
- MathWorks (2021). What is a convolutional neural network?

- Mikołajczyk, A. and Grochowski, M. (2018). Data augmentation for improving deep learning in image classification problem. pages 117–122. IEEE.
- Mostafa, A. M., Kumar, S. A., Meraj, T., Rauf, H. T., Alnuaim, A. A., and Alkhayyal, M. A. (2021). Guava disease detection using deep convolutional neural networks: A case study of guava plants. *Applied Sciences*, 12:239.
- Muhammad, Y., Tahir, M., Hayat, M., and Chong, K. T. (2020). Early and accurate detection and diagnosis of heart disease using intelligent computational model. *Scientific reports*, 10:1–17.
- Naz, S., Bibi, K., and Ahmad, R. (2022). Deepsignature: fine-tuned transfer learning based signature verification system. *Multimedia Tools and Applications*, pages 1–10.
- Novitasari, D. C. R., Wulandari, P., and Haq, D. Z. (2022). Cervical cancer diagnosis system using convolutional neural network residualnet. *International Journal of Computing*, 21:61–68.
- Pavone, V., Vescio, A., Turchetta, M., Giardina, S. M. C., Culmone, A., and Testa, G. (2021). Injection-based management of osteoarthritis of the knee: A systematic review of guidelines. *Frontiers in Pharmacology*, 12:661805.
- Peterson, L. S. (2020). *Mayo Clinic Guide to Arthritis: Managing Joint Pain for an Active Life*. Rosetta Books.
- Philips, T., Vanmierlo, B., and Goubau, J. F. (2022). Arthroplasty of the proximal interphalangeal joint with the tactys prosthesis: Clinical and radiographic results with a mean follow-up of 5 years. *HAND*, page 15589447211030962.
- Prostem, I. (2019). Osteoarthritis (oa).

- Rakamawati, J. V. N. (2021). Klasifikasi diabetic retinopathy berdasarkan foto fundus menggunakan convolutional neural network (cnn) jenis densenet.
- Raman, S., Maskeliūnas, R., and Damaševičius, R. (2021). Markerless dog pose recognition in the wild using resnet deep learning model. *Computers*, 11:2.
- Razzak, I., Shoukat, G., Naz, S., and Khan, T. M. (2020). Skin lesion analysis toward accurate detection of melanoma using multistage fully connected residual network. pages 1–8. IEEE.
- Rizani, F. (2021). Image quality improvement using image processing method image brightness contrast and image sharpening. *MULTICA SCIENCE AND TECHNOLOGY (MST) JOURNAL*, 1:6–12.
- Sahoo, S. (2018). Residual blocks — building blocks of resnet.
- Sarwinda, D., Paradisa, R. H., Bustamam, A., and Anggia, P. (2021). Deep learning in image classification using residual network (resnet) variants for detection of colorectal cancer. *Procedia Computer Science*, 179:423–431.
- Sathesh, A. and Adam, E. E. B. (2021). Hybrid parallel image processing algorithm for binary images with image thinning technique. *Journal of Artificial Intelligence*, 3:243–258.
- Sharma, S., Zhang, Y., Aliaga, J. M. R., Bouneffouf, D., Muthusamy, V., and Varshney, K. R. (2020). Data augmentation for discrimination prevention and bias disambiguation. pages 358–364.
- Shi, J., Dang, J., Cui, M., Zuo, R., Shimizu, K., Tsunoda, A., and Suzuki, Y. (2021). Improvement of damage segmentation based on pixel-level data balance using vgg-unet. *Applied Sciences*, 11:518.

- Shorten, C. and Khoshgoftaar, T. M. (2019). A survey on image data augmentation for deep learning. *Journal of big data*, 6:1–48.
- Showkat, S. and Qureshi, S. (2022). Efficacy of transfer learning-based resnet models in chest x-ray image classification for detecting covid-19 pneumonia. *Chemometrics and Intelligent Laboratory Systems*, 224:104534.
- Sihotang, J. (2019). Implementation of gray level transformation method for sharpening 2d images. *INFOKUM*, 8:16–19.
- Singh, Y. K. (2021). Detection of fire regions using rgb color variance. *Int. J. Res. Eng. Sci*, 9:10–14.
- Sukegawa, S., Yoshii, K., Hara, T., Matsuyama, T., Yamashita, K., Nakano, K., Takabatake, K., Kawai, H., Nagatsuka, H., and Furuki, Y. (2021). Multi-task deep learning model for classification of dental implant brand and treatment stage using dental panoramic radiograph images. *Biomolecules*, 11:815.
- Szyc, K. (2019). An impact of different images color spaces on the efficiency of convolutional neural networks. pages 506–514. Springer.
- Tabrizchi, H., Javidi, M. M., and Amirzadeh, V. (2021). Estimates of residential building energy consumption using a multi-verse optimizer-based support vector machine with k-fold cross-validation. *Evolving Systems*, 12:755–767.
- Tekade, R. and Rajeswari, K. (2018). Lung cancer detection and classification using deep learning. pages 1–5.
- Teoh, Y. X., Lai, K. W., Usman, J., Goh, S. L., Mohafez, H., Hasikin, K., Qian, P., Jiang, Y., Zhang, Y., and Dhanalakshmi, S. (2022). Discovering knee osteoarthritis

imaging features for diagnosis and prognosis: Review of manual imaging grading and machine learning approaches. *Journal of Healthcare Engineering*, 2022.

Thienkarochanakul, K., Javadi, A. A., Akrami, M., Charnley, J. R., and Benattayallah, A. (2020). Stress distribution of the tibiofemoral joint in a healthy versus osteoarthritis knee model using image-based three-dimensional finite element analysis. *Journal of Medical and Biological Engineering*, 40:409–418.

Thomas, K. N., Jain, N., Mohindra, N., Misra, D., Agarwal, V., and Gupta, L. (2022). Mri and sonography of the knee in acute reactive arthritis: An observational cohort study. *JCR: Journal of Clinical Rheumatology*, 28:e511–e516.

Tölli, J. P., Koski, J., Kaukinen, P., Haapea, M., Tervonen, O., Arokoski, J., and Saarakkala, S. (2017). Structure-symptom relationship with wide-area ultrasound scanning of knee osteoarthritis. *Scientific Reports*, 7:44470.

Venkatachalam, J., Natesan, M., Eswaran, M., Johnson, A. K. S., Bharath, V., and Singh, Z. (2018). Prevalence of osteoarthritis of knee joint among adult population in a rural area of kanchipuram district, tamil nadu. *Indian journal of public health*, 62:117.

Victoria, L. W. (2022). Rheumatoid arthritis: An autoimmune disorder. *Acta Rheuma*, 9:10.

Wang, Q., Ma, Y., Zhao, K., and Tian, Y. (2022). A comprehensive survey of loss functions in machine learning. *Annals of Data Science*, 9:187–212.

Wang, S.-H., Phillips, P., Sui, Y., Liu, B., Yang, M., and Cheng, H. (2018). Classification of alzheimer's disease based on eight-layer convolutional neural

- network with leaky rectified linear unit and max pooling. *Journal of medical systems*, 42:1–11.
- Wang, Y., Ge, Q., Lu, W., and Yan, X. (2020). Well-logging constrained seismic inversion based on closed-loop convolutional neural network. *IEEE Transactions on Geoscience and Remote Sensing*, 58:5564–5574.
- Wen, L., Li, X., and Gao, L. (2020). A transfer convolutional neural network for fault diagnosis based on resnet-50. *Neural Computing and Applications*, 32:6111–6124.
- WHO (2022). Noncommunicable diseases.
- Xiao, B., yu Li, X., guang Li, C., and fang Xu, Q. (2020). A novel pooling block for improving lightweight deep neural networks. *Pattern Recognition Letters*, 135:307–312.
- Xiong, N. N., Shen, Y., Yang, K., Lee, C., and Wu, C. (2018). Color sensors and their applications based on real-time color image segmentation for cyber physical systems. *EURASIP Journal on Image and Video Processing*, 2018:1–16.
- Yahaya, I., Wright, T., Babatunde, O. O., Helliwell, T., Dikomitis, L., and Mallen, C. D. (2021). Prevalence of osteoarthritis in lower middle-and low-income countries: a systematic review and meta-analysis. *Rheumatology international*, 41:1221–1231.
- Yani, M. (2019). Application of transfer learning using convolutional neural network method for early detection of terry’s nail. volume 1201, page 012052. IOP Publishing.
- Yingge, H., Ali, I., and Lee, K.-Y. (2020). Deep neural networks on chip-a survey. pages 589–592. IEEE.

- Yoo, J. J., Kim, D. H., and Kim, H. A. (2018). Risk factors for progression of radiographic knee osteoarthritis in elderly community residents in korea. *BMC Musculoskeletal Disorders*, 19:1–7.
- Yudhana, A., Umar, R., and Ayudewi, F. M. (2019). The monitoring of corn sprouts growth using the region growing methods. volume 1373, page 012054. IOP Publishing.
- Zeng, G. (2020). On the confusion matrix in credit scoring and its analytical properties. *Communications in Statistics - Theory and Methods*, 4:2080–2093.
- Zeng, K., Hua, Y., Xu, J., Zhang, T., Wang, Z., Jiang, Y., Han, J., Yang, M., Shen, J., and Cai, Z. (2021). Multicentre study using machine learning methods in clinical diagnosis of knee osteoarthritis. *Journal of Healthcare Engineering*, 2021.
- Zhang, C., Sargent, I., Pan, X., Gardiner, A., Hare, J., and Atkinson, P. (2018). Vprs-based regional decision fusion of cnn and mrf classifications for very fine resolution remotely sensed images. *IEEE Transactions on Geoscience and Remote Sensing*. pages 1–15.
- Zhu, J., Jang-Jaccard, J., and Watters, P. A. (2020). Multi-loss siamese neural network with batch normalization layer for malware detection. *IEEE Access*, 8:171542–171550.