TUGAS AKHIR – EB184803

DETEKSI PNEUMONIA PADA ANAK-ANAK DARI CITRA X-RAY BERBASIS *CONVOLUTIONAL NEURAL NETWORK*

Alif Hilmi Fauziyah
07311640000013

DOSEN PEMBIMBING:
Dr. Tri Arief Sardjono, S.T., M.T.
Muhammad Yazid, B.Eng., M.Eng.

PROGRAM SARJANA
DEPARTEMEN TEKNIK BIOMEDIK
FAKULTAS TEKNOLOGI ELEKTRO DAN INFORMATIKA CERDAS
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA
2020

TUGAS AKHIR – EB184803

DETEKSI PNEUMONIA PADA ANAK-ANAK DARI CITRA X-RAY BERBASIS *CONVOLUTIONAL NEURAL NETWORK*

Alif Hilmi Fauziyah
07311640000013

DOSEN PEMBIMBING:
Dr. Tri Arief Sardjono, S.T., M.T.
Muhammad Yazid, B.Eng., M.Eng.

PROGRAM SARJANA
DEPARTEMEN TEKNIK BIOMEDIK
FAKULTAS TEKNOLOGI ELEKTRO DAN INFORMATIKA CERDAS
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA
2020

FINAL PROJECT – EB184803

PNEUMONIA DETECTION IN CHILDREN FROM X-RAY IMAGES BASED ON CONVOLUTIONAL NEURAL NETWORK

Alif Hilmi Fauziyah
07311640000013

SUPERVISOR:
Dr. Tri Arief Sardjono, S.T., M.T.
Muhammad Yazid, B.Eng., M.Eng.

UNDERGRADUATE PROGRAM
BIOMEDICAL ENGINEERING DEPARTMENT
FACULTY OF INTELLIGENT ELECTRICAL AND INFORMATICS TECHNOLOGY
SEPULUH NOPEMBER INSTITUTE OF TECHNOLOGY
SURABAYA
2020

PERNYATAAN KEASLIAN TUGAS AKHIR

Dengan ini saya menyatakan bahwa sebagian maupun keseluruhan isi tugas akhir saya yang berjudul "**Deteksi Penumonia pada Anak-Anak dari Citra X-Ray Berbasis *Convolutional Neural Netwrok***" merupakan hasil karya intelektual mandiri, diselesaikan tanpa menggunakan bahan-bahan yang tidak diijinkan dan bukan merupakan karya pihak lain yang saya akui sebagai karya mandiri.

Semua referensi yang dikutip maupun dirujuk telah ditulis secara lengkap pada daftar pustaka. Apabila ternyata pernyataan ini tidak benar, saya bersedia menerima sanksi sesuai peraturan yang berlaku.

Surabaya, Juli 2020

Alif Hilmi Fauziyah

NRP. 07311640000013

Tugas Akhir disusun untuk memenuhi salah satu syarat memperoleh gelar
Sarjana Teknik (S.T.)
di
Institut Teknologi Sepuluh Nopember

Oleh
Alif Hilmi Fauziyah
NRP. 07311640000013

Tanggal Ujian : **14 Juli 2020**
Periode Wisuda : **September 2020**

Disetujui Oleh :

Dosen Penguji :

1. Dr. Rachmad Setiawan, S.T., M.T.
NIP. 196905291995121001

2. Atar Fuady Babgei, S.T., M.Sc.
NIP. 198911112018121001

3. Nada Fitrieyatul Hikmah, S.T., M.T.
NIP. 199001072018032001

Dosen Pembimbing :

1. Dr. Tri Arief Sardjono, S.T., M.T.
NIP. 197002121995121001

2. Muhammad Yazid, B.Eng., M.Eng.
NIP. 198004202015041001

Kepala Departemen Teknik Biomedik,

Dr. Achmad Arifin, S.T., M.Eng.
NIP. 197103141997021001

DETEKSI PNEUMONIA PADA ANAK-ANAK DARI CITRA X-RAY BERBASIS *CONVOLUTIONAL NEURAL NETWORK*

Nama mahasiswa	:	Alif Hilmi Fauziyah
NRP	:	07311640000013
Pembimbing	:	Dr. Tri Arief Sardjono, S.T., M.T.
	:	Muhammad Yazid, B.Eng., M.Eng.

Abstrak

Deteksi suatu penyakit melalui suatu citra X-Ray masih bergantung pada diagnosis tenaga medis. Deteksi secara manual berdasarkan pengamatan visual dari tenaga medis masih banyak dilakukan untuk membaca hasil foto X-Ray. Cara manual ini memiliki kelemahan yaitu adanya perbedaan interpretasi antar pengamat sehingga dapat terjadi kesalahan dalam diagnosis suatu penyakit. Serta membutuhkan waktu yang cukup lama saat mendiagnosis secara manual. Pada penelitian ini dibuat suatu sistem untuk mendeteksi pneumonia pada anak-anak dari hasil foto X-ray dengan menggunakan metode *Convolutional Neural Network* (CNN). Tahap pemrosesan awal juga dilakukan untuk meningkatkan kualitas citra. Metode median filter dan *contrast limited adaptive histogram equalization* (CLAHE) diterapkan untuk *image enhancement* sebelum citra memasuki proses CNN. Pada CNN terdapat jaringan dengan 2 lapisan utama, yaitu *feature learning* yang terdiri dari *convolution layer* dan *pooling layer*, serta lapisan klasifikasi yang terdiri dari *fully connected layer*. Pada *fully connected layer* akan diberlakukan fungsi aktivasi sigmoid untuk mengklasifikasikan citra ke dalam 2 kelas, yaitu normal dan pneumonia. Hasil akhir yang didapatkan dalam tugas akhir ini adalah performa model yang cukup baik dengan akurasi yang didapatkan yaitu sebesar 96,06 % untuk akurasi validasi dan akurasi pelatihan sebesar 96,18 %. *Confusion matrix* juga digunakan untuk mengukur kinerja sistem. Berdasarkan *confusion matrix* perhitungan presisi, *recall*, dan spesifisitas yang didapatkan adalah 97,83 %, 97,12 %, dan 92,27 %. Kurva ROC yang dihasilkan juga menunjukkan bahwa nilai *Area Under Curve* (AUC) sebesar 0,95 dengan grafik yang menjauhi garis *baseline*.

Kata kunci : Pneumonia, *Convolutional Neural Network*, CLAHE, Median filter.

PNEUMONIA DETECTION IN CHILDREN FROM X-RAY IMAGES BASED ON CONVOLUTIONAL NEURAL NETWORK

Name	:	Alif Hilmi Fauziyah
Student identity number:		07311640000013
Supervisor	:	Dr. Tri Arief Sardjono, S.T., M.T.
	:	Muhammad Yazid, B.Eng., M.Eng.

Abstract

Detection of the disease through X-Ray images is still according to the diagnosis of medical personnel. Manual detection of visual reading from medical personnel is still mostly done to read X-Ray images. This manual method has the disadvantage that there are differences in interpretation between observers so that errors can occur in the diagnosis of a disease. And requires quite a long time when diagnosing manually. In this study a system was made to test pneumonia in children from X-ray images using the Convolutional Neural Network (CNN) method. The initial maintenance phase is also carried out to improve image quality. The median filter and contrast limited adaptive histogram equalization (CLAHE) method is applied for image enhancement before the process of CNN. CNN consists of 2 main layers, namely the feature learning feature consists of a convolution layer and a pooling layer, as well as a classification layer consist of a fully connected layer. In the fully connected layer the sigmoid activation function will be applied to classify the image into 2 classes, there is normal and pneumonia. The final results obtained in this thesis is a good performance model with an accuracy is 96.06% for validation accuracy and training accuracy is 96.18%. The confusion matrix is also used to measure system performance. Based on the confusion matrix the calculation of precision, recall, and specificity obtained were 97.83%, 97.12%, and 92.27%. The resulting ROC curve also shows an Area Under Curve (AUC) value of 0.95 with a graph that moves away from the baseline.

Keywords : Pneumonia, Convolutional Neural Network, CLAHE, Median filter.

KATA PENGANTAR

Segala puji bagi Allah SWT atas segala rahmat-Nya, karena-Nya Tugas Akhir ini dapat berjalan lancar dan selesai tepat pada waktunya. Maksud dan tujuan dari penulisan Tugas Akhir ini adalah untuk memenuhi persyaratan kelulusan program Studi Strata I pada Jurusan Teknik Biomedik di Institut Teknologi Sepuluh Nopember. Menyadari penyusunan laporan ini tidak lepas dari bantuan berbagai pihak, maka pada kesempatan ini penulis menyampaikan ucapan terima kasih kepada:

1. Kedua orang tua yang telah mendukung, memberi nasehat, saran, serta memberikan fasilitas untuk membantu penyelesaian Tugas Akhir ini.
2. Bapak Dr. Tri Arief Sardjono, S.T., M.T. dan Muhammad Yazid, B.Eng., M.Eng. selaku dosen pembimbing yang senantiasa membimbing, mengarahkan, serta menginspirasi penulis selama pengerjaan penelitian ini.
3. Dr. Achmad Arifin, S.T., M.Eng. selaku kepala departemen yang telah memberi inspirasi, motivasi, serta bimbingan selama pengerjaan tugas akhir terlebih selama perkuliahan di ITS.
4. Ibu Nada Fitrieyatul Hikmah, S.T., M.T. selaku dosen wali.
5. Seluruh staf pengajar Teknik Biomedik ITS yang telah membimbing dan memberikan materi perkuliahan kepada penulis.
6. Rekan-rekan Teknik Biomedik angkatan 2016 dan juga pihak-pihak lain yang turut membantu dalam penyelesaian Tugas Akhir ini.

Penulis merasa bahwa dalam menyusun laporan ini masih menemui beberapa kesulitan dan hambatan, disamping itu juga menyadari bahwa penulisan laporan ini masih jauh dari sempurna dan masih banyak kekurangan-kekurangan lainnya, maka dari itu penulis mengharapkan saran dan kritik yang membangun dari semua pihak. Semoga Tugas Akhir ini akan memberikan manfaat bagi penulis maupun bagi pihak-pihak yang sedang membutuhkan.

Surabaya, Juli 2020

Alif Hilmi Fauziyah

NRP. 07311640000013

DAFTAR ISI

PERNYATAAN KEASLIAN TUGAS AKHIR i
LEMBAR PENGESAHAN ii
Abstrak iii
Abstract iv
KATA PENGANTAR v
DAFTAR GAMBAR viii
DAFTAR TABEL x
BAB I PENDAHULUAN 1
1.1 Latar Belakang 1
1.2 Rumusan Masalah 2
1.3 Batasan Masalah 3
1.4 Tujuan dan Manfaat 3
1.5 Kontribusi Penelitian 3
BAB II TINJAUAN PUSTAKA 4
2.1 Anatomi dan Fisiologi Paru 4
2.2 Pneumonia 5
2.3 Radiografi Dada 6
2.4 Pengolahan Citra Digital 6
2.5 Pra-Pemrosesan Citra 7
2.5.1 *Resize* 7
2.5.2 *Image Filtering* 8
2.5.3 Perbaikan Kualitas Citra (*Image Enhancement*) 8
2.6 *Machine Learning* 9
2.7 *Deep Learning* 10
2.8 *Convolutional Neural Network* 11
2.9 Python 13
2.10 Keras 14
2.11 *Google Colaboratory* 14
2.12 *Graphical User Interface* (GUI) 14
2.13 Rasional 15
BAB III PERANCANGAN SISTEM 18
3.1 Desain Sistem Secara Umum 18
3.2 *Dataset* 19
3.2.1 Keterangan *Dataset* 19
3.2.2 Pembuatan Data Latih dan Data Validasi 21
3.3 *Image Pre-Processing* 21
3.3.1 *Filtering* 22
3.3.2 *Histogram Equalization* 23
3.4 *Convolutional Neural Network* (CNN) 23

3.4.1 Perancangan Arsitektur CNN 24
3.4.2 Perancangan Proses Pelatihan pada CNN 26
3.5 Metode Pengujian 28
BAB IV PENGUJIAN SISTEM 30
4.1 *Image Pre-Processing* 30
4.1.1 Filter Median 30
4.1.2 *Contrast Limited Adaptive Histogram Equalization* 32
4.2 CNN dalam Pendeteksian Citra 32
4.3 Prediksi Citra Menggunakan GUI 35
4.4 Pengujian dengan *Balancing Data* 36
4.4.1 *Balancing Data* dengan Metode *Undersampling* 36
4.4.2 *Balancing Data* dengan Metode Augmentasi Citra 39
4.5 Perbandingan Performa CNN Berdasarkan Kondisi *Dataset* 43
BAB V HASIL DAN PEMBAHASAN 44
5.1 *Image Pre-Processing* 44
5.1.1 *Median Filter* 45
5.1.2 *Contrast Limited Adaptive Histogram Equalization* 46
5.2 CNN untuk Deteksi Citra 47
5.3 Penentuan Parameter Model 52
5.3.1 Pengaruh Jumlah Lapisan Konvolusi 54
5.3.2 Pengaruh Jumlah *Epoch* 56
5.3.3 Pengaruh *Learning Rate* 56
5.4 Model Hasil *Training* 57
BAB VI KESIMPULAN DAN SARAN 60
6.1 Kesimpulan 60
6.2 Saran 60
DAFTAR PUSTAKA 62
LAMPIRAN 64
BIODATA PENULIS 65

DAFTAR GAMBAR

Gambar 2. 1 Anatomi Paru-Paru [7] 4
Gambar 2. 2 Struktur pemodelan jaringan pada *Machine Learning* [11] 9
Gambar 2. 3 Struktur pemodelan jaringan pada *Deep Learning* [11] 10
Gambar 2. 4 Arsitektur CNN [12] 12
Gambar 2. 5 Operasi Konvolusi [13] 12
Gambar 2. 6 Operasi *Max Pooling* [13] 13
Gambar 2. 7 Diagram *Fishbone* 17
Gambar 3. 1 Diagram Blok Sistem Secara Umum 19
Gambar 3. 2 Sampel citra paru-paru normal dari dataset 20
Gambar 3. 3 Sampel citra paru-paru pneumonia 20
Gambar 3. 4 Sampel citra paru-paru yang sulit dibedakan (a) Normal; (b) Pneumonia 21
Gambar 3. 5 Diagram Alir Tahap *Preprocessing* 22
Gambar 3. 6 Diagram alir sistem klasifikasi 24
Gambar 4. 1 Citra masukan 30
Gambar 4. 2 (a) Citra sebelum di*resize*; (b) Citra setelah di*resize* 31
Gambar 4. 3 (a) Citra sebelum difilter; (b) Hasil dari proses *median filter* 31
Gambar 4. 4 (a) Citra sebelum CLAHE; (b) Hasil dari proses CLAHE 32
Gambar 4. 5 Hasil *Confusion Matrix* 33
Gambar 4. 6 Hasil Kurva ROC 34
Gambar 4. 7 Tampilan antarmuka GUI 35
Gambar 4. 8 Tampilan antarmuka setelah GUI dijalankan 36
Gambar 4. 9 Distribusi dataset (a) Sebelum dilakukan *undersampling*; (b) Setelah dilakukan *undersampling* 37
Gambar 4. 10 Hasil *confusion matrix* berdasarkan dataset yang telah di *undersampling* 38
Gambar 4. 11 (a) Grafik *train and validation loss* setelah *undersampling;* (b) Grafik *train and validation accuracy* setelah *undersampling* 38
Gambar 4. 12 Kurva ROC berdasarkan *dataset* yang telah di *undersampling* .. 39
Gambar 4. 13 Distribusi *dataset* (a) Sebelum dilakukan augmentasi; (b) Setelah dilakukan augmenasi 40
Gambar 4. 14 (a) Grafik *train and validation loss* setelah augmentasi; (b) Grafik *train and validation accuracy* setelah augmentasi 41
Gambar 4. 15 Hasil *confusion matrix* berdasarkan *dataset* yang telah di augmentasi 42
Gambar 4. 16 Kurva ROC berdasarkan *dataset* yang telah di augmentasi 42
Gambar 5. 1 Hasil median filter dengan ukuran *kernel* (a) Citra setelah di*resize*; (b) 3x3; (c) 5x5; (d) 7x7 45
Gambar 5. 2 Hasil CLAHE dengan ukuran *clip limit* (a) Citra setelah di*resize* dan difilter; (b) 0,005; (c) 0,01; (d) 0,02; (e) 0,5 46

Gambar 5. 3 Hasil f*eature maps* pada bagian *feature learning*..... 54
Gambar 5. 4 (a) Grafik *training and validation accuracy*; (b) Grafik *training and validation loss* 58

DAFTAR TABEL

Tabel 3. 1 Distribusi *Dataset* 20
Tabel 3. 2 Distribusi *dataset* yang baru 21
Tabel 3. 3 Arsitektur CNN 25
Tabel 3. 4 Pengaturan proses augmentasi data 27
Tabel 3. 5 *Hyperparameter* untuk Pembelajaran CNN 27
Tabel 4. 1 Hasil Klasifikasi Menggunakan CNN 34
Tabel 4. 2 Performa CNN untuk pengujian *balancing data* dengan metode *undersampling* 37
Tabel 4. 3 Pengaturan metode augmentasi untuk citra normal pada data latih ... 40
Tabel 4. 4 Performa CNN untuk pengujian balancing data dengan metode augmentasi 40
Tabel 4. 5 Hasil perbandingan performa CNN berdasarkan kondisi *dataset* 43
Tabel 5. 1 Hasil perbandingan akurasi dari CNN dengan menggunakan *preprocessing* dengan tidak menggunakan *preprocessing* 44
Tabel 5. 2 Keluaran model CNN 55
Tabel 5. 3 Perbandingan akurasi berdasarkan jumlah lapisan konvolusi 55
Tabel 5. 4 Perbandingan akurasi berdasarkan jumlah *epoch* 56
Tabel 5. 5 Perbandingan akurasi berdasarkan perubahan *learning rate* 57
Tabel 5. 6 Perbandingan performa model dengan penelitian sebelumnya 59

BAB I PENDAHULUAN

1.1 Latar Belakang

Menurut *World Health Organization* (WHO) definisi dari pneumonia adalah bentuk infeksi pernapasan akut yang menyerang paru-paru [1]. Paru-paru terdiri dari kantung-kantung kecil yang disebut dengan alveoli, yang akan terisi udara ketika bernapas. Ketika seseorang menderita pneumonia, alveoli dalam paru-parunya dipenuhi dengan nanah dan cairan, yang membuat pernapasan terasa menyakitkan dan membatasi jumlah oksigen yang masuk ke dalam tubuh. Penyakit ini terjadi karena adanya infeksi bakteri, virus, atau parasit. Kasus pneumonia yang paling parah disebabkan oleh bakteri *Streptococcus pneumoniae* dan *Haemophilus influenza.* Pneumonia dapat menyebar dalam beberapa cara. Virus dan bakteri yang berada di udara kemudian terhirup dapat menginfeksi paru-paru. Penderita pneumonia juga dapat menyebarkan virus dan bakteri tersebut dalam bentuk droplet ke udara saat batuk atau bersin. Selain itu pneumonia juga dapat menyebar melalui darah, terutama saat kelahiran, darah dari ibu yang menderita pneumonia akan menular ke bayinya. Terjadinya pneumonia biasanya ditandai dengan gejala batuk atau kesulitan dalam bernapas.

Pneumonia dapat menyerang segala usia tetapi paling sering terjadi pada anak-anak dan orang tua. Menurut *World Health Organization* (WHO), pneumonia adalah penyebab kematian utama pada anak-anak di seluruh dunia, yang telah menewaskan sekitar 1,9 juta anak di bawah lima tahun setiap tahunnya [2]. Di Indonesia pneumonia masih menjadi penyebab kematian tertinggi pada anak di bawah usia lima tahun (balita) maupun bayi baru lahir. Berdasarkan data dari Riset Kesehatan Dasar (Riskesdas) tahun 2018 menunjukkan prevalensi pneumonia naik dari 1,6 % pada 2013 menjadi 2 % dari populasi balita yang ada di Indonesia pada 2018. Berdasarkan data tersebut dapat disimpulkan bahwa pneumonia banyak menyerang anak-anak dan bahkan dapat menyebabkan kematian. Jika dilakukan diagnosis dini maka terapi antibiotik dapat dilakukan untuk mengobati pneumonia secara efektif. Oleh karena itu sangat penting untuk membuat diagnosa yang akurat dan secepat mungkin. Serta dengan diagnosis dini dan pengobatan pneumonia sedini mungkin dapat mengurangi angka kematian pada anak-anak penderita pneumonia.

Diagnosa pneumonia pada anak cukup sulit, karena sensitivitas yang rendah dari tes mikrobiologis dan spesifisitas yang rendah dari tanda-tanda dan gejala klinis. Terdapat beberapa cara untuk mendiagnosa pneumonia salah satunya adalah kultur dahak (*sputum*), tes ini dilakukan pada dahak yang keluar dari paru-paru dan masuk ke mulut. Akan tetapi cara ini cukup sulit dilakukan untuk mendiagnosa pneumonia pada anak-anak karena kesulitan untuk mendapatkan sampel dahak pada anak-anak dibandingkan pada orang dewasa. Dengan demikian, radiografi dada menggunakan X-Ray menjadi modalitas penting untuk mendeteksi pneumonia

pada anak-anak. Foto polos dada dengan tampilan *posteoanterior* dan *lateral* adalah pemeriksaan pencitraan pilihan untuk mengevaluasi pneumonia. Foto toraks pada pasien pneumonia biasanya ditandai dengan adanya konsolidasi pada paru. Konsolidasi paru adalah suatu temuan pada citra foto toraks berupa bercak berawan pada paru. Konsolidasi terjadi karena udara yang biasanya mengisi alveoli digantikan oleh unsur yang lain, bisa berupa cairan seperti nanah, darah, atau air, bisa juga berupa benda padat.

Diagnosis pneumonia memerlukan kombinasi pengetahuan klinis, uji mikrobiologis yang sesuai, dan pengetahuan mengenai radiografi. Dokter yang mengevaluasi pasien dengan diagnosis adanya infeksi pada paru menghadapi tantangan dimana sebagian besar tanda dan gejalanya serupa, dan pendeteksian pneumonia biasanya tidak terlalu spesifik [3]. Beberapa penelitian juga menunjukkan bahwa sering terjadi kesalahan pada interpretasi radiografi dada yang disebabkan oleh adanya perbedaan penafsiran antar pengamat [4]. Munculnya pneumonia pada citra X-Ray sering tidak jelas, dapat membingungkan dan dapat menyerupai dengan penyakit lainnya. Hal inilah yang dapat menyebabkan keputusan subyektif dan perbedaan penafsiran di antara ahli radiologi dalam mendiagnosis pneumonia [5]. Oleh karena itu, diperlukan sistem untuk membantu ahli radiologi dalam mendiagnosa pneumonia dari gambar radiografi dada. Metode *deep learning* yang saat ini memiliki hasil paling signifikan dalam pengenalan citra adalah *Convolutional Neural Network* (CNN) [6]. Perkembangan terbaru dalam bidang *deep learning,* khususnya *Convolutional Neural Network* (CNN) menunjukkan keberhasilan besar dalam klasifikasi gambar. Namun CNN, seperti metode *deep learning* lainnya, memiliki kelemahan yaitu proses pelatihan model yang lama.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah diuraikan diatas maka peneliti merumuskan beberapa permasalahan. Rumusan masalah yang pertama adalah bagaimana perancangan sebuah sistem untuk mendeteksi pneumonia pada anak-anak dari citra X-Ray. Karena deteksi pneumonia secara manual masih memiliki kekurangan seperti tanda dan gejala pneumonia dapat menyerupai dengan kelainan yang lain, kemudian adanya perbedaan penafsiran antar pengamat.

Rumusan masalah yang kedua adalah mengenai pengaruh *image enhancement* (CLAHE) terhadap hasil klasifikasi yang berbasis *Convolutional Neural Network*. Karena pada penelitian sebelumnya mengenai deteksi pneumonia pada anak-anak menggunakan metode *Convolutional Neural Network* hasil yang didapatkan adalah 94%, tanpa adanya tahap *pre-processing* pada citra yang diproses.

1.3 Batasan Masalah

Pada penelitian ini dibuat batasan-batasan masalah yang akan diteliti. Batasan masalah dalam penelitian ini adalah pengujian dilakukan berdasarkan *dataset* yang di dapat dari suatu website *open source* berupa citra x-ray dada dari pasien anak berusia satu hingga lima tahun dari *Guangzhou Women and Children's Medical Center*, dengan foto toraks dalam posisi *Posteo Anterior* (PA) yakni sinar dipancarkan melalui punggung (posterior) pasien. *Dataset* yang digunakan untuk memproses citra didapatkan dari dataset yang sudah tersedia tanpa melakukan pengambilan data secara klinis.

1.4 Tujuan dan Manfaat

Tujuan dari penelitian ini adalah merancang suatu sistem untuk mendeteksi pneumonia pada anak-anak dari hasil citra X-Ray serta mengetahui pengaruh dari penambahan *image enhancement* dalam mengklasifikasikan suatu citra menggunakan CNN.

Manfaat yang ingin dicapai dalam penelitian ini meliputi aspek teoritis dan aspek praktis. Manfaat dalam aspek teoritis yang didapatkan adalah dapat menambah wawasan dan khasanah ilmiah yang berkaitan dengan citra pneumonia menggunakan X-Ray serta dapat menambah wawasan mengenai *image enhancement* pada citra dan penggunaan CNN untuk mengklasifikasikan citra. Manfaat dalam aspek praktis yang diharapkan adalah efektivitas dalam diagnosa pneumonia pada hasil radiografi dada pada anak-anak sehingga memudahkan dokter untuk memberikan penanganan yang tepat pada pasien terutama pada anak-anak.

1.5 Kontribusi Penelitian

Kontribusi dari penelitian ini dapat dilihat dari dua aspek yaitu ilmiah dan praktis. Pada aspek ilmiah, kontribusinya adalah menjadi suatu pengembangan untuk pendidikan dan penelitian dalam bidang biomedik. Penelitian ini merupakan Sistem Pendeteksian Pneumonia pada Anak-Anak dari Hasil Radiografi Dada Menggunakan Modalitas X-Ray.

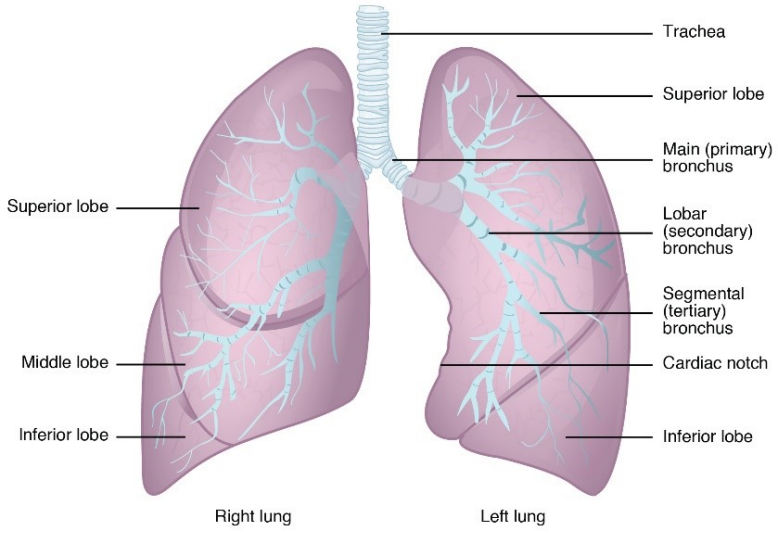
Lebih dari itu, kontribusi penelitian ini pada aspek praktis adalah penelitian yang dilakukan pada tugas akhir ini dapat menjadi solusi untuk dokter dalam memantau atau mendiagnosis pneumonia pada anak-anak dari citra X-Ray, dokter juga dimudahkan dalam memberikan penanganan yang tepat bagi pasien karena hasil yang didapat nantinya dapat menunjukkan kondisi paru-paru dalam keadaan normal atau mengalami penyumbatan pada alveolar atau yang dikenal dengan pneumonia.

BAB II TINJAUAN PUSTAKA

2.1 Anatomi dan Fisiologi Paru

Paru-paru adalah sepasang organ yang terletak di dalam rongga dada pada sisi lain jantung dan pembuluh darah besar. Paru-paru berbentuk seperti piramida dengan puncak di bagian atas dan alas di bawah. Paru-paru terhubung dengan trakea melalui bronkus kanan dan kiri, pada permukaan inferior paru-paru dibatasi oleh diafragma. Secara anatomi paru-paru tertutup oleh pleura yang melekat pada mediastinum. Paru-paru dibagi menjadi beberapa lobus dengan jumlah lobus antara paru-paru kiri dan kanan yang berbeda. Paru-paru sebelah kanan memiliki tiga lobus. Lobus bawah dipisahkan oleh fisura oblik dan lobus sisanya dipisahkan oleh fisura horizontal yang menjadi lobus atas dan lobus tengah. Sedangkan paru-paru sebelah kiri dibagi menjadi dua lobus yaitu lobus atas dan lobus bawah.

Fungsi utama paru-paru adalah sebagai organ pernapasan yaitu sebagai tempat terjadinya pertukaran udara yang bertujuan menghirup udara dari atmosfer ke dalam paru-paru dan mengeluarkan udara dari paru-paru ke atmosfer. Secara anatomi saluran pernapasan terdiri dari rongga hidung, rongga mulut, faring, laring, trakea, dan paru. Pada laring saluran pernapasan terbagi menjadi dua saluran yaitu saluran pernapasan atas dan bawah. Pada pernapasan melalui paru-paru atau pernapasan eksternal, oksigen dihirup melalui hidung atau mulut. Pada saat respirasi atau bernapas, oksigen yang dihirup dari hidung atau mulut akan melalui trakea dan pipa bronkiolus dan masuk ke alveolus di paru-paru. Udara dapat bergerak masuk dan keluar paru-paru karena ada selisih tekanan antara atmosfer dan alveolus akibat pergerakan mekanik dari otot-otot.



Gambar 2. 1 Anatomi Paru-Paru [7]

2.2 Pneumonia

Pneumonia merupakan salah satu infeksi pada satu atau kedua paru-paru yang terjadi di kantung udara atau alveoli paru-paru, yang menimbulkan konsolidasi jaringan paru dan gangguan pertukaran gas setempat. Pneumonia biasanya disebabkan oleh virus atau bakteri yang ada di lingkungan atau adanya penularan dari orang lain yang terinfeksi. Secara umum bakteri mikroorganisme yang menjadi penyebab pneumonia adalah *Streptococcus pneumonia*, *Hemophilus influenza*, *Staphylococcus aureus*, serta kuman atipik klamidia dan mikoplasma. Umumnya mikroorganisme penyebab pneumonia tersebut terhisap ke paru bagian perifer melalui saluran pernapasan. Mikroorganisme yang terhirup dan masuk ke dalam paru-paru akan menyebabkan paru-paru meradang. Peradangan tersebut terjadi akibat adanya proliferasi atau pembiakan dari mikroba-mikroba yang masuk ke dalam paru-paru.

Pada pneumonia yang disebabkan oleh mikroorganisme baik virus maupun bakteri, yang menjadi pertanyaan penting adalah penyebab dari pneumonia tersebut apakah disebabkan oleh virus atau bakteri. Pada pneumonia yang disebabkan oleh bakteri umumnya infeksi terjadi pada trakea yang kemudian mencapai parenkim paru-paru. Sedangkan pada pneumonia yang disebabkan oleh virus, awal terjadinya infeksi adalah di sepanjang saluran pernapasan yang disertai dengan lesi pada epitel salurah pernapasan. Akibat infeksi yang disebabkan oleh bakteri atau virus, paru akan mengalami obstruksi atau penyumbatan karena peradangan. Infeksi yang berat lebih rentan dialami oleh anak-anak terutama bayi, karena anatomi saluran pernapasan yang lebih kecil. Penyumbatan saluran napas akan menyebabkan hipoksemia yaitu kondisi dimana kadar oksigen dalam darah rendah [8].

Pneumonia dapat ditularkan secara langsung yaitu menyentuh benda yang sebelumnya disentuh oleh penderita pneumonia atau menyentuh penderita misalnya melalui jabat tangan. Penularan pneumonia juga bisa terjadi ketika menghirup droplet di udara dari batuk atau bersin yang berasal dari individu yang terinfeksi pneumonia. Ketika paru-paru terinfeksi maka akan terjadi peradangan pada paru-paru dan alveoli terisi oleh lendir atau cairan lain. Ketika alveoli paru-paru terisi oleh lendir dan cairan lain maka kemampuan untuk menyerap oksigen akan berkurang. Kekurangan oksigen dapat menyebabkan sel-sel dalam tubuh rusak. Oleh karena itu penderita pneumonia bisa meninggal.

Diagnosis pneumonia dilakukan dengan maksud pengarahan dalam pemberian *treatment* atau perawatan yaitu dengan mencakup bentuk dan luas penyakit, tingkat keparahan, dan perkiraan jenis mikroorganisme penyebeb infeksi. Secara klinis, diagnosis pneumonia didasarkan atas tanda-tanda kelainan fisis dan adanya gambaran konsolidasi pada foto X-ray dada. Konsolidasi merupakan suatu gambaran pada radiografi dada yang berupa bercak berawan pada paru. Namun diagnosis lengkap harus mencakup diagnosis etiologi dan anatomi. Diagnosis pneumonia dapat dilakukan dalam beberapa metode yaitu :

a. Pemeriksaan laboratorium
   Pemeriksaan ini dilakukan dari sample darah atau sputum pasien.
b. Radiografi dada
   Pemeriksaan dengan x-ray akan mengidentifikasi adanya penyebaran dan juga menunjukkan multipel abses/infiltrat, empiema (*staphilococcus*), penyebaran atau lokasi infiltrasi (bakterial).
c. Analisis gas darah
   Pemeriksaan ini dilakukan dengan menilai tingkat hipoksemia pada darah.

2.3 Radiografi Dada

Radiografi dada atau yang sering disebut dengan C*hest X-Ray* (CXR) merupakan suatu proyeksi radiografi dari toraks atau dada untuk mendiagnosis kondisi-kondisi yang mempengaruhi toraks, isi, dan struktur-struktur di dalamnya. Radiografi dada menggunakan radiasi terionisasi dalam bentuk X-Ray. Dosis radiasi yang digunakan pada orang dewasa untuk membentuk radiografi adalah sekitar 0.06 mSv, sedangkan untuk pasien pediatrik (anak-anak) dosis yang diberikan harus dimodifikasi berdasarkan umur dan/atau berat badan pasien. Tes radiografi dada merupakan pemeriksan yang dilakukan dengan relatif cepat, mudah dilakukan, dan lebih murah dibandingkan dengan pemeriksaan menggunakan MRI atau CT Scan.

Radiografi dada digunakan untuk mendiagnosis banyak kondisi yang melibatkan dinding toraks, tulang toraks, dan struktur yang berada di dalam toraks termasuk paru-paru, jantung, dan saluran-saluran yang besar. Secara umum kegunaan radiografi dada adalah sebagai berikut :

a. Untuk melihat abnormalitas *congenital* (jantung, vaskuler)
b. Untuk melihat adanya trauma (pneumothorax, hematothorax)
c. Untuk melihat adanya infeksi (TBC)
d. Untuk memeriksa keadaan jantung
e. Untuk memeriksa keadaan paru-paru

Citra toraks dikatakan normal apabila pada citra tersebut di bagian paru-paru, jantung, atau saluran/organ lainnya tidak memiliki bercak, tekstur citra halus, bentuk dan ukuran tidak berubah. Sedangkan citra toraks dikatakan abnormal apabila pada citra tersebut memiliki bercak, tekstur citra yang tidak halus, bentuk dan ukuran dari organ jantung atau paru-paru berubah. Bentuk dan ukuran dari citra organ paru-paru atau jantung yang mengalami perubahan dapat dilihat dari intensitas piksel pada objek.

2.4 Pengolahan Citra Digital

Image processing atau pengolahan citra digital berkaitan dengan pengolahan citra yang menggunakan komputer. Persepsi virtual banyak dilibatkan pada analisis citra yang merupakan bagian dari proses pengolahan citra. Pengolahan

citra dimulai dengan satu citra dan akan menghasilkan versi modifikasi dari citra itu sendiri. Pengolahan citra secara umum merujuk pada pemrosesan gambar dan sebuah dimensi dengan menggunakan komputer. Tujuan utama pengolahan citra digital adalah supaya citra mudah diinterpretasikan baik oleh manusia maupun mesin. Pengolahan citra merupakan sebuah teknik atau metode yang bisa digunakan untuk mengolah gambar atau citra dengan cara memanipulasi citra yang ada, menjadi sebuah data gambar yang diinginkan agar menghasilkan informasi tertentu.

Pengolahan citra lebih fokus pada transformasi citra dan dalam hal peningkatan efek visual dari citra itu sendiri. Hal itu berarti bahwa dengan teknik pengolahan citra yang mengubah suatu citra menjadi citra lainnya dengan input berupa citra dan output juga berupa citra, akan tetapi citra output akan lebih baik hasilnya dibandingkan dengan citra input. Terdapat beberapa contoh operasi pada pemrosesan citra yaitu *cropping* citra, menghitung histogram dari sebuah citra, mengatur tingkat kecerahan, mengubah ukuran citra. Adapun 2 tujuan utama pada pengolahan citra, yaitu :

1. Memperbaiki kualitas citra yaitu citra output yang dihasilkan bias menampilkan informasi dengan jelas sehingga dapat diinterpretasikan oleh manusia.
2. Mengekstraksi informasi ciri-ciri yang unik pada sebuah citra, dimana hasilnya berupa citra yang bias didapat oleh manusia secara numerik ataupun informasi pada citra yang berhubungan dengan nilai-nilai data.

2.5 Pra-Pemrosesan Citra

Pra-pemrosesan atau *preprocessing* adalah proses awal dilakukannya perbaikan suatu citra untuk menghilangkan *noise*. *Preprocessing* merupakan proses untuk memperbaiki kualitas citra dengan cara menghilangkan bagian-bagian yang tidak diperlukan pada gambar input untuk proses selanjutnya. Beberapa proses yang dapat dilakukan pada tahap *preprocessing* antara lain :

2.5.1 *Resize*

Resize adalah mengubah besarnya ukuran citra digital dalam piksel. Mengubah ukuran gambar mempunyai tujuan untuk mengubah dimensi gambar tersebut seperti ukuran lebar, tinggi, dan aspek ratio. Pengertion aspek ratio suatu gambar adalah angka yang menunjukkan perbandingan panjang dan lebar suatu bidang gambar yang dinyatakan sebagai panjang atau lebar. Ketika ukuran gambar diubah, maka informasi piksel dalam gambar tersebut juga diubah. Misalnya, ukuran gambar diperkecil aka informasi piksel yang tidak dibutuhkan akan dibuang. Ada kalanya ukurannya berubah menjadi lebih kecil dari file aslinya dan ada kalanya sebaliknya. Ukurannya berubah menjadi kecil atau besar dilakukan secara proporsional baik pada panjang maupun lebar citra. Dalam mengklasifikasikan citra menggunakan *Convolutional Neural Network* proses mengubah ukuran gambar ini

diperlukan karena tahap ini akan sangat membantu dalam proses *training* data agar tidak memakan waktu yang lama, serta proses *resize* dilakukan karena masukan pada CNN hanya menerima masukan dengan ukuran tertentu. Pada salah satu penelitian menunjukkan bahwa mengubah ukuran citra menjadi lebih kecil dari ukuran aslinya dapat menghabiskan proses *training* yang lebih cepat [9].

2.5.2 *Image Filtering*

Image filtering merupakan suatu cara untuk mengekstraksi bagian tertentu pada suatu citra dengan menghilangkan bagian atau elemen yang tidak diinginkan. Pemfilteran pada citra berguna dalam banyak aplikasi seperti menghaluskan (*smoothing*), menajamkan citra (*sharpening*), menghilangkan *noise* atau gangguan, dan deteksi tepi. Filter didefinisikan oleh sebuah kernel, yang merupakan array kecil yang diterapkan untuk setiap piksel dalam suatu citra. Proses yang digunakan untuk menerapkan suatu filter pada citra dikenal sebagai konvolusi yang dapat diterapkan dalam domain spasial atau frekuensi. Dalam pengolahan citra, filter dirancang untuk menyaring informasi spectral sehingga menghasilkan citra baru yang mempunyai variasi nilai spectral yang berbeda dengan citra asli.

2.5.3 Perbaikan Kualitas Citra (*Image Enhancement*)

Perbaikan kualitas citra bertujuan untuk meningkatkan kualitas citra dengan cara memanipulasi fitur-fitur pada citra. Perbaikan citra dilakukan karena tidak semua citra masukan memiliki kelayakan untuk langsung dilakukan proses klasifikasi atau segmentasi. Ketidaklayakan tersebut bisa terjadi karena kualitas citra yang kurang baik sehingga hasil yang didapatkan dari proses klasifikasi atau segmentasi kurang akurat. Dengan adanya perbaikan kualitas citra, ciri-ciri khusus yang terdapat di dalam citra lebih ditonjolkan. Salah satu metode untuk memperbaiki kualitas suatu citra adalah *Contrast Limited Adaptive Histogram Equalization* (CLAHE).

CLAHE merupakan pengembangan dari metode AHE. *Adaptive Histogram Equalization* (AHE) merupakan teknik pemrosesan citra digital yang berguna untuk meningkatkan kontras pada sebuah citra. Metode AHE didasarkan pada metode HE (*Histogram Equalization*) bahwa metode adaptif merumuskan setiap histogram pada setiap bagian gambar untuk mendistribusikan kembali nilai intensitas gambar. Metode AHE cocok digunakan untuk meningkatkan kontras lokal dari suatu gambar dan menampilkan lebih banyak detail. Namun ada kekurangan juga yaitu hampir sama seperti pemerataan histogram secara global yaitu memperkuat noise di bagian yang relatif homogen. Untuk mengatasi masalah tersebut diusulkan metode *Contrast Limited Adaptive Histogram Equalization* (CLAHE).

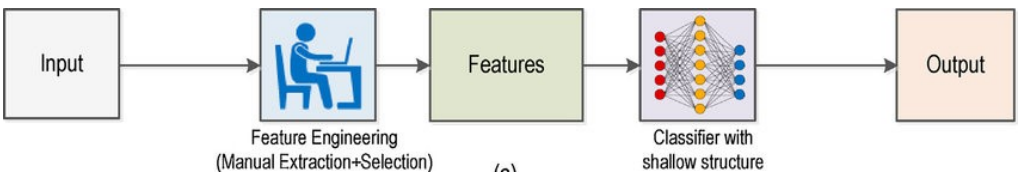
Perbaikan kontras menggunakan CLAHE ini bertujuan untuk mendapatkan citra dengan kontras yang lebih baik tanpa mengurangi kualitas dari citra itu sendiri.

Kontras merupakan persebaran gelap dan terang dalam sebuah citra. Metode CLAHE ini dapat meningkatkan kualitas sebuah citra yang memiliki kontras rendah yang kemudian akan terjadi perubahan intensitas warna. Metode CLAHE melakukan pemerataan histogram namun dengan membatasi level kontras dari citra itu sendiri untuk menghindari terjadinya peningkatan kontras secara berlebihan dengan cara memotong histogram pada nilai ambang yang telah ditentukan atau yang disebut dengan *clip limit*. Algoritma CLAHE beroperasi pada daerah kecil dalam gambar yang disebut dengan *tile* atau petak dan menghitung beberapa histogram, masing-masing sesuai dengan bagian gambar yang berbeda dan menggunakannya untuk mendistribusikan kembali nilai-nilai kontras pada gambar. Pada suatu penelitian menunjukkan bahwa metode CLAHE dapat memperbaiki citra X-Ray yang memiliki *noise* [10].

2.6 *Machine Learning*

Machine Learning (ML) atau pembelajaran mesin adalah aplikasi kecerdasan buatan (AI) yang menyediakan sistem kemampuan untuk belajar secara otomatis berdasarkan dari pengalaman tanpa deprogram secara eksplisit. Pembelajaran mesin berfokus pada pengembangan program komputer yang dapat mengakses data dan menggunakannya untuk belajar sendiri. Proses pembelajaran dimulai dengan pengamatan atau data, contohnya pengalaman langsung, atau instruksi untuk mencari pola dalam data dan membuat keputusan yang lebih baik di masa depan berdasarkan contoh yang kami berikan. Tujuan utamanya adalah untuk memungkinkan komputer belajar secara otomatis tanpa campur tangan atau bantuan manusia dan menyesuaikan tindakan yang sesuai.

Gambar 2.2 merupakan gambaran dari struktur pemodelan jaringan pada *Machine Learning*. Aplikasi utama dalam *machine learning* adalah klasifikasi dan prediksi atau regresi. Klasifikasi adalah metode dalam *machine learning* yang digunakan oleh mesin untuk memilah obyek berdasarkan ciri tertentu sebagaimana manusia mencoba membedakan benda satu dengan yang lain. Sedangkan prediksi atau regresi digunakan oleh mesin untuk menerka keluaran dari suatu data masukan berdasarkan data yang sudah dipelajari dalam pelatihan atau *training*. Ciri khas dari *machine learning* adalah adanya proses pelatihan, pembelajaran, atau *training*. Oleh karena itu *machine learning* membutuhkan data untuk belajar sehingga bisa



Gambar 2. 2 Struktur pemodelan jaringan pada *Machine Learning* [11]

juga diistilahkan dengan *learn from data*. Ada banyak algoritma *machine learning* yang dikembangkan setiap tahunnya.

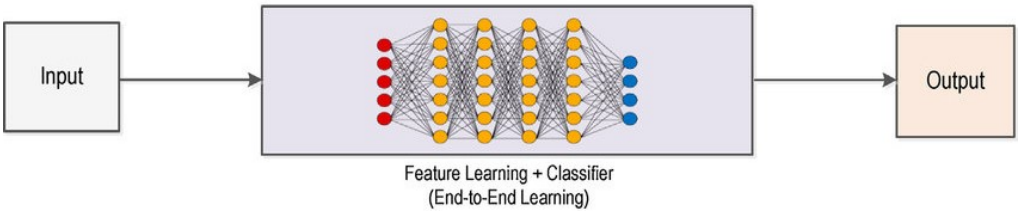
Ada banyak algoritma *machine learning* yang dikembangkan setiap tahunnya. Terdapat tiga (3) jenis algoritma pembelajaran dalam *machine learning* yaitu :

a. *Supervised learning* : algoritma ini dapat menerapkan apa yang telah dipelajari sebelumnya ke data yang baru menggunakan contoh berlabel untuk memprediksi peristiwa di masa yang akan dating.
b. *Unsupervised learning* : algoritma ini digunakan ketika informasi yang digunakan untuk melatih tidak diklasifikasikan atau dilabeli, tetapi dapat mengklasifikasikan data berdasarkan kemiripan fiturnya.
c. *Semi-supervised learning* : merupakan algoritma yang berada diantara algoritma *supervised learning* dan *unsupervised learning*, karena algoritma ini menggunakan data berlabel dan tidak berlabel untuk proses pelatihan.
d. *Reinforcement learning* : algoritma pembelajaran yang berinteraksi dengan lingkungannya dengan menghasilkan tindakan dan menemukan kesalahan atau penghargaan.

2.7 *Deep Learning*

Deep learning atau pembelajaran mendalam adalah fungsi kecerdasan buatan yang meniru cara kerja otak manusia dalam memproses data dan menciptakan pola untuk digunakan dalam pengambilan keputusan. Pembelajaran mendalam adalah salah satu cabang pembelajaran mesin (*machine learning*) dalam kecerdasan buatan (AI) yang memiliki jaringan yang mampu belajar tanpa pengawasan dari data yang tidak terstruktur atau tidak berlabel. Juga dikenal sebagai *deep neural learning* atau *deep neural network*.

Gambar 2.3 merupakan gambaran dari struktur pemodelan jaringan pada *Deep Learningi.* Algoritma pada *Deep Learning* memiliki fitur yang unik yaitu sebuah fitur yang mampu mengekstraksi secara otomatis. Hal ini berarti algoritma yang dimilikinya secara otomatis dapat menangkap fitur yang relevan sebagai keperluan dalam pemecahan suatu masalah. Algortima semacam ini sangat penting dalam sebuah kecerdasan buatan karena mampu mengurangi beban pemrograman dalam memilih fitur yang eksplisit.



Gambar 2. 3 Struktur pemodelan jaringan pada *Deep Learning* [11]

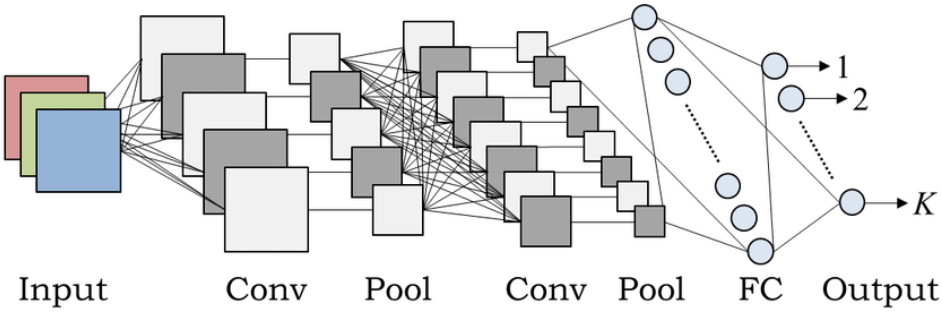
Pembelajaran mendalam telah berkembang seiring dengan era digital, yang telah menyebabkan ledakan data dalam segala bentuk dan dari setiap wilayah di dunia. Data ini, yang dikenal sebagai data besar, diambil dari sumber-sumber seperti media sosial, mesin pencari internet, platform e-commerce, dan bioskop online. Jumlah data yang sangat besar ini mudah diakses dan dapat dibagikan melalui aplikasi fintech seperti *cloud computing*. Pada tahun 2012 muncul terobosan baru pada *deep learning* yaitu sebuah *Convolutional Neural Network* (CNN) dapat melakukan pengenalan citra dengan akurasi yang menyaingi manusia pada *dataset* tertentu. *Convolutional Neural Network* dengan arsitektur tertentu yang dipadukan dengan berbagai trik *drop out regularization*, pemanfaatan *Rectified Linear Unit (ReLU)* sebagai fungsi aktivasi, *data augmentation* mampu mencapai terobosan pada *large scale image classification* (*ImageNet*) yang memiliki 1000 kategori objek dan lebih dari 1 juta gambar. Metode *deep learning* yang saat ini memiliki hasil paling signifikan dalam pengenalan citra adalah CNN. Namun CNN, seperti metode *deep learning* lainnya, memiliki kelemahan yaitu proses pelatihan model yang lama.

2.8 *Convolutional Neural Network*

Convolutional Neural Network (CNN) adalah algoritma *deep learning* dengan data masukan berupa gambar dengan menetapkan bobot dan bias yang dapat dipelajari untuk berbagai aspek atau objek dalam gambar dan dapat membedakan gambar satu dan yang lainnya. *Convolutional Neural Network* (CNN) merupakan bagian dari *artificial neural network* yang telah banyak digunakan dalam *computer vision*. CNN dirancang untuk secara otomatis dan adaptif mempelajari hierarki spasial fitur melalui *backpropagation* dengan menggunakan beberapa blok atau *layer*, yaitu lapisan konvolusi, lapisan *pooling,* dan *fully connected layer* seperti yang dijelaskan pada Gambar 2.4. Arsitektur yang terdiri dari pengulangan dari beberapan lapisan konvolusi dan lapisan *pooling*, dan diikuti oleh satu atau lebih *fully connected layer*. Langkah dimana data input ditransformasikan menjadi output melalui proses-proses tersebut dinamakan *forward propagation*. Berikut ini merupakan blok atau *layer* dalam sebuah arsitektur CNN.

a) *Convolution Layer*

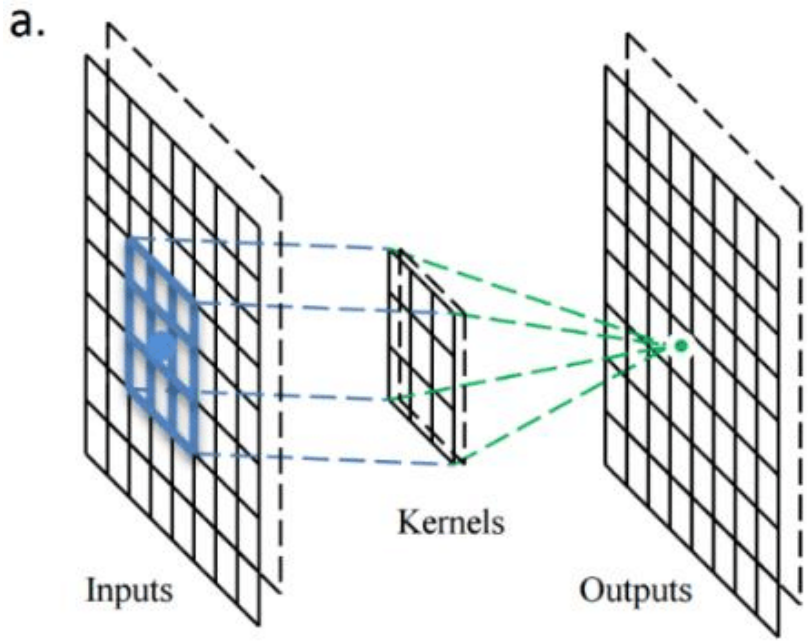
Convolution layer merupakan komponen dasar arsitektur CNN yang melakukan proses ekstraksi fitur, yang biasanya terdiri dari kombinasi operasi linier dan nonlinier. *Convolution Layer* melakukan operasi konvolusi pada *output* dari lapisan sebelumnya. Konvolusi adalah jenis operasi linier khusus yang digunakan untuk ekstraksi fitur, dimana sejumlah kecil angka yang disebut kernel diterapkan melalui input, yang merupakan array angka yang disebut tensor. Dua parameter kunci yang dapat menentukan operasi konvolusi adalah ukuran filter dan jumlah kernel. Operasi konvolusi memiliki persamaan seperti pada Persamaan 2.1.



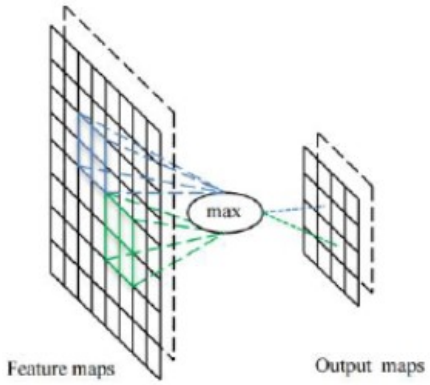
Gambar 2. 4 Arsitektur CNN [12]

$$s(t) = (x * w)(t) \tag{2.1}$$

Dari Persamaan 2.1, x merepresentasikan *input* atau masukan, w adalah kernel, dan t merupakan *timestep*. *Element-wise product* antara setiap elemen kernel dan tensor input dihitung di setiap lokasi tensor dan dijumlahkan untuk mendapatkan nilai *output* di posisi yang sesuai dari tensor *output*, yang disebut dengan *feature map* seperti yang diilustrasikan pada Gambar 2.4. Proses tersebut diulang dengan menerapkan beberapa kernel untuk membentuk sejumlah *feature map.* Selain sebagai ekstraktor fitur, lapisan konvolusi juga berpesan sebagai penentu keluaran dari jaringan.



Gambar 2. 5 Operasi Konvolusi [13]



Gambar 2. 6 Operasi *Max Pooling* [13]

b) *Pooling Layer*

Pooling layer adalah proses mereduksi ukuran dari sebuah data. Dalam pengolahan citra, *pooling* juga bertujuan untuk meningkatkan invariansi posisi dari fitur. Dalam sebagian besar CNN, metode *pooling* yang digunakan adalah *max pooling*. *Max pooling* membagi output dari *convolution layer* menjadi beberapa *grid* kecil lalu mengambil nilai maksimal dari setiap *grid* untuk menyusun matriks citra yang telah direduksi seperti yang ditunjukkan pada Gambar 2.5. Grid yang berwarna jingga, abu-abu, biru dan hijau pada Gambar 2.5 merupakan kelompok grid yang akan dipilih nilai maksimumnya. Proses tersebut memastikan fitur yang didapatkan akan sama meskipun objek citra mengalami translasi (pergeseran).

c) *Fully Connected Layer*

Fully connected layer adalah *layer* yang biasanya digunakan dalam penerapan MLP dan bertujuan untuk melakukan transformasi pada dimensi data agar data dapat diklasifikasikan secara linear. Setiap neuron pada *convolution layer* perlu ditransformasi menjadi data satu dimensi terlebih dahulu sebelum dapat dimasukkan ke dalam sebuah *fully connected layer*. Karena hal tersebut menyebabkan data kehilangan informasi spasialnya dan tidak reversibel, *fully connected layer* hanya dapat diimplementasikan di akhir jaringan.

2.9 Python

Python merupakan bahasa pemrograman yang banyak digunakan dalam berbagai macam aplikasi. Python pertama kali dikembangkan pada akhir tahun 1980 oleh Guido van Rossum. Bahasa pemrograman ini bersifat *open source*. Python telah banyak digunakan dalam pengembangan berbagai macam perangkat lunak seperti *internet scripting*, *systems programming*, *user interfaces*, *product customization*, dan masih banyak lagi [14]. Banyak fitur yang dimiliki oleh bahasa pemrograman Python seperti berorientasi pada objek, mudah untuk dikembangkan,

memiliki tata bahasa yang mudah dipelajari, dan memiliki banyak fasilitas pendukung.

2.10 Keras

Keras adalah *library neural network open source* yang berbasis pemrograman Python dan mampu bekerja berdasarkan *software* seperti *Deep learning*, TensorFlow, dan Theano. *Library* ini awalnya dirancang untuk memungkinkan eksperimen cepat dengan *deep neural network*. Keras dikembangkan sebagai bagian dari upaya penelitian proyek ONEIROS (*Open-ended Neuro-Electronic Intelligent Robot Operating System*) dan penulis serta pengelolanya adalah Francois Chollet. Pada tahun 2017, tim *Google Tensorflow* memutuskan untuk mendukung Keras dalam *TensorFlow's core library*. Keras berisi banyak implementasi dari blok jaringan saraf yang biasa digunakan seperti lapisan, tujuan, fungsi aktivasi, pengoptimal, dan sejumlah alat untuk memudahkan pekerjaan dengan data gambar dan teks.

2.11 *Google Colaboratory*

Google Colaboratory merupakan *tools* yang disediakan oleh *Google Research* yang berbasis *cloud* dan gratis yang ditujukan untuk penelitian. *Tools* ini sangat berguna bagi para peneliti atau orang yang ingin mempelajari dan mengolah data menggunakan *machine learning* atau *deep learning* namun memiliki keterbatasan dalam perangkat yang digunakan. Saat menggunakan Google Colab, pengguna akan difasilitasi dengan adanya akses *cloud* komputer, serta diberikan akses untuk menggunakan GPU secara gratis yang dapat digunakan untuk menjalankan program selama 12 jam. Google Colab dibuat dengan menggunakan *environtment* jupyter notebook sehingga hampir sama dengan jupyter notebook yang menggunakan Python sebagai bahasa pemrogramannya. Google Colab mendukung hampir semua *library* yang biasanya digunakan dalam *machine learning*. Google Colab dapat diintegrasikan dengan Google Drive sehingga dapat mengakses atau menyimpan data ke dalam Google Drive.

2.12 *Graphical User Interface* (GUI)

Graphical User Interface (GUI) merupakan antarmuka yang digunakan untuk memudahkan pengguna dalam berinteraksi dengan perangkat elektronik seperti komputer, telepon genggam, dan perangkat elektronik lainnya. Antarmuka ini dapat menggunakan ikon, menu, kursor, tombol, dan representasi visual lainnya untuk menampilkan informasi dan kontrol pengguna. GUI dianggap lebih mudah digunakan dibandingkan dengan antarmuka berbasis teks seperti MS-DOS. GUI sekarang merupakan antarmuka komputer standar. GUI pertama kali dikembangkan oleh Xerox PARC oleh Alan Kay, Douglas Engelbart, dan sekelompok peneliti lain pada tahun 1981.

2.13 Rasional

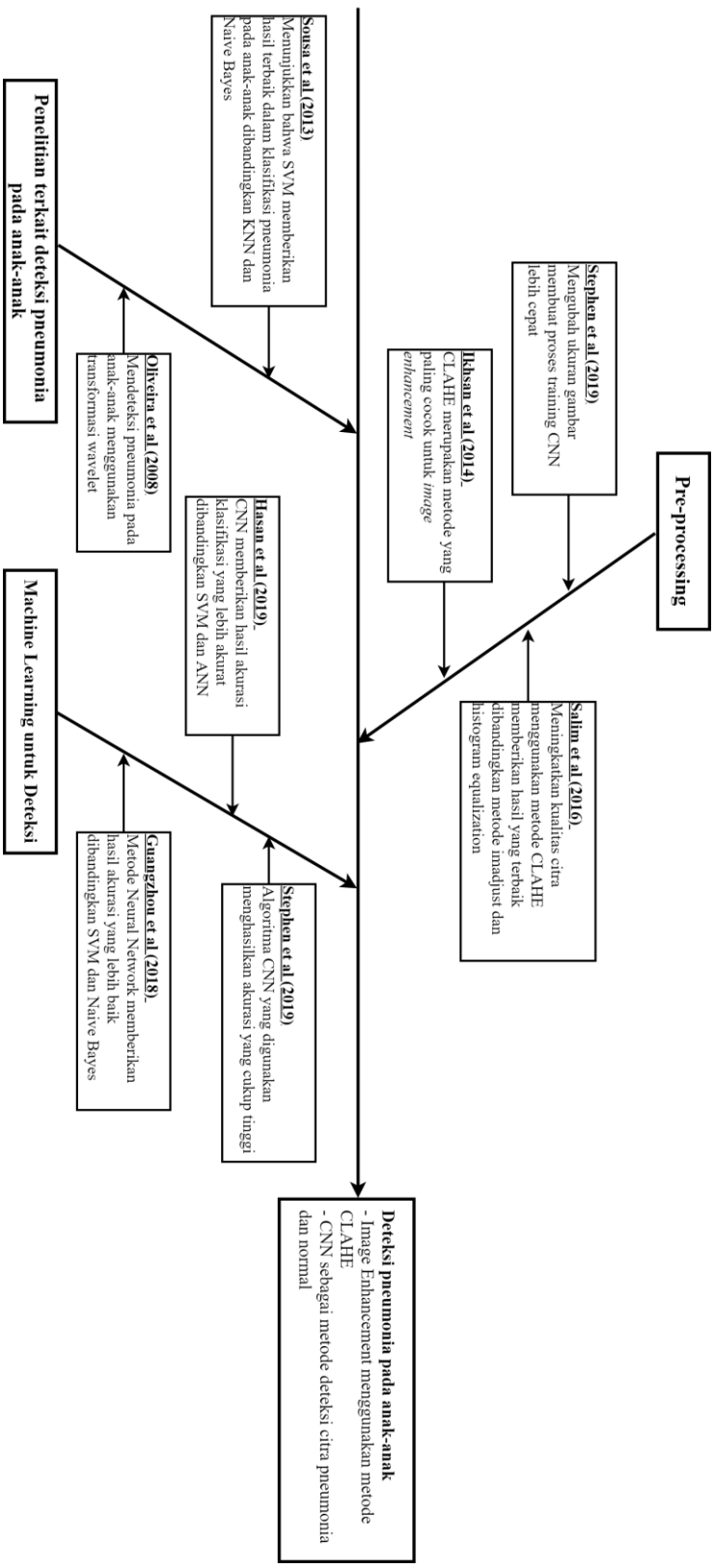
Secara umum, rasional dari penelitian ini digambarkan dalam *fish bone diagram* pada Gambar 2.7. Seperti yang sudah dijelaskan pada subbab sebelumnya, bahwa selama tenaga medis pada bidang radiologi masih mengandalkan kemampuan visualnya dalam menganalisa hasil X-Ray. Oleh karena itu, sistem klasifikasi citra itu diperlukan untuk mengurangi faktor subjektivitas tersebut. Terdapat beberapa referensi yang terkait dengan penelitian ini. Pertama penelitian dari Leandro L. G. Oliveira, *et al* (2008) yang berjudul "*Computer-aided diagnosis in chest radiography for detection of childhood pneumonia*.", Leandro dalam penelitiannya melakukan deteksi pneumonia pada anak-anak dengan menggunakan skema dan transformasi wavelet. Sistem dalam penelitiannya diberi nama Pneumo-CAD, system tersebut menggunakan wavelet Haar dan memberikan akurasi yang baik dalam membedakan citra dengan pneumonia atau tidak adanya pneumonia.

Selanjutnya penilitian dari Rafael T. Sousa, *et al* (2013) yang berjudul "*Comparative performance analysis of machine learning classifiers in detection of childhood pneumonia using chest radiographs*". Dalam penelitian tersebut, peneliti membandingkan tiga metode *machine learning* untuk mengklasifikasikan dan mendeteksi pneumonia pada anak-anak. Tiga metode tersebut adalah *Naïve bayes, K-Nearest Neighbor* (KNN), dan *Support Vector Machine* (SVM). Hasil yang diperoleh dari penelitian tersebut menyatakan bahwa metode SVM memberikan hasil klasifikasi yang lebih baik dibandingkan 2 metode lainnya. Akan tetapi pada penelitian yang berjudul "*Comparison of Machine Learning Classification Model for Glaucoma Management"* dari Guangzhou An, *et al* (2018) menyatakan bahwa metode klasifikasi menggunakan *Neural Network* (NN) akurasi validasinya lebih baik dibandingkan metode SVM dan *Naïve Bayes.*

Kemudian penelitian dari M. ul Hasan (2019), yang berjudul "*Comparative Analysis of SVM, ANN, and CNN for Classifying Vegetation Species Using Hyperspectral Thermal Infrared Data".* Dalam penelitian tersebut membandingkan tiga metode *machine learning* untuk mengklasifikasi vegetasi dan hasilnya menunjukkan bahwa CNN memberikan akurasi klasifikasi yang lebih tinggi dibandingkan SVM dan ANN dengan nilai akurasi yang diperoleh yaitu 99%. Penelitian dari O. Stephen, *et al* (2019) yang berjudul "*An efficient deep learning approach to pneumonia classification in healthcare".* Dalam penelitian tersebut peneliti melakukan klasifikasi terhadap citra X-Ray toraks pada anak-anak menggunakan *Convolutional Neural Network*. Algoritma yang diusulkan dimulai dengan mentransformasi citra X-ray ke dalam ukuran yang lebih kecil dibandingkan ukuran citra aslinya, kemudian dilakukan proses identifikasi dan klasifikasi menggunakan CNN. Dalam penelitiannya O. Stephen menunjukkan bahwa semakin besar ukuran gambar maka proses *training* dan biaya yang dikeluarkan juga lebih banyak. Kemudian penelitian dari I. A. M. Ikhsan, *et al* (2014) dalam penelitiannya yang berjudul "*An Analysis of X-Ray Image*

Enhancement Methods for Vertebral Bone Segmentation" menunjukkan bahwa metode CLAHE (*contrast limited adaptive histogram*) merupakan algortima yang paling cocok untuk *image enhancement* pada citra X-ray. Ditambahkan dengan penelitian dari J. Salim (2016) dalam penelitiannya yang berjudul "*Enhancement of Chest X-ray Images for Diagnosis Purposes*" yang membandingkan tiga metode *image enhancement* yaitu *imadjust*, *Histogram Equalization* (HE), dan *Contrast Limited Adaptive Histogram Equalization* (CLAHE). Hasil yang didapatkan dari penelitian tersebut adalah metode peningkatan kualitas citra menggunakan CLAHE hasilnya lebih baik meskipun kontras yang dihasilkan kurang cerah tetapi detail gambar yang dihasilkan pada metode tersebut lebih baik dibandingkan dua metode lainnya.

Dari penjabaran beberapa metode yang ada, maka deteksi pneumonia dapat dilakukan dengan menggunakan serangkaian proses tersebut. *Contrast limited Adaptive Histogram Equalization* sebagai proses dari tahap pemrosesan awal atau *pre-processing* citra untuk mempertajam kualitas dari citra X-ray. Dan yang terakhir melakukan proses deteksi menggunakan metode CNN yang memiliki tingkat akurasi lebih baik untuk pengolahan citra. Dari proses tersebut akan mempunyai 2 keluaran atau *output* yaitu citra normal dan pneumonia.



Gambar 2. 7 Diagram *Fishbone*

BAB III PERANCANGAN SISTEM

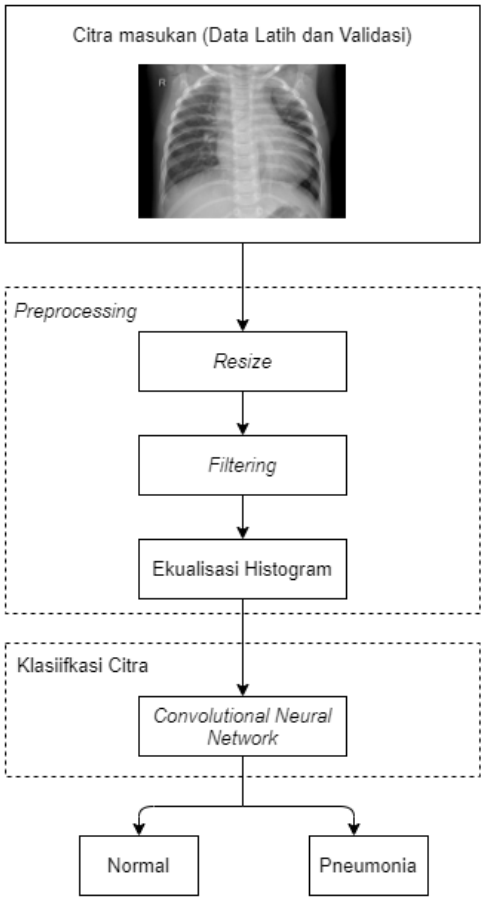
Dalam tugas akhir ini dilakukan beberapa tahapan atau langkah-langkah. Tahapan-tahapan tersebut berhubungan satu sama lain secara sistematis. Tahapan ini bertujuan untuk memudahkan dan menguraikan kegiatan yang dilaksanakan selama penelitian. Kerangka kerja dalam tugas akhir ini dapat dibagi menjadi tiga bagian yaitu : (1) tahap persiapan *dataset*, (2) tahap *pre-processing* citra, (3) perancangan arsitektur CNN untuk mendeteksi citra pneumonia dan normal. Serangkaian kerangka kerja ini merupakan bagian dari bidang penelitian pengolahan citra medis yang berfokus pada pengolahan citra yang sudah ada dengan menggunakan modalitas X-ray untuk mendapatkan informasi tertentu. Bab ini akan menjelaskan kerangka kerja yang dilakukan dalam tugas akhir ini.

3.1 Desain Sistem Secara Umum

Secara umum, alur kerja sistem dalam tugas akhir ini dijelaskan dalam diagram blok pada Gambar 3.1. Pada tugas akhir ini, terdapat beberapa langkah untuk mendeteksi pneumonia pada anak-anak dari citra X-ray. Langkah-langkah tersebut diuraikan dalam tiga tahapan. Tahap pertama adalah persiapan *dataset* berupa citra X-ray dada yang kemudian dilanjutkan dengan tahapan pengolahan citra. Tahap kedua dalam tugas akhir ini adalah tahapan *image pre-processing* dimana citra X-ray yang didapatkan diolah untuk mengubah ukuran dan dilakukan proses *image enhancement* untuk meningkatkan kontras serta mereduksi *noise* pada citra. Tahap ketiga adalah proses klasifikasi citra menggunakan CNN. Proses *learning* yang dilakukan dengan menggunakan CNN digunakan untuk mendeteksi citra X-ray apakah citra tersebut termasuk dalam kategori normal atau pneumonia.

Pada subbab 3.2 akan menjelaskan mengenai *dataset* yang digunakan untuk melatih CNN dan menguji sistem secara umum. Subbab 3.3 akan menjelaskan mengenai *image pre-processing* untuk mengubah ukuran dan melakukan proses peningkatan kualitas pada citra. Subbab 3.4 akan menjelaskan mengenai proses perancangan sistem CNN yang akan digunakan untuk proses klasifikasi citra yang nantinya akan mengidentifikasi citra menjadi dua kelas yaitu normal dan pneumonia.

.



Gambar 3. 1 Diagram Blok Sistem Secara Umum

3.2 *Dataset*

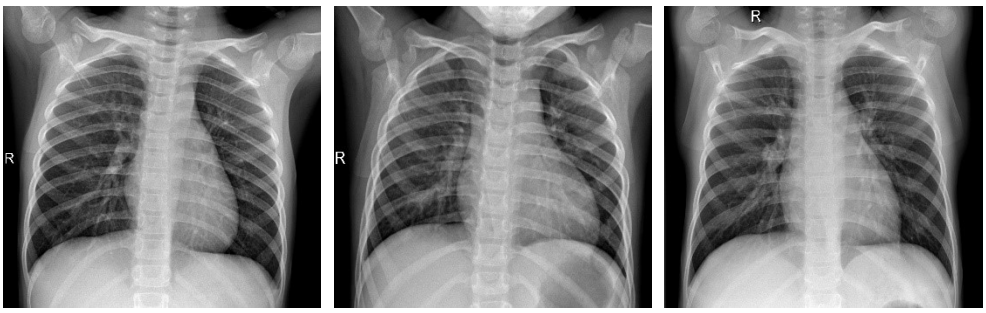
Dataset yang digunakan dalam penelitian ini digunakan untuk melatih CNN dan menguji sistem secara umum. *Dataset* yang digunakan didapatkan dari penelitian sebelumnya yang dilakukan oleh Kermany *et al* [15].

3.2.1 Keterangan *Dataset*

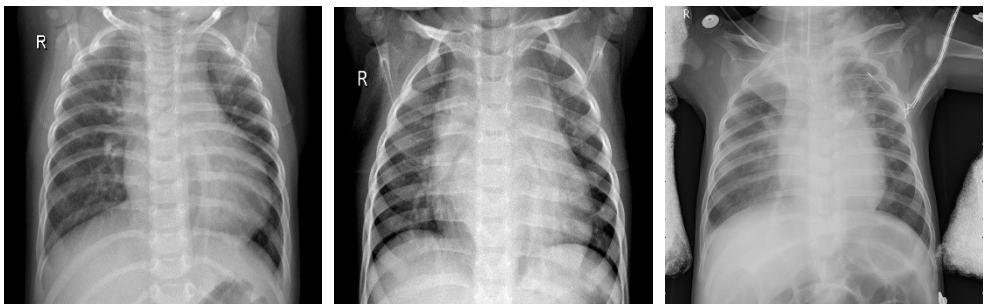
Dataset tersebut merupakan citra X-ray dada dalam posisi *posteoanterior* (PA) dari pasien pediatrik (anak-anak) yang berusia satu hingga lima tahun yang berasal dari *Guangzhou Women and Children's Medical Center*. Citra dalam *dataset* tersebut memiliki format .jpeg dan setiap citra memiliki ukuran piksel yang berbeda. *Dataset* tersebut tersusun dalam 3 folder utama yaitu *train, test,* dan *val*, masing-masing berisi dua subfolder yaitu folder "Normal" yang berisi citra X-ray normal dan folder "Pneumonia" yang berisi citra X-ray dari pasien yang menderita pneumonia. Dalam *dataset* tersebut terdapat 5863 gambar X-ray dengan rincian citra normal sebanyak 1583 dan citra pneumonia sebanyak 4273. Dari rincian tersebut dapat disimpulkan bahwa dalam *dataset* tersebut jumlah citra X-ray

Tabel 3. 1 Distribusi *Dataset*

Kategori	*Train*	*Test*	*Validation*
Normal	1341	234	8
Pneumonia	3875	390	8
Total	5216	624	16

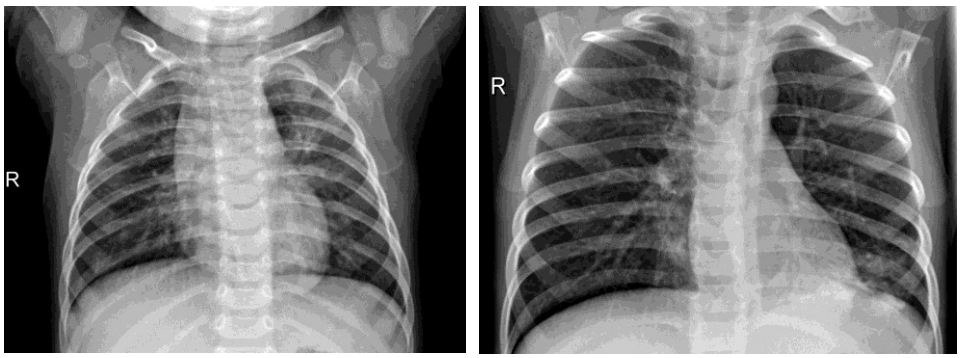


Gambar 3. 2 Sampel citra paru-paru normal dari *dataset*



Gambar 3. 3 Sampel citra paru-paru pneumonia

pneumonia lebih besar dibandingkan dengan citra X-ray normal. Citra X-ray normal menggambarkan paru-paru yang jernih tanpa adanya bercak atau tanda-tanda keabnormalan. Sedangkan pada citra X-ray dari pasien yang terkena pneumonia memiliki beberapa bentuk kekeruhan. Hal tersebut bisa berupa konsolidasi lobus fokal untuk pneumonia akibat bakteri atau beberapa pola difus untuk pneumonia yang diakibatkan oleh virus. Gambar 3.2 merupakan sampel citra paru-paru normal dan Gambar 3.3 sampel untuk citra paru-paru dengan pneumonia dari *dataset* yang didapatkan. Berdasarkan Gambar 3.2 dan Gambar 3.3 jika dilihat secara visual terlihat jelas perbedaan antara citra normal dengan citra pneumonia, namun jika dilihat secara keseluruhan pada *dataset* ada beberapa citra yang sulit dibedakan antara citra normal dengan pneumonia. Gambar 3.4 merupakan sampel citra normal dan pneumonia yang terlihat cukup sulit dibedakan. Tabel 3.1 merepresentasikan distribusi dari masing-masing folder dan subfolder pada *dataset* tersebut.



(a) (b)

Gambar 3. 4 Sampel citra paru-paru yang sulit dibedakan (a) Normal; (b) Pneumonia

3.2.2 Pembuatan Data Latih dan Data Validasi

Sebelum *dataset* digunakan dalam tahapan selanjutnya, *dataset* tersebut yang sebelumnya terdiri dari 3 folder dibagi menjadi 2 folder saja yaitu *train* dan *val*. Proses pembagian ini dilakukan karena perbandingan antara jumlah data tiap folder terlalu jauh atau tidak seimbang. Proses pembagian tersebut dilakukan secara acak menggunakaan program. Pertama semua citra dari masing-masing kategori (normal dan pneumonia) digabungkan dalam satu folder dengan kategori yang sama. Kemudian data tersebut dibagi ke dalam dua folder yaitu *train* dan *val* dengan masing-masing folder terdiri dari 2 subfolder yaitu normal dan pneumonia. Pembagian ini dilakukan dengan perbandingan data pneumonia dan normal sebesar 7:3 untuk setiap folder (*train* dan *val*). Sebanyak 3722 citra dialokasikan untuk set pelatihan (*train*) dan 2134 citra untuk set validasi (*val*). Pembagian jumlah citra tersebut disesuaikan dengan referensi [9]. Sehingga distribusi dari *dataset* yang digunakan dalam tugas akhir ini dapat dilihat pada Tabel 3.2.

3.3 *Image Pre-Processing*

Tahap *pre-processing* merupakan bagian yang penting sebelum citra diolah dalam *machine learning*. Tahap ini diperlukan untuk meningkatkan kualitas citra X-ray. Citra dari *dataset* yang sudah didapatkan dan dimodifikasi seperti yang dijelaskan pada subsubbab 3.2.2 menjadi citra *input* atau masukan dalam tahapan

Tabel 3. 2 Distribusi *dataset* yang baru

Kategori	*Train*	*Validation*
Normal	1117	466
Pneumonia	2605	1668
Total	3722	2134

ini. Sebelum dilakukan pembagian data latih dan validasi, *dataset* tersebut diunggah ke dalam *Google Drive* yang kemudian dihubungkan dengan *Google Colaboratory* agar *dataset* tersebut dapat dibaca dan digunakan dalam *Google Colaboratory*. Dalam tahap *pre-processing* pada tugas akhir ini ada beberapa percobaan seperti *filtering* dan *histogram equalization* terhadap citra. Setiap percobaan *preprocessing* yang dilakukan akan berpengaruh terhadap performa pembelajaran CNN. Kerangka kerja dalam tahap *pre-processing* diilustrasikan dalam *flowchart* pada Gambar 3.5.

3.3.1 *Filtering*

Tahapan pertama adalah melakukan proses *resize* yaitu proses perubahan ukuran citra menjadi ukuran yang lebih kecil. Proses ini dilakukan terlebih dahulu pada citra *input* untuk berkas latih dan validasi untuk mengubah ukuran citra ke ukuran yang diperlukan. Proses selanjutnya adalah melakukan proses *filtering* pada citra untuk mengurangi *noise* yang ada. Jenis filter yang digunakan dalam tugas akhir ini adalah filter median. Penggunaan filter ini berfungsi untuk menghaluskan dan mengurangi *noise* atau gangguan pada citra. Filter median ini sangat popular karena untuk tipe-tipe *noise* tertentu, filter ini memberikan kemampuan untuk mengurangi *noise* yang sangat baik [16].

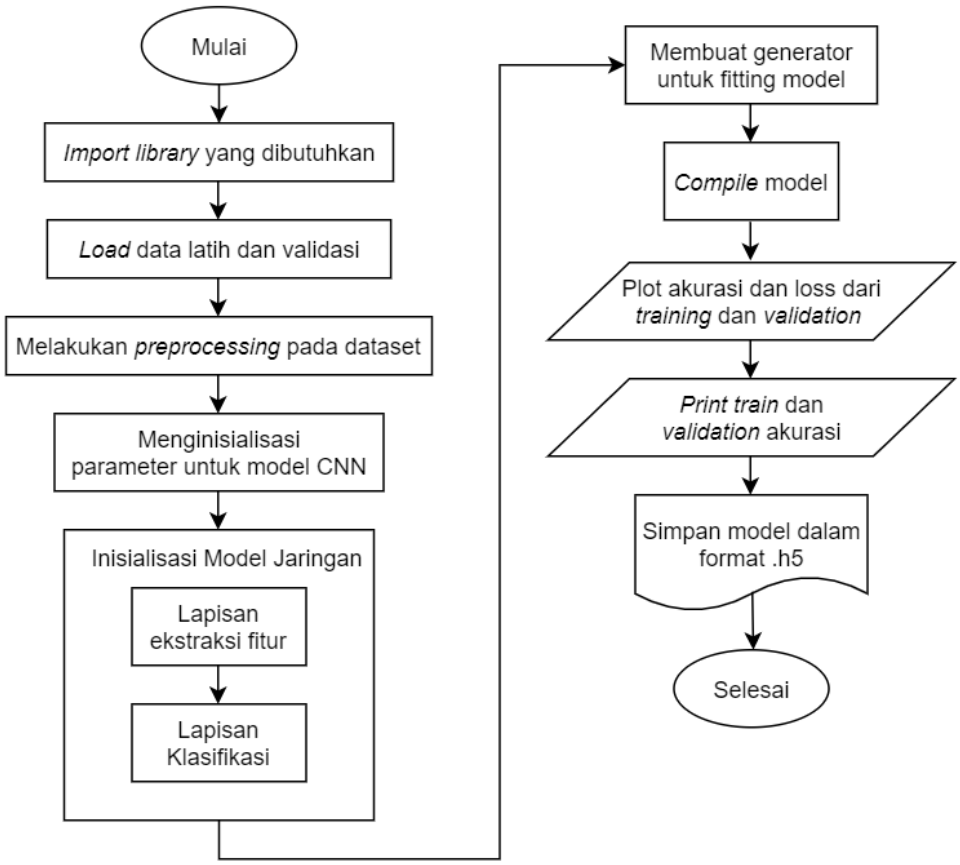
Gambar 3. 5 Diagram Alir Tahap *Preprocessing*

3.3.2 *Histogram Equalization*

Tahap berikutnya adalah melakukan *histogram equalization*. *Histogram equalization* merupakan suatu proses pemerataan histogram, dimana distribusi nilai derajat keabuan pada suatu citra dibuat rata. Proses *histogram equalization* dalam tugas akhir ini menggunakan metode *Contrast Limited Adaptive Histogram Equalization* yang biasa disingkat dengan CLAHE. CLAHE merupakan pengembangan dari metode *Adaptive Histogram Equalization* (AHE). Metode ini dapat menghasilkan citra yang kualitasnya lebih baik dibandingkan dengan citra aslinya. Proses ini dilakukan untuk menyesuaikan intensitas gambar sehingga dapat meningkatkan kontras suatu citra secara lokal sehingga memunculkan bagian-bagian yang tidak terlihat serta untuk memperbaiki kualitas citra. Metode ini dilakukan dengan menginisiasi nilai ambang batas atau *clip limit* pada histogram untuk penambahan kontras pada citra. *Clip limit* dalam tahap *preprocessing* ini ditetapkan sebesar 0.01. Penetapan nilai *clip limit* ini dilakukan dengan melakukan pengujian terhadap performa algoritma CNN dengan melihat pengaruh nilai *clip limit*. Metode CLAHE tersebut digunakan untuk meningkatkan pendeteksian *clouds* atau adanya konsolidasi pada citra X-ray dari pasien pneumonia.

3.4 *Convolutional Neural Network* (CNN)

Tahap ini merupakan proses untuk mengklasifikasikan citra X-ray ke dalam dua kategori yaitu normal dan pneumonia. Algoritma *machine learning* yang digunakan dalam tugas akhir ini adalah algoritma *supervised learning* karena *dataset* yang digunakan sudah berlabel dan adanya proses *training* untuk melatih CNN dalam mengenali *dataset* yang digunakan agar dapat dilakukan proses prediksi atau klasifikasi. *Dataset* yang sudah melalui tahap *pre-processing* menjadi citra masukan untuk tahap CNN. Citra masukan untuk arsitektur yang digunakan ditetapkan dengan ukuran 200 x 200 piksel sesuai dengan referensi [9]. Arsitektur CNN yang digunakan dalam tugas akhir ini berasal dari penelitian sebelumnya yang dilakukan oleh O. Stephen *et al* [9] dengan sedikit modifikasi untuk mendapatkan akurasi yang lebih baik. Dalam subbab ini akan dijelaskan perancangan sistem untuk klasifikasi citra. Secara umum alur kerja pada tahap ini terdiri dari perancangan arsitektur dan perancangan proses pembelajaran pada CNN. Keseluruhan rancangan sistem dalam tugas akhir ini diilustrasikan pada Gambar 3.6.



Gambar 3. 6 Diagram alir sistem klasifikasi

3.4.1 Perancangan Arsitektur CNN

Tabel 3.3 menyajikan arsitektur CNN berdasarkan referensi yang digunakan dengan adanya modifikasi. Sebelumnya dilakukan percobaan dengan memodifkasi arsitektur yang ada dan performa dari masing-masing arsitektur yang telah dicoba sebelumnya sudah terlampir pada halaman lampiran. Arsitektur yang digunakan dalam tugas akhir ini terdiri dari lapisan ekstraksi fitur dan lapisan klasifikasi yang digabungkan bersama. Secara total terdapat delapan lapisan konvolusi, empat lapisan maxpool, dua lapisan *dropout*, dan dua lapisan *dense*. Seperti yang diilustrasikan pada Tabel 3.3, lapisan konvolusi memiliki kernel konvolusi berukuran 3x3 yang diikuti dengan *Rectified Linear Unit* (ReLU) sebagai fungsi aktivasi. Setiap dua lapisan konvolusi dengan jumlah filter yang sama ditambahkan lapisan maxpool berukuran 2x2. Setiap setelah lapisan maxpool, jumlah filter dalam lapisan konvolusi ditingkatkan sebanyak dua kali. Hal tersebut dilakukan untuk memastikan kira-kira adakah jumlah perhitungan yang sama dalam jaringan. Lapisan maxpool yang ditambahkan setelah lapisan konvolusi dalam arsitektur CNN dapat mengurangi ukuran volume keluaran pada *feature map*, sehingga

Tabel 3. 3 Arsitektur CNN

Input	***Transformed Images***	
	200 x 200 x 1	
Layer	***Type***	***Details***
1	*Convolutional layer 1*	*Number of filter = 32, Kernel size = (3x3), Activation = ReLU*
2	*Convolutional layer 2*	*Number of filter = 32, Kernel size = (3x3), Activation = ReLU*
3	*Maxpooling layer 1*	*Pool size = (2x2)*
4	*Convolutional layer 3*	*Number of filter = 64, Kernel size = (3x3), Activation = ReLU*
5	*Convolutional layer 4*	*Number of filter = 64, Kernel size = (3x3), Activation = ReLU*
6	*Maxpooling layer 2*	*Pool size = (2x2)*
7	*Convolutional layer 5*	*Number of filter = 128, Kernel size = (3x3), Activation = ReLU*
8	*Convolutional layer 6*	*Number of filter = 128, Kernel size = (3x3), Activation = ReLU*
9	*Maxpooling layer 3*	*Pool size = (2x2)*
10	*Convolutional layer 7*	*Number of filter = 256, Kernel size = (3x3), Activation = ReLU*
11	*Convolutional layer 8*	*Number of filter = 256, Kernel size = (3x3), Activation = ReLU*
12	*Maxpooling layer 4*	*Pool size = (2x2)*
13	*Flatten*	
14	*Dropout layer*	*Dropout rate = 0.5*
15	*Dense layer 1*	*Units = 512, Activation = ReLU*
16	*Dense layer 2*	*Units = 1, Activation = Sigmoid*

mengurangi jumlah parameter dan perhitungan di jaringan untuk mengendalikan *overfitting*.

Setelah lapisan konvolusi dan maxpool, selanjutnya adalah lapisan klasifikasi. Lapisan ini berfungsi untuk mengklasifikasikan tiap neuron yang telah dilakukan proses ekstraksi fitur pada lapisan konvolusi dan maxpool. Lapisan ini terdiri dari *flatten*, lapisan *dropout* dan lapisan *dense* atau *fully connected*. Lapisan *flatten* digunakan untuk membentuk ulang *feature map* karena *feature map* yang

dihasilkan pada lapisan ekstraksi fitur berbentuk array multidimensi sehingga harus diubah menjadi sebuah vektor agar dapat digunakan sebagai masukan pada lapisan *fully connected*, lapisan *flatten* diikuti oleh sebuah lapisan *dropout* yang berukuran 0.5, *dropout* berfungsi untuk mencegah terjadinya *overfitting* dan mempercepat proses *learning* pada jaringan saraf. Setelah lapisan *flatten* dan *dropout*, terdapat dua lapisan *dense*. Lapisan *dense* pertama terdiri dari 512 unit yang menggunakan sebuah fungsi aktivasi ReLU. Lapisan *dense* kedua memiliki 1 unit dan diikuti oleh Sigmoid sebagai fungsi aktivasinya. Pada lapisan *dense* kedua menghasilkan keluaran berupa kelas yang diprediksi dalam hal ini adalah "Normal" dan "Pneumonia".

3.4.2 Perancangan Proses Pelatihan pada CNN

Sebelum dilakukan proses pelatihan atau *training*, teknik augmentasi data dilakukan guna meningkatkan keragaman data. Teknik augmentasi data dapat dilakukan dengan melakukan pemotongan, pergeseran, rotasi, atau membalik data, dan lain-lain. Dalam tugas akhir ini digunakan beberapa teknik augmentasi data. Operasi augmentasi data yang pertama adalah melakukan *rescale* 1/255 yaitu membagi nilai RGB dari 0-255 dengan 255, sehingga nilai RGB yang didapatkan berada pada rentang 0-1. Kemudian selain *rescale* dilakukan proses rotasi sebesar 40 derajat. Proses *width* dan *height shift* juga digunakan untuk proses augmentasi data yang ditetapkan sebesar 0,2. Selanjutnya, proses yang dilakukan adalah *shear range, zoom range,* dan *horizontal flip*. Tabel 3.4 merupakan rangkuman mengenai metode-metode yang dilakukan untuk augmentasi data. Teknik augmentasi data ini membantu dalam membuat beberapa versi dari gambar yang sama dan meningkatkan ukuran *dataset* yang digunakan untuk pelatihan. Metode *flow_from_directory* dari ImageDataGenerator digunakan untuk mengubah "raw image" menjadi *dataset* yang akan digunakan untuk proses pelatihan atau *training*.

Setelah data dilakukan proses augmentasi, selanjutnya data akan dilatih. Pada proses pelatihan ini terdapat beberapa *hyperparameter* yang telah ditentukan sebelumnya yang tertera pada Tabel 3.5. Citra masukan pada CNN ditentukan sebagai sebuah array berukuran 200 x 200 x 3. Model jaringan pada tugas akhir ini dirancang untuk memecahkan masalah klasifikasi biner, sehingga *binary cross entropy* digunakan sebagai *loss function* untuk mengevaluasi penggunaan model pada *dataset* yang digunakan. Parameter lain yang ditetapkan adalah optimasi *Adam* sebagai pengoptimal bobot model saat proses pelatihan dengan mengurangi nilai jarak dari hasil *binary cross entropy*. *Adaptive Moment Estimation* (Adam) merupakan metode optimasi yang menghitung tingkat pembelajaran secara adaptif untuk setiap parameter. Seperti halnya Adadelta dan RMSProp, optimasi *Adam* menyimpan rata-rata *gradient* proses sebelumnya secara eksponensial. Optimasi *Adam* merupakan teknik sederhana dan efisien secara komputasi untuk optimalisasi fungsi kerugian atau *loss function* berbasis gradien [17]. Hasil empiris yang

ditunjukkan dalam [17], menunjukkan bahwa pengoptimal *Adam* memiliki kemampuan yang lebih baik dibandingkan metode optimasi stokastik lainnya.

Tingkat pembelajaran atau *learning rate* dalam model jaringan ini juga ditetapkan di awal yaitu sebesar 0.0001. Tingkat pembelajaran merupakan suatu parameter yang berfungsi untuk meningkatkan kecepatan belajar dari suatu model jaringan. Selanjutnya *hyperparameter* yang ditentukan di awal adalah jumlah *epoch*. *Epoch* adalah suatu kondisi ketika seluruh *dataset* sudah melewati proses pembelajaran yaitu melalui proses *forward* dan *backward* dalam sekali putaran. Jumlah *epoch* yang digunakan dalam pembelajaran ini ditetapkan sebanyak 100 *epoch*. Penentuan jumlah *epoch* dan *learning rate* ini dilakukan dengan melakukan pengujian dengan melihat pengaruh jumlah *epoch* dan *learning rate* terhadap performa model untuk mendapatkan performa CNN yang lebih baik. Karena ukuran satu *epoch* untuk dimasukkan ke dalam komputer itu terlalu besar maka dari itu perlu membaginya menjadi satuan yang lebih kecil yang dikenal dengan *batch*. Dalam satu *batch* terdapat *batch size* yang merupakan total jumlah *dataset* yang digunakan dalam satu *batch*. Dalam tugas akhir ini *batch size* juga termasuk dalam *hyperparameter* yang ditetapkan di awal. Dalam suatu model jaringan juga dikenal istilah iterasi. Iterasi adalah jumlah *batch* yang dibutuhkan untuk menyelesaikan satu *epoch*.

Tabel 3. 4 Pengaturan proses augmentasi data

Metode	Pengaturan
Rescale	1/255
Rotation	40
Width shift	0,2
Height shift	0,2
Shear range	0,2
Zoom range	0,2
Horizontal flip	*True*

Tabel 3. 5 *Hyperparameter* untuk Pembelajaran CNN

Parameter	Keterangan
Loss Function	Binary Cross Entropy
Optimasi	Adam
Tingkat pembelajaran	0,0001
Jumlah *Epoch*	100
Batch Size	16

3.5 Metode Pengujian

Pengukuran kinerja suatu sistem merupakan hal yang penting. Kinerja sistem ini menggambarkan seberapa baik sistem dalam mengklasifikasikan data. Sehingga setelah serangkaian tahapan dalam pembuatan sistem dilakukan, selanjutnya melakukan pengujian terhadap keseluruhan sistem untuk mengetahui seberapa besar kinerja sistem. Pengujian dilakukan dengan menggunakan input citra validasi yang berasal dari folder *val* yang sudah dilakukan *preprocessing*. Citra tersebut kemudian diujikan kepada sistem untuk mendapatkan hasil berupa pendeteksian menggunakan *confusion matrix*. *Confusion matrix* merupakan suatu metode untuk mengukur kinerja suatu sistem klasifikasi. Metode ini mengandung informasi yang membandingkan hasil klasifikasi yang dilakukan oleh sistem dengan data klasifikasi yang sebenarnya. Pada pengukuran kinerja menggunakan *confusion matrix* terdapat empat istilah yang digunakan. Keempat istilah tersebut adalah *True Positive* (TP), *False Positive* (FP), *True Negative* (TN), dan *False Negative* (FN).

Berdasarkan *confusion matrix* yang didapatkan maka dapat dihitung berbagai *performance metrics* untuk mengukur kinerja sistem yang telah dibuat. Beberapa *performance metrics* yang dihitung dalam pengujian ini adalah *precission*, *recall* atau *sensitivity*, dan *specificity*. Presisi merupakan tingkat keakuratan antara data yang diminta dengan hasil prediksi yang diberikan oleh model, untuk mendapatkan nilai presisi digunakan Persamaan 3.1. Persamaan 3.2 digunakan untuk menghitung nilai *recall*. *Recall* atau *sensitivity* adalah keberhasilan model dalam mendapatkan kembali sebuah informasi. Persamaan 3.3 untuk mendapatkan nilai spesifisitas dari model.

Sedangkan akurasi dalam sistem ini diukur dengan menggunakan fungsi yang disediakan dalam Keras yaitu *model.evaluate()*. Akurasi yang dihitung adalah akurasi untuk *train* dan *validation*. Jika menghitung nilai akurasi menggunakan tabel *confusion matrix* nilainya sama dengan nilai akurasi dari validasi ketika menggunakan fungsi dari Keras, karena data yang diujikan dengan data yang dilatih sama. Kurva ROC (*Receiver Operating Characteristics*) juga digunakan dalam tahap pengujian sistem. ROC merupakan metode atau suatu *tools* yang digunakan untuk mengukur performa dari sistem klasifikasi. Kurva ROC dibuat berdasarkan nilai yang telah didapatkan pada *confusion matrix* yaitu antara *False Positive Rate* dengan *True Positive Rate*. Persamaan 3.4 merupakan cara untung menghitung nilai FPR dan persamaan 3.5 untuk menghitung nilai TPR.

$$Precission = \frac{TP}{FP+TP} x\ 100\% \tag{3.1}$$

$$Recall = \frac{TP}{FN+TP} x\ 100\% \tag{3.2}$$

$$Specificity = \frac{TN}{FP+TN} x\ 100\% \quad (3.3)$$

$$False\ Positive\ Rate\ (FPR) = \frac{FP}{FP+TN} \quad (3.4)$$

$$True\ Positive\ Rate\ (FPR) = \frac{TP}{TP+FN} \quad (3.5)$$

BAB IV PENGUJIAN SISTEM

Setelah sistem selesai dirancang selanjutnya dilakukan pengujian untuk mengukur performa dari sistem. Pengujian ini dilakukan pada setiap tahapan dalam sistem yang dirancang, meliputi tahapan *image preprocessing* dan tahapan pembelajaran CNN. Bab ini akan menjelaskan mengenai hasil pengujian pada sistem.

4.1 *Image Pre-Processing*

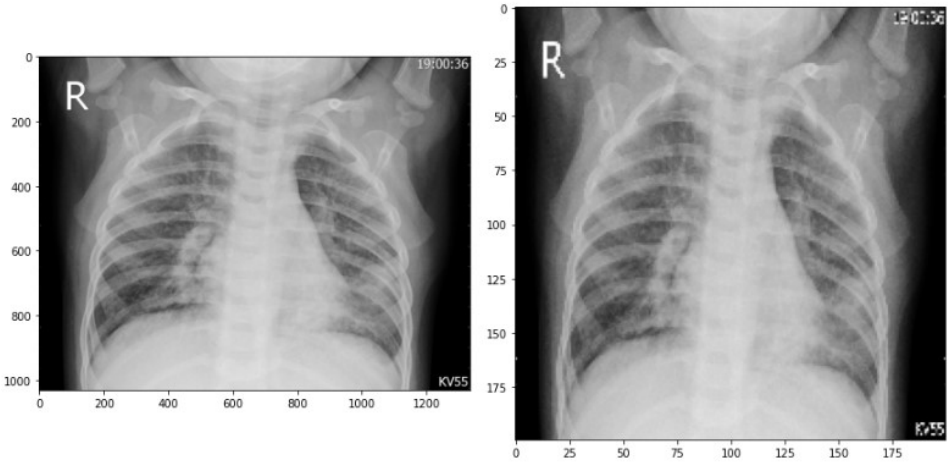
Pada tahap *image pre-processing*, citra *input* atau masukan yang digunakan diambil dari *dataset* yang sudah dijelaskan sebelumnya pada subsubbab 3.2.2. Dari *dataset* tersebut, dipilih salah satu citra masukan yaitu person49_virus_101.jpeg. Gambar 4.1 merupakan citra person49_virus_101.jpeg yang digunakan sebagai masukan. Citra tersebut kemudian dipanggil dan dijalankan dalam program untuk dilakukan proses *filtering* dan *histogram equalization*. Lokasi citra berada di Google Drive sehingga dituliskan secara manual dalam *source code* sesuai dengan lokasinya. Citra tersebut dibuka dan dibaca menggunakan *library* skimage yang tersedia pada Python sebagai sebuah matriks dengan ukuran [lebar, panjang, 3]. Angka 3 di sini merepresentasikan warna dari citra tersebut yaitu RGB. Nilai matriks pada citra tersebut menjadi masukan untuk proses selanjutnya.

4.1.1 Filter Median

Dataset citra yang terbaca oleh program dilakukan proses penyaringan atau *filtering*. Sebelum dilakukan pemfilteran ukuran citra diubah terlebih dahulu menjadi ukuran yang lebih kecil yaitu 200 x 200. Gambar 4.2 merupakan perbandingan antara citra original dengan citra yang sudah di *resize*. Setelah citra di *resize* selanjutnya citra akan di filter menggunakan filter median. Filter median ini berfungsi untuk mereduksi *noise salt and pepper*, gangguan ini berbentuk bintik-

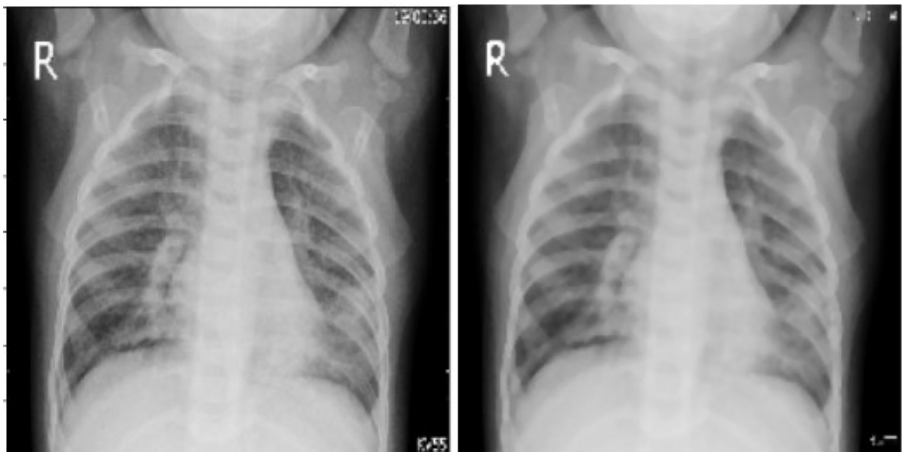
Gambar 4. 1 Citra masukan

bintik putih atau hitam pada citra. *Noise* merupakan suatu gangguan yang disebabkan oleh penyimpanan data digital yang diterima oleh alat penerima data citra yang dapat mengganggu kualitas citra. Gangguan atau *noise* pada citra dapat berasal dari gangguan fisik (optik) pada X-ray, pergerakan dari pasien, atau dari proses pengolahan yang tidak sesuai. Gambar 4.3 berikut ini merupakan perbandingan antara citra yang sudah di *resize* dengan citra yang sudah dilakukan proses *filtering* dengan ukuran kernel sebesar 3x3.



(a) (b)

Gambar 4. 2 (a) Citra sebelum di*resize*; (b) Citra setelah di*resize*



(a) (b)

Gambar 4. 3 (a) Citra sebelum difilter; (b) Hasil dari proses *median filter*

(a) (b)

Gambar 4. 4 (a) Citra sebelum CLAHE; (b) Hasil dari proses CLAHE

4.1.2 *Contrast Limited Adaptive Histogram Equalization*

Kemudian digunakan metode *Contrast Limited Adaptive Histogram Equalization* (CLAHE) untuk mengganti nilai intensitas setiap piksel citra masukan dengan memberikan batas nilai untuk penambahan kontras pada citra. Dengan menginisiasi nilai ambang batas atau yang dinamakan dengan *clip limit*, kontras pada citra akan dilakukan pemerataan serta metode ini juga dapat mereduksi *noise* pada citra. Setelah citra dilakukan proses *resize* dan *filtering* menggunakan median filter, citra akan dilakukan ekualisasi histogram menggunakan CLAHE. Hasil dari tahapan ini ditunjukkan dalam Gambar 4.4 dengan nilai *clip limit* sebesar 0,01. Metode ini memberikan hasil yang cukup baik untuk menampilkan *hidden feature* atau fitur-fitur yang tidak terlihat pada citra.

4.2 CNN dalam Pendeteksian Citra

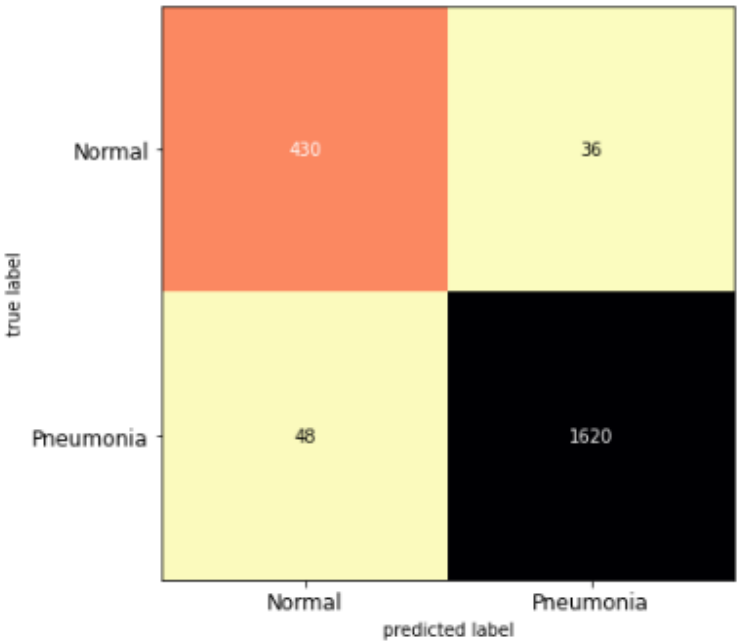
Dataset yang telah melalui tahapan *preprocessing*, hasilnya disimpan ke dalam folder baru. Kemudian data tersebut digunakan untuk tahapan pembelajaran menggunakan *Convolutional Neural Network*. Dalam algoritma CNN dibutuhkan proses pelatihan (*training*) dan pengujian (*testing*). Proses pelatihan ini bertujuan untuk melatih algoritma CNN dalam mengenali *dataset* yang digunakan dan membentuk sebuah model berdasarkan pelatihan tersebut. Pada proses pelatihan ini citra akan di ekstraksi fitur-fiturnya untuk mendapatkan informasi dari masing-masing citra tersebut. Proses ekstraksi fitur ini terjadi pada bagian *feature learning* pada algoritma CNN tepatnya pada lapisan konvolusi. Setelah ekstraksi fitur dilakukan, citra akan melalui tahap klasifikasi pada lapisan *fully connected*. Model dan bobot selama pelatihan disimpan dalam format .h5.

Pengujian dilakukan dengan memanggil model yang sudah dibentuk dan disimpan sebelumnya pada saat pelatihan atau *training* data. Citra masukan yang

digunakan dalam pengujian sama dengan citra masukan pada saat proses pelatihan. Citra masukan untuk pengujian sistem ini berasal dari berkas *val* yang berjumlah 2.134 citra digital dengan rincian 466 citra normal dan 1.668 citra pneumonia. Ukuran citra yang digunakan dalam proses pengujian ini berukuran 200 x 200 x 3 sama dengan ukuran saat citra tersebut di *training*. Citra yang digunakan untuk pengujian dilakukan *preprocessing* terlebih dahulu.

Dalam pengujian sistem dalam mengklasifikasikan citra, hasil prediksi yang didapatkan dimuat dalam sebuah matriks yang disebut dengan *confusion matrix*. Gambar 4.5 merupakan hasil *confusion matrix* dari algoritma CNN yang diujikan. Berdasarkan Gambar 4.3 dapat dilihat bahwa sistem yang digunakan menghasilkan empat kemungkinan yaitu :

- *True Negative* (TN) dalam hal ini adalah kasus dimana citra diprediksi normal dan sebenarnya memang termasuk dalam kategori normal ada 430 citra.
- *True Positive* (TP) adalah kasus dimana citra diprediksi merupakan citra dengan pneumonia dan sebenarnya memang termasuk dalam kelas citra pneumonia ada 1620 citra.
- *False Negative* (FN) dalam hal ini adalah kasus dimana citra diprediksi sebagai citra normal namun sebenarnya citra tersebut masuk dalam kategori pneumonia ada 48 citra.
- *False Positive* (FP) dalam hal ini kasus dimana citra diprediksi dalam kategori pneumonia namun sebenarnya citra tersebut merupakan citra normal ada 36 citra.



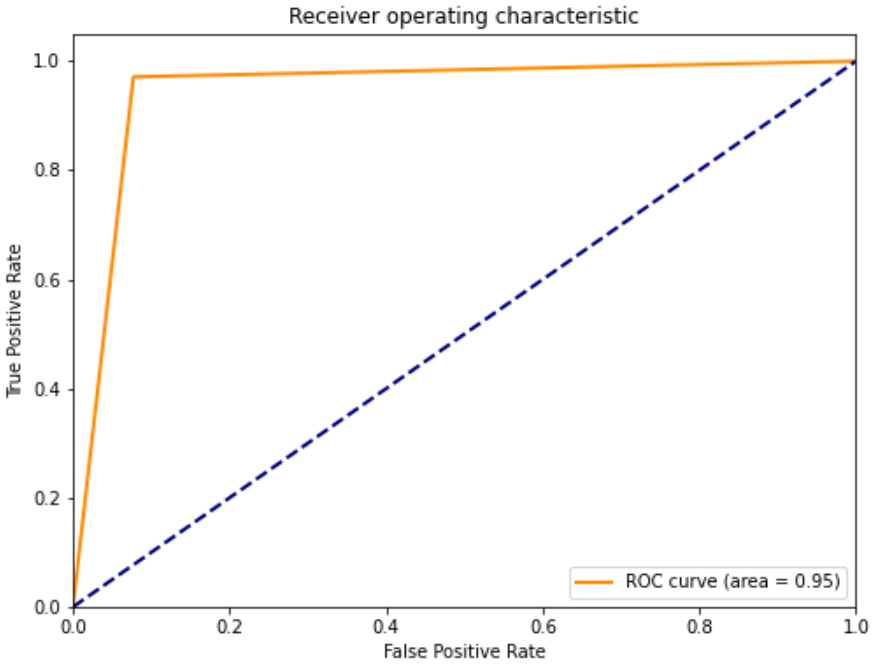
Gambar 4. 5 Hasil *Confusion Matrix*

Tabel 4. 1 Hasil Klasifikasi Menggunakan CNN

Train Accuracy	*Validation Accuracy*	*Presicion*	*Recall*	*Specificity*
96,18 %	96,06 %	97,83 %	97,12 %	92,27 %

Berdasarkan *confusion matrix* tersebut dapat dihitung nilai *accuracy*, *precission*, *recall* atau *sensitivity*, dan *specificity* dari model. Tabel 4.1 menyajikan hasil perhitungan akurasi, presisi, *recall* serta spesifisitas dari model yang dibuat.

Kinerja algoritma klasifikasi juga dapat dilihat berdasarkan kurva ROC. Kurva ROC dari model diilustrasikan pada Gambar 4.6. Kurva ROC merupakan hubungan antara uji sensitivitas atau *recall* dengan spesifisitas [18]. Kinerja algoritma CNN dikatakan baik jika kurva mendekati titik (0,1), sedangkan kinerja algoritma yang dikatakan buruk terjadi jika kurva yang dihasilkan mendekati garis *baseline* atau garis yang melintang dari titik (0,0) hingga (1,1). Berdasarkan Gambar 4.4 terlihat bahwa kurva yang dihasilkan mendekati titik (0,1) dengan nilai *Area Under Curve* (AUC) sebesar 0.95. Sehingga dapat dikatakan bahwa performa algoritma klasifikasi ini cukup baik.



Gambar 4. 6 Hasil Kurva ROC

4.3 Prediksi Citra Menggunakan GUI

Setelah keseluruhan proses mulai dari pembuatan data latih dan validasi sampai dengan didapatkannya hasil *training* dan model sudah disimpan dalam format .h5. Selanjutnya adalah penggunaan *Graphical User Interface* (GUI) untuk memudahkan pendeteksian citra. Model yang sudah disimpan dalam format .h5 akan dimuat atau di *load* dalam tkinter agar tidak perlu melakukan proses *training* kembali pada GUI. Namun untuk tahapan pra pemrosesan dapat dilakukan dengan menggunakan GUI. Gambar 4.7 merupakan antarmuka dari GUI yang sudah dibuat untuk memprediksi citra pneumonia dan normal. Penggunaan GUI ini diawali dengan memilih citra yang akan di deteksi kemudian dilakukan *preprocessing* agar performa pengklasifikasiannya lebih baik. Selanjutnya citra hasil *preprocessing* akan disimpan kemudian dilakukan proses klasifikasi. Gambar 4.8 merupakan contoh hasil penggunaan GUI untuk memprediksi citra.

Gambar 4. 7 Tampilan antarmuka GUI

Gambar 4. 8 Tampilan antarmuka setelah GUI dijalankan

4.4 Pengujian dengan *Balancing Data*

Pada subbab 3.2 telah dijelaskan bahwa dalam proses *training* digunakan data dalam kondisi yang tidak seimbang atau *imbalanced dataset*. Sehingga dilakukan pengujian dengan melakukan *balancing data* untuk melihat performa algoritma CNN dengan data yang seimbang. Pengujian dengan *balancing data* dilakukan dua kali, pertama *balancing data* dengan melakukan metode *undersampling* untuk menyamaratakan jumlah data dalam folder *train* untuk kategori pneumonia. Jumlahnya disamakan dengan jumlah data untuk kategori normal. Kemudian pengujian yang kedua adalah melakukan proses augmentasi untuk menambah variasi data, data yang dilakukan proses augmentasi adalah data dengan kategori normal dan jumlahnya disamakan dengan jumlah data pada kategori pneumonia.

4.4.1 *Balancing Data* dengan Metode *Undersampling*

Undersampling merupakan sebuah metode yang digunakan untuk menyeimbangkan distribusi kelas untuk *dataset* yang tidak seimbang. *Dataset* yang tidak seimbang memiliki satu atau lebih kelas yang jumlah datanya lebih sedikit atau disebut kelas minoritas dan memiliki satu atau lebih kelas dengan jumlah data yang lebih banyak yang disebut dengan kelas mayoritas. Dalam *dataset* yang digunakan dalam tugas akhir ini hanya memiliki dua kelas dimana kelas atau

```
total picture numbers in train_nor:  1117
total picture numbers in train_pneu:  2605
total picture numbers in val_nor:  466
total picture numbers in val_pneu:  1668
```

(a)

```
total picture numbers in train_nor:  1117
total picture numbers in train_pneu:  1117
total picture numbers in val_nor:  466
total picture numbers in val_pneu:  1668
```

(b)

Gambar 4. 9 Distribusi dataset (a) Sebelum dilakukan *undersampling*; (b) Setelah dilakukan *undersampling*

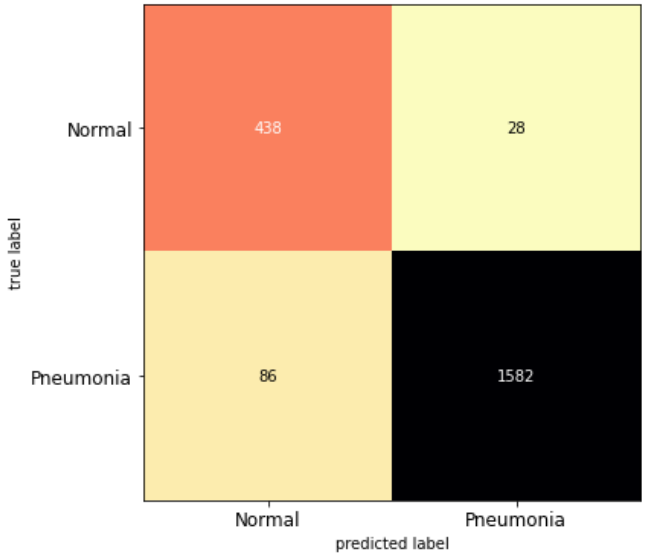
Tabel 4. 2 Performa CNN untuk pengujian *balancing data* dengan metode *undersampling*

Train Accuracy	***Validation Accuracy***	***Presicion***	***Recall***	***Specificity***
95,57 %	94,66 %	98,26 %	94,84 %	93,99 %

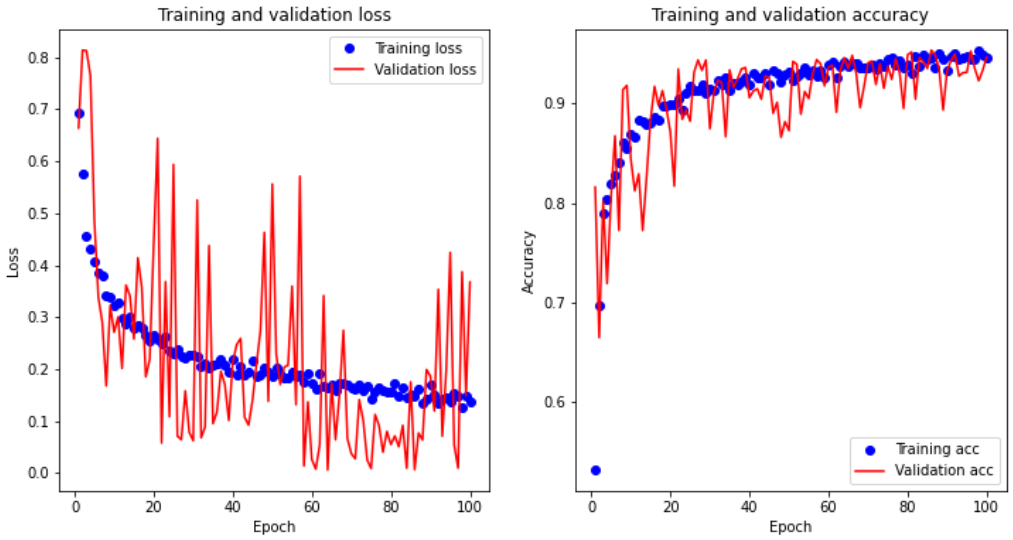
kategori normal merupakan kelas minoritas dan kelas pneumonia merupakan kelas mayoritas karena jumlah datanya yang lebih banyak. Teknik *undersampling* ini menghapus beberapa data sesuai kebutuhan dari *dataset* pelatihan yang menjadi kelas mayoritas untuk menyeimbangkan distribusi kelas. Proses menghapus data ini dilakukan secara acak. Proses *undersampling* dilakukan pada *dataset* pelatihan dan Gambar 4.9 merupakan perbandingan distribusi *dataset* sebelum dilakukan *undersampling* dengan *dataset* setelah dilakukan *undersampling*.

Berdasarkan Gambar 4.9 dapat dilihat bahwa setelah dilakukan proses *undersampling* jumlah data *train* yang sebelumnya sebanyak 3722 citra berubah menjadi 2234 citra dengan perbandingan kelas normal dan pneumonia dalam data *train* menjadi 1:1. Tahapan *preprocessing* hingga proses *training* sama seperti pengujian dengan *dataset* yang tidak seimbang, hanya berbeda dalam pembagian data latih. Gambar 4.11 merupakan grafik yang mengilustrasikan nilai akurasi dan kerugian selama proses *training*. Hasil dari pengujian dengan metode *undersampling* ditunjukkan dalam Tabel 4.2. Berdasarkan Tabel 4.2 performa model dengan *dataset* setelah dilakukan proses *undersampling* menghasilkan akurasi *train* sebesar 95,57 % dan akurasi validasi sebesar 94,66 %. Gambar 4.10 merupakan *confusion matrix* yang dihasilkan, dari Gambar 4.10 dapat dilihat bahwa performa CNN dalam mengkasifikasikan citra pneumonia dengan benar sebanyak 1582 citra dan 86 citra pneumonia yang diprediksi salah. Jumlah citra normal yang

diprediksi benar sebanyak 438 citra dan citra normal yang diprediksi ke dalam kategori pneumonia sebanyak 28 citra. Dari *confusion matrix* tersebut dapat dihitung nilai presisi, *recall*, dan spesifisitas yang hasilnya telah dilampirkan pada Tabel 4.2. Kemudian juga dilakukan analisa performa algoritma CNN berdasarkan kurva ROC. Kurva ROC digambarkan pada Gambar 4.12. Berdasarkan kurva ROC tersebut nilai luasan dibawah kurva atau *Area Under Curve* (AUC) sebesar 0.94.

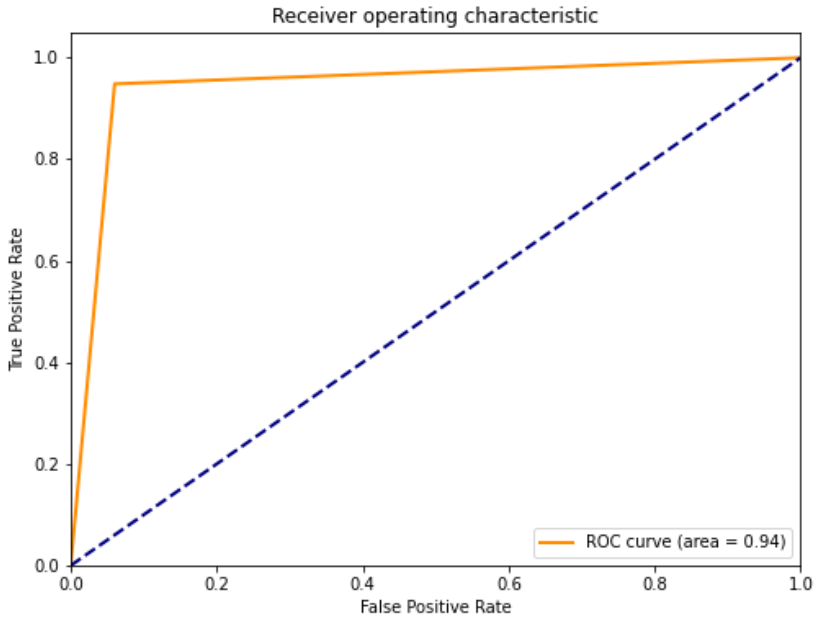


Gambar 4. 10 Hasil *confusion matrix* berdasarkan dataset yang telah di *undersampling*



(a) (b)

Gambar 4. 11 (a) Grafik *train and validation loss* setelah *undersampling;* (b) Grafik *train and validation accuracy* setelah *undersampling*



Gambar 4. 12 Kurva ROC berdasarkan *dataset* yang telah di *undersampling*

4.4.2 *Balancing Data* dengan Metode Augmentasi Citra

Augmentasi citra adalah proses mengubah atau memodifkasi gambar untuk menambah variasi gambar yang akan digunakan untuk melatih model dalam CNN. Metode augmentasi ini dilakukan pada data latih yang jumlah datanya lebih sedikit yaitu citra dalam kelas normal yang berjumlah 1117 citra. Metode augmentasi yang dilakukan meliputi proses *rotate*, *width shift*, *height shift*, *shearing*, *zooming*, dan *horizontal flip*. Tabel 4.3 merupakan rangkuman mengenai pengaturan untuk metode augmentasi yang dilakukan.

Hasil dari proses augmentasi ini disimpan ke dalam folder baru agar tidak tercampur dengan data aslinya. Proses augmentasi dilakukan sebanyak 5 kali untuk mendapatkan variasi citra yang lebih banyak. Sehingga jumlah data normal setelah dilakukan proses augmentasi yang awalnya sebanyak 1117 citra menjadi 5585 citra. Karena setelah dilakukan augmentasi jumlah citra normal pada data latih menjadi lebih banyak dibandingkan dengan citra pneumonia maka hanya diambil sebanyak 1488 citra yang telah di augmentasi. Citra normal asli yang berjumlah 1117 dicampur dengan 1488 citra yang telah di augmentasi agar jumlah citra normal sama dengan jumlah data. Proses pemilihan 1488 citra tersebut dilakukan secara acak menggunakan program. Sehingga distribusi data latih setelah proses augmentasi seperti yang digambarkan pada Gambar 4.13. Jumlah data latih setelah dilakukan proses augmentasi menjadi 5210 dengan perbandingan kelas normal dan pneumonia dalam data latih menjasi 1:1.

Tabel 4. 3 Pengaturan metode augmentasi untuk citra normal pada data latih

Metode	Pengaturan
Rotation	40
Width shift	0,2
Height shift	0,2
Shear range	0,2
Zoom range	0,2
Horizontal flip	*True*

```
total picture numbers in train_nor:  1117
total picture numbers in train_pneu:  2605
total picture numbers in val_nor:  466
total picture numbers in val_pneu:  1668
```

(a)

```
total picture numbers in train_nor:  2605
total picture numbers in train_pneu:  2605
total picture numbers in val_nor:  466
total picture numbers in val_pneu:  1668
```

(b)

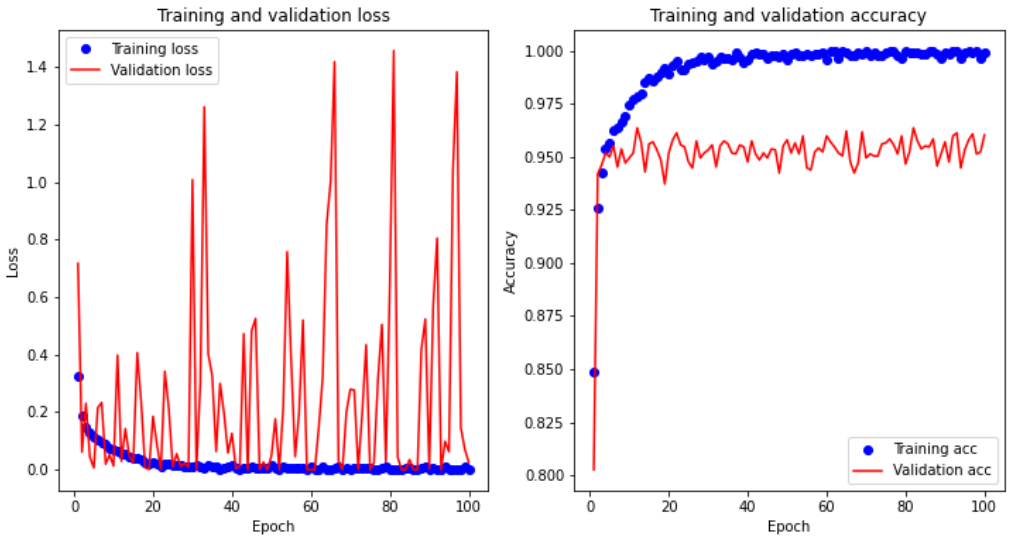
Gambar 4. 13 Distribusi *dataset* (a) Sebelum dilakukan augmentasi; (b) Setelah dilakukan augmenasi

Dataset yang digambarkan pada Gambar 4.13 dilakukan tahapan pra pemrosesan terlebih dahulu sebelum memasuki proses *training*. Tahap *preprocessing* dan *training* ini sama seperti tahap *preprocessing* dan *training* saat distribusi data latih yang tidak seimbang. Hasil yang didapatkan dari pengujian dengan metode augmentasi ini dirangkum pada Tabel 4.4. Berdasarkan Tabel 4.4 dapat dilihat bahwa nilai akurasi *train* mencapai nilai maksimum yaitu 100 % dan akurasi validasi mencapai 95,88 %.

Tabel 4. 4 Performa CNN untuk pengujian *balancing data* dengan metode augmentasi

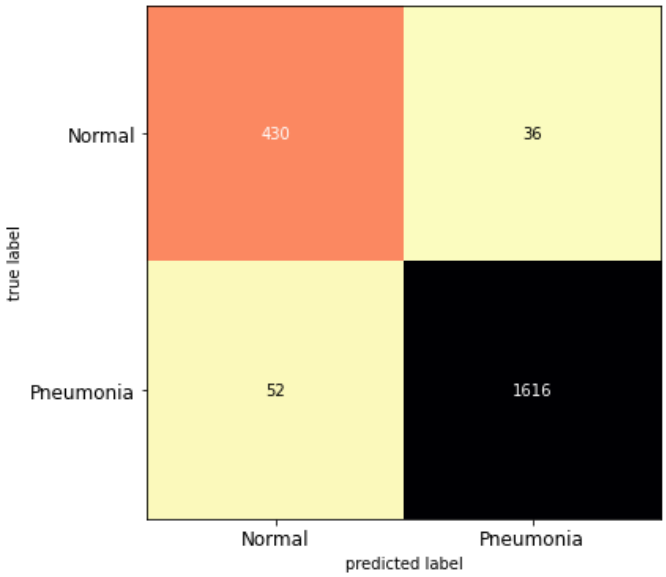
Train Accuracy	***Validation Accuracy***	***Presicion***	***Recall***	***Specificity***
100 %	95,88 %	97,82 %	96,88 %	92,27 %

Gambar 4.14 mengilustrasikan nilai akurasi dan kerugian selama proses *training*. Hasil prediksi dari pengujian dengan *balancing data* menggunakan metode augmentasi digambar pada *confusion matrix* yang digambarkan pada Gambar 4.15. Berdasarkan Gambar 4.15 performa CNN setelah data latih dilakukan proses augmentasi untuk menyeimbangkan dataset berhasil memprediksi citra pneumonia ke dalam kategori pneumonia (*True Positive*) sebanyak 1616 citra dan memprediksi citra pneumonia ke dalam kategori normal (*False Positive*) sebanyak 52 citra. Kemudian citra normal yang diprediksi ke dalam kategori normal (*True Negative*) sebanyak 430 citra dan citra normal yang diprediksi sebagai citra pneumonia (*False Negative*) sebanyak 36 citra. Berdasarkan TP, FP, TN, dan FN dari *confusion matrix* pada Gambar 4.15 dapat dihitung nilai presisi, *recall*, dan spesifisitas yang telah dilampirkan pada Tabel 4.4. Untuk menguji performa CNN juga dilakukan dengan menganalisa kurva ROC. Gambar 4.16 merupakan kurva ROC yang didapatkan dengan *Area Under Curve* (AUC) sebsar 0,95. AUC yang didapatkan sama dengan AUC saat perbandingan kelas pneumonia dan normal pada data latih sebesar 7:3.

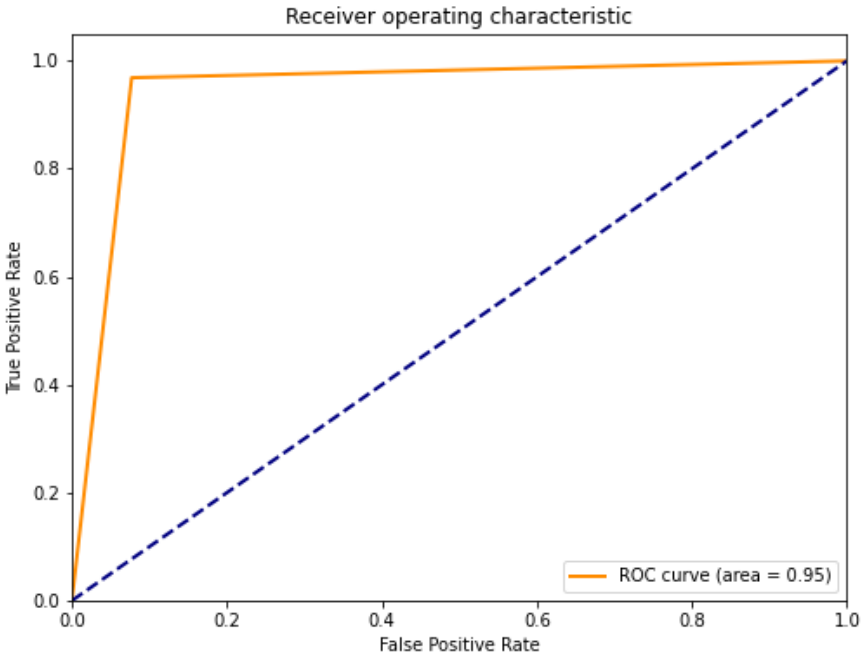


(a) (b)

Gambar 4. 14 (a) Grafik *train and validation loss* setelah augmentasi; (b) Grafik *train and validation accuracy* setelah augmentasi



Gambar 4. 15 Hasil *confusion matrix* berdasarkan *dataset* yang telah di augmentasi



Gambar 4. 16 Kurva ROC berdasarkan *dataset* yang telah di augmentasi

4.5 Perbandingan Performa CNN Berdasarkan Kondisi *Dataset*

Seperti yang sudah dijelaskan pada subsubbab 3.2.2, bahwa kondisi *dataset* yang digunakan dalam proses *training* kondisinya tidak seimbang antara kelas normal dengan kelas pneumonia. Kondisi ini juga dikenal dengan istilah *imbalanced dataset*. Pada kondisi tersebut jumlah citra pneumonia lebih besar dibandingkan dengan jumlah citra normal dengan perbandingan jumlah citra pneumonia dengan citra normal adalah 7:3. Dapat dikatakan bahwa kondisinya sangat tidak seimbang. Sehingga dilakukan pengujian dengan melakukan proses *balancing data* menggunakan dua metode yaitu *undersampling* dan augmentasi seperti yang sudah dijelaskan pada subsubbab 4.4.1 dan 4.4.2.

Dari pengujian tersebut dapat dibandingkan performa algoritma CNN dalam mengklasifikasikan citra dengan tiga kondisi data *train* yang berbeda. Tabel 4.5 merupakan perbandingan performa CNN antara *imbalanced dataset* dengan *balanced dataset,* dengan tiga kondisi data *train* yang berbeda. Berdasarkan Tabel 4.5 jika dilihat pada akurasi validasi atau akurasi dalam pengujian dan berdasarkan nilai *recall*, saat kondisi data tidak seimbang akurasi dan nilai *recall* lebih tinggi dalam mengklasifikasikan citra dan jika dilihat pada *confusion matrix* saat data yang digunakan tidak seimbang, dalam memprediksi citra pneumonia dengan benar lebih banyak dibandingkan saat kondisi datanya seimbang tiap kelasnya. Namun jika dilihat berdasarkan nilai presisi dan spesifisitas ketika data dalam kondisi seimbang terutama ketika menggunakan metode *undersampling* untuk menyamaratakan jumlah data tiap kelas nilai presisi dan spesifisitasnya lebih besar dibandingkan saat data dalam kondisi tidak seimbang. Ketika nilai presisi lebih rendah menunjukkan bahwa nilai *False Positive* (FP) yang tinggi. *False Positive* adalah kondisi dimana citra pada kategori normal diprediksi sebagai citra pneumonia. Nilai spesifisitas berisi informasi mengenai performa algoritma klasifikasi dalam memprediksi kelas negatif. Karena nilai FP yang besar ketika data dalam kondisi tidak seimbang maka nilai spesifisitas dan presisinya lebih rendah. Hal ini dapat terjadi karena saat data dalam kondisi *imbalanced* kemungkinan terjadi *overfitting* pada salah satu kelas.

Tabel 4. 5 Hasil perbandingan performa CNN berdasarkan kondisi *dataset*

Kondisi *Dataset*	*Train Accuracy*	*Validation Accuracy*	*Precision*	*Recall*	*Specificity*
Imbalanced dataset	96,18 %	96,06 %	97,83 %	97,12 %	92,27 %
Balanced dataset (*Undersampling*)	95,57 %	94,66 %	98,26 %	94,84 %	93,99 %
Balanced dataset (Augmentasi)	100 %	95,88 %	97,82 %	96,88 %	92,27 %

BAB V HASIL DAN PEMBAHASAN

Pada tugas akhir ini dilakukan proses klasifikasi dua kelas citra X-ray yaitu normal dan pneumonia. Proses dalam pembuatan model ini diawali dengan tahapan *preprocessing* yang kemudian dilanjutkan dengan proses pelatihan atau *training* pada *dataset*. Proses pelatihan ini bertujuan untuk pembentukan model dan pengenalan *dataset* pada model yang nantinya digunakan dalam pengujian atau *testing*. Parameter yang digunakan untuk mengukur performa model yang diusulkan adalah nilai akurasi. Nilai akurasi model didapatkan dengan melakukan proses pengujian pada data validasi.

5.1 *Image Pre-Processing*

Pada tahapan pengujian, citra uji dibuka dan dibaca menggunakan *library* skimage pada Python sebagai sebuah matriks yang berukuran [lebar, panjang, 3]. Angka 3 menunjukkan bahwa citra memiliki 3 *channel* artinya citra tersebut dibaca sebagai data RGB. Citra uji tersebut kemudian masuk pada tahap *preprocessing*. Tahapan *preprocessing* ini meliputi pemfilteran dan ekualisasi histogram. Hasil pengujian akurasi sistem menggunakan *dataset* yang sudah dilakukan *preprocessing* dengan citra yang belum dilakukan *preprocessing* disajikan pada Tabel 5.1.

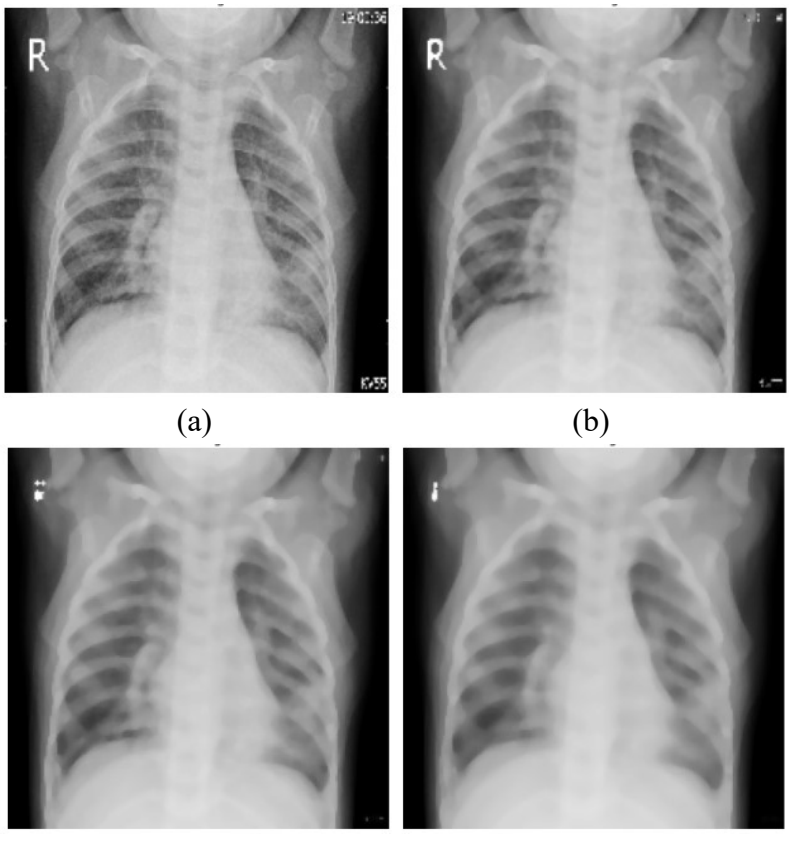
Dengan adanya tahap *preprocessing* performa model yang diusulkan menghasilkan akurasi yang lebih tinggi dibandingkan tidak menggunakan *preprocessing*. Hal ini bisa terjadi karena citra dari *dataset* tersebut memiliki *noise* atau kontras yang kurang baik. Berdasarkan Tabel 5.1 nilai akurasi pada *train* dan *validation*, presisi, dan spesifisitas pada *dataset* yang telah di *preprocessing* lebih besar dibandingkan dengan nilai pada *dataset* yang tidak di *preprocessing*. Namun nilai *recall* pada model CNN dengan *dataset* yang tidak dilakukan *preprocessing* lebih tinggi.

Tabel 5. 1 Hasil perbandingan akurasi dari CNN dengan menggunakan *preprocessing* dengan tidak menggunakan *preprocessing*

Keterangan	*Train Accuracy*	*Validation Accuracy*	*Precision*	*Recall*	*Specificity*
Tanpa *preprocessing*	95,46 %	95,78 %	96,58 %	98,08 %	87,55 %
Dengan *preprocessing*	96,18 %	96,06 %	97,83 %	97,12 %	92,27 %

5.1.1 *Median Filter*

Penggunaan median filter pada tahap *preprocessing* memberikan efek *blurring* pada citra masukan seperti yang ditunjukkan pada Gambar 4.1. Dari Gambar 4.1 tersebut terlihat bahwa ada perbedaan antara citra asli dengan citra hasil median filter. Citra hasil pemfilteran ini memberikan hasil berupa reduksi *noise* pada citra. Metode penyaringan menggunakan median filter ini dilakukan dalam beberapa percobaan dengan menentukan ukuran kernel untuk mendapatkan hasil yang lebih baik. Gambar 5.1 merupakan perbandingan hasil pemfilteran berdasarkan ukuran kernel. Percobaan dilakukan dengan ukuran kernel 3x3, 5x5 dan 7x7. Berdasarkan Gambar 5.1, hasil yang didapatkan saat ukuran kernel 5x5 efek *blurring* yang dihasilkan membuat citra menjadi lebih halus namun detail pada citranya sedikit berkurang, sedangkan ukuran kernel 3x3 efek pengaburan yang dihasilkan membuat citra menjadi lebih halus namun tidak kehilangan detail pada citranya. Saat ukuran kernel 7x7 efek *blurring* atau pengaburan pada citra terlalu berlebihan dan detail pada citra semakin berkurang. Setelah dilakukan pengujian pada algoritma CNN dengan melihat pengaruh *kernel size* pada filter median hasil yang lebih baik didapatkan saat ukuran kernel sebesar 3x3. Hasil pengujian



(a) (b)

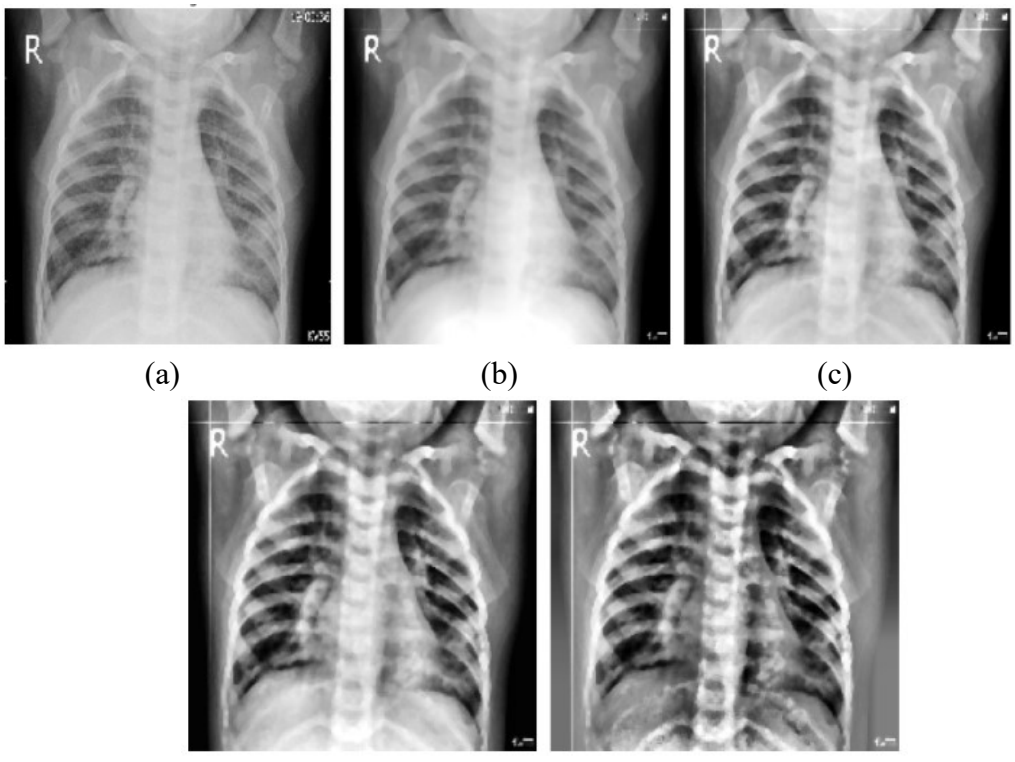
(c) (d)

Gambar 5. 1 Hasil median filter dengan ukuran *kernel* (a) Citra setelah di*resize*; (b) 3x3; (c) 5x5; (d) 7x7

berdasarkan ukuran kernel ini terlampir pada lampiran. Sehingga ukuran kernel 3x3 dipilih karena hasilnya lebih baik dibandingkan dua ukuran kernel yang lain. Oleh karena itu metode ini digunakan dalam sistem untuk meningkatkan performa algoritma CNN dalam mengklasifikasikan data.

5.1.2 *Contrast Limited Adaptive Histogram Equalization*

Contrast Limited Adaptive Histogram Equalization (CLAHE) merupakan proses untuk meningkatkan kontras pada citra, sehingga *noise* pada citra dapat berkurang. Metode CLAHE menghasilkan keluaran berupa adanya peningkatan kontras pada citra, sehingga bagian-bagian yang tersembunyi dapat terlihat. Pada Gambar 4.2 dapat dilihat perbedaan antara citra sebelum dilakukan proses CLAHE dengan citra setelah dilakukan proses CLAHE. Dalam metode ini beberapa parameter ditentukan untuk mendapatkan hasil yang sesuai. Gambar 5.2 merupakan perbandingan hasil CLAHE berdasarkan ukuran *clip limit* yang digunakan, pada gambar tersebut sudah dilakukan *median filtering* terlebih dahulu untuk memberikan efek *smooting*. Berdasarkan Gambar 5.2, hasil yang didapatkan saat ukuran *clip limit* 0,5 kontras yang dihasilkan tampak berlebihan dan *noise* pada citra semakin banyak. Sedangkan pada *clip limit* 0,02 beberapa bagian yang berwarna putih pada citra terutama pada area paru-paru tampak berkurang, sehingga



(a) (b) (c)

(d) (e)

Gambar 5. 2 Hasil CLAHE dengan ukuran *clip limit* (a) Citra setelah di*resize* dan difilter; (b) 0,005; (c) 0,01; (d) 0,02; (e) 0,5

dikhawatirkan konsolidasi pada area paru-paru ikut dihilangkan. Saat *clip limit* sebesar 0,005 kontras yang dihasilkan semakin terang terutama pada area jantung dan pada beberapa citra kontrasnya terlihat berlebihan. Kemudian dilakukan pengujian terhadap algoritma CNN dengan melihat pengaruh nilai *clip limit* pada CLAHE dan hasil yang terbaik didapatkan ketika nilai *clip limit* sebesar 0,01. Hasil pengujian algoritma CNN berdasarkan nilai *clip limit* ini terlampir pada lampiran. Sehingga *clip limit* 0,01 dipilih untuk meningkatkan kualitas pada citra. Oleh karena itu proses ini digunakan dalam sistem untuk meningkatkan akurasi dalam pendeteksian citra pneumonia.

5.2 CNN untuk Deteksi Citra

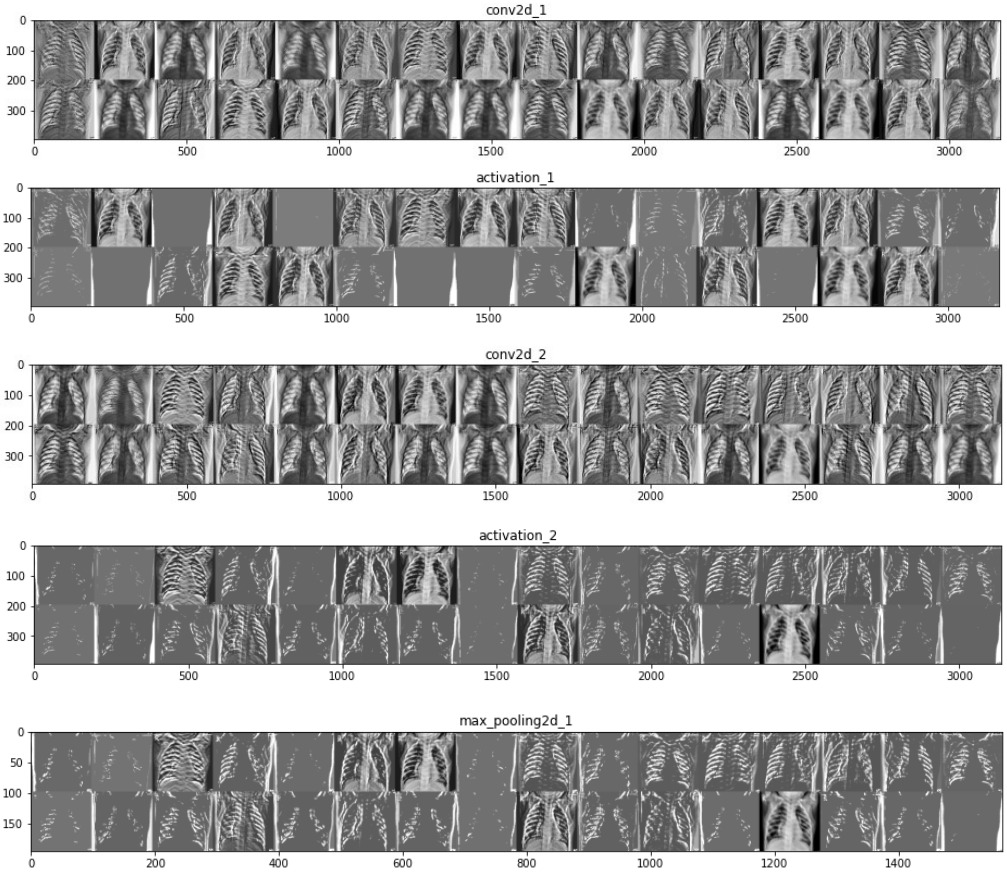
Convolutional Neural Network merupakan salah satu jenis *neural network* yang digunakan untuk data berupa *image* atau citra. *Convolutional Neural Network* dapat digunakan untuk mendeteksi dan melakukan klasifikasi pada sebuah citra digital. Oleh karena itu dalam tugas akhir ini metode CNN digunakan untuk mendeteksi citra normal dan pneumonia dari suatu modalitas pencitraan X-ray. Dalam algoritma CNN pembentukan aristektur jaringan mempengaruhi akurasi pada model. Pada Tabel 3.3 diuraikan mengenai arsitektur jaringan yang digunakan pada proses pelatihan untuk menghasilkan performa model yang optimal. Pada tugas akhir ini digunakan citra masukan yang berukuran 200 x 200 x 3, ukuran citra ini disamakan dengan citra masukan pada penelitian O. Stephen *et al* [9] untuk dapat dibandingkan performanya, dan menurut penelitian tersebut ukuran 200 x 200 menghasilkan performa yang lebih baik dengan waktu *training* yang lebih cepat.

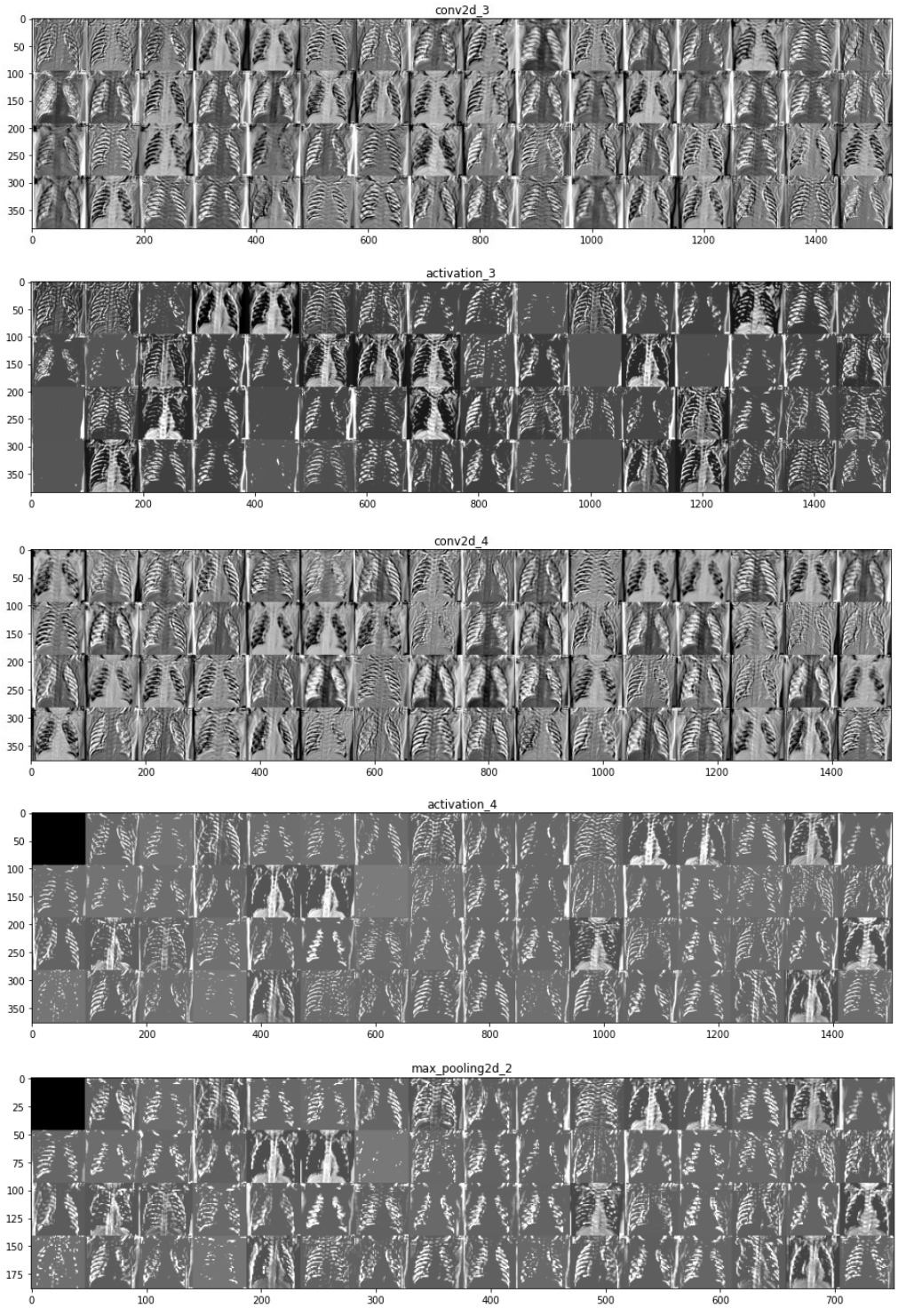
Pada arsitektur CNN yang diusulkan citra masukan akan melalui proses konvolusi pertama yang memiliki kernel berukuran 3x3 dan jumlah filter yang digunakan sebanyak 32 filter. Proses ini merupakan proses kombinasi antara dua buah matriks yang berbeda untuk menghasilkan sebuah matriks dengan nilai yang baru. Pada proses konvolusi ini digunakan sebuah fungsi aktivasi yaitu *Rectified Linear Unit* (ReLU) yang tujuannya adalah untuk menghilangkan nilai negatif yang ada pada matriks. Matriks yang dihasilkan dari konvolusi pertama akan dilakukan proses konvolusi yang kedua dengan kernel dan jumlah filter yang sama dengan proses konvolusi pertama. Hasil dari proses konvolusi ini adalah matriks sebanyak jumlah filter yang digunakan.

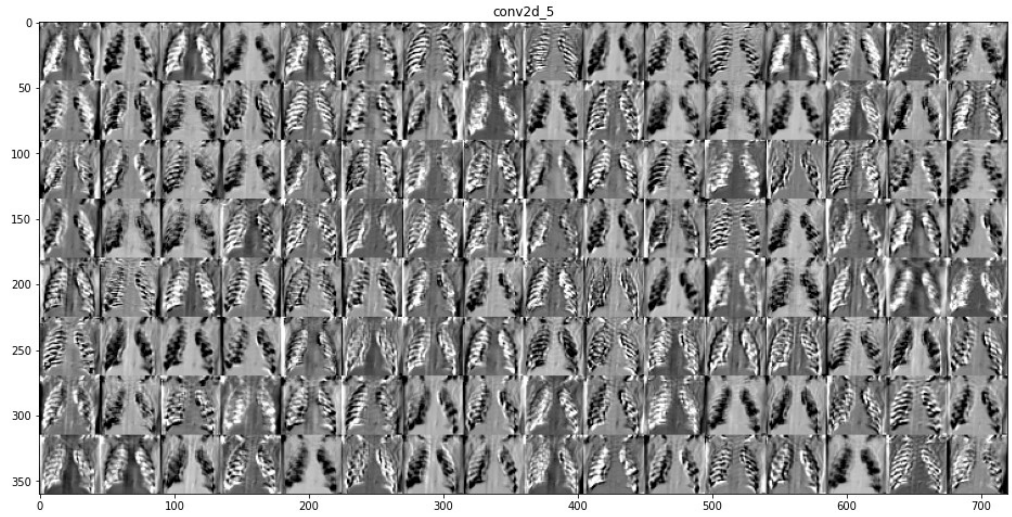
Setelah proses konvolusi pertama dan kedua, selanjutnya adalah proses *pooling*. Proses *pooling* merupakan proses reduksi ukuran matriks dengan mengunakan operasi *pooling*. Terdapat beberapa metode *pooling* yang digunakan yaitu *maxpooling*, *average pooling*, dan *L2-norm pooling*. Pada tugas akhir ini digunakan metode *maxpooling* untuk mendapatkan nilai matriks yang baru dari proses *pooling.* Ukuran *pooling* yang digunakan dalam tugas akhir ini adalah 2x2. Ukuran ini ditentukan mengikuti arsitektur yang digunakan, bila ukuran diubah menjadi lebih kecil yaitu 1x1, maka sama saja dengan tidak melakukan proses

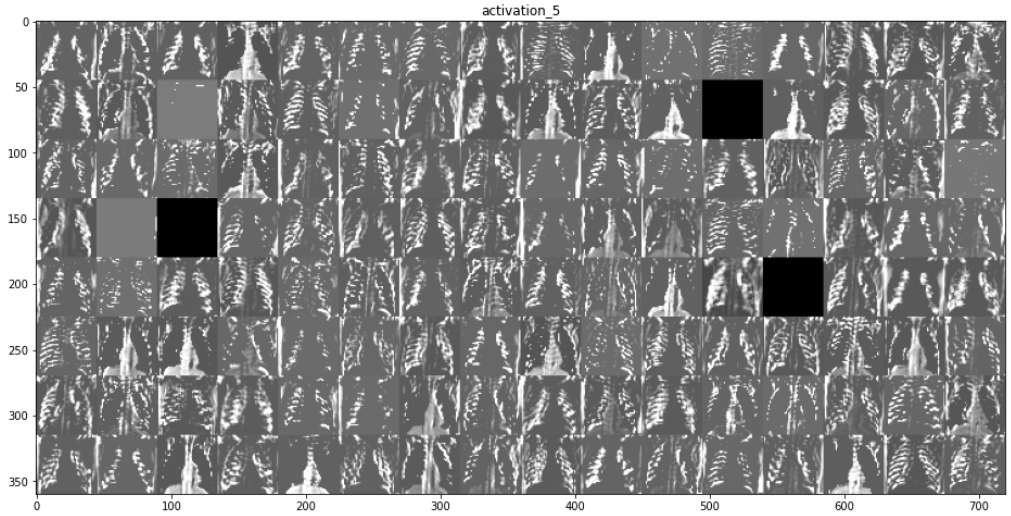
pooling. Sehingga fitur yang dihasilkan antara proses *pooling* dengan proses konvolusi itu sama. Proses konvolusi ketiga dan seterusnya prosesnya sama hanya jumlah filternya saja yang berubah. Setiap dua kali konvolusi jumlah filter ditingkatkan sebanyak dua kali dari jumlah filter pada konvolusi sebelumnya.

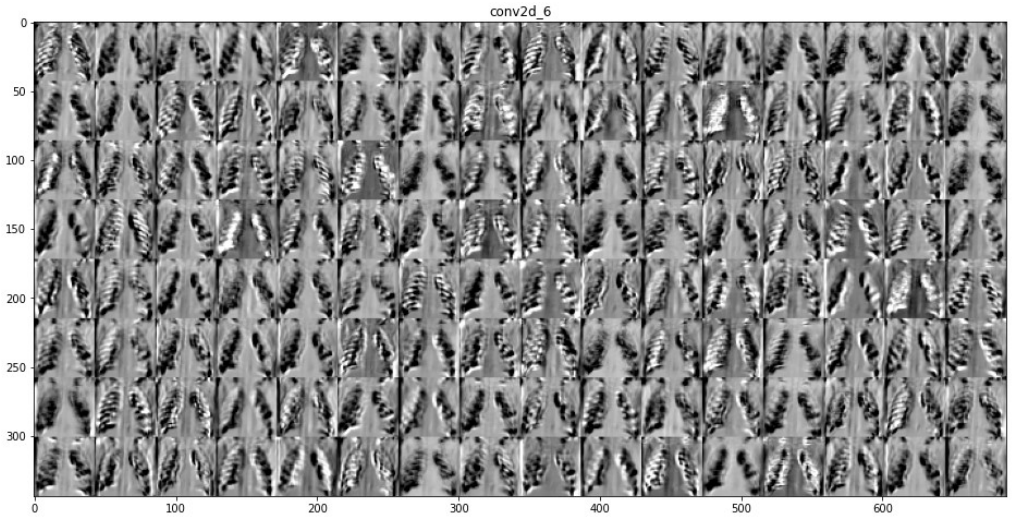
Hasil dari proses konvolusi berfungsi sebagai masukan pada *Fully Connected Layer*. Gambar 5.3 merupakan hasil dari tahapan *feature learning* mulai dari lapisan konvolusi tahap pertama hingga lapisan konvolusi ke-8. Keluaran dari proses konvolusi diubah menjadi sebuah vektor menggunakan *flatten*. Sebelum dilakukan proses klasifikasi untuk memprediksi citra, ditambahkan *dropout* untuk menyeleksi neuron secara acak dan tidak akan dipakai selama proses *training*. Tujuannya adalah untuk mengurangi *overfitting* pada saat pelatihan data. *Fully connected* atau *dense layer* yang digunakan dalam tugas akhir ini dibangun sebanyak dua lapisan. Lapisan *dense* pertama memiliki 512 unit dengan fungsi aktivasi ReLU, dan lapisan *dense* kedua memiliki 1 unit dengan fungsi aktivasi Sigmoid. Fungsi aktivasi Sigmoid merupakan fungsi nonlinear, dan keluaran dari fungsi ini memiliki range nilai antara 0 sampai 1. Berdasarkan arsitektur yang digunakan untu melatih *dataset* dalam tugas akhir ini, Tabel 5.2 merupakan keluaran model pelatihan yang digunakan.

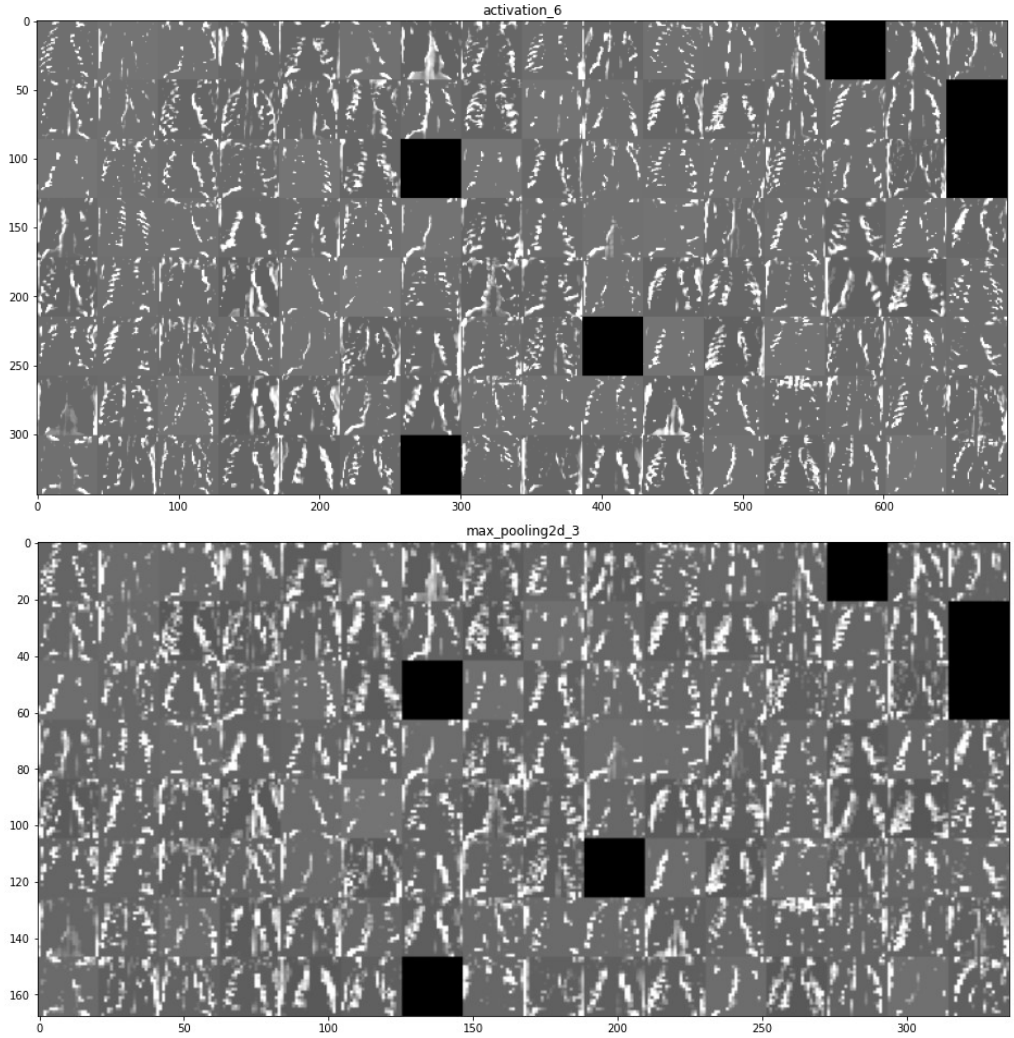


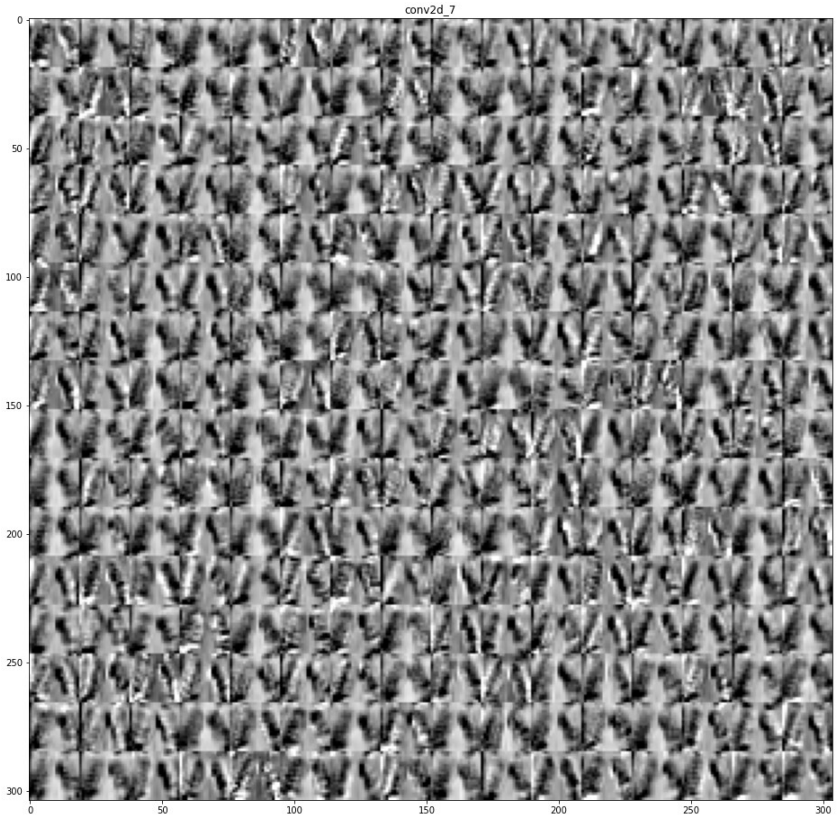

conv2d_3
activation_3
conv2d_4
activation_4
max_pooling2d_2

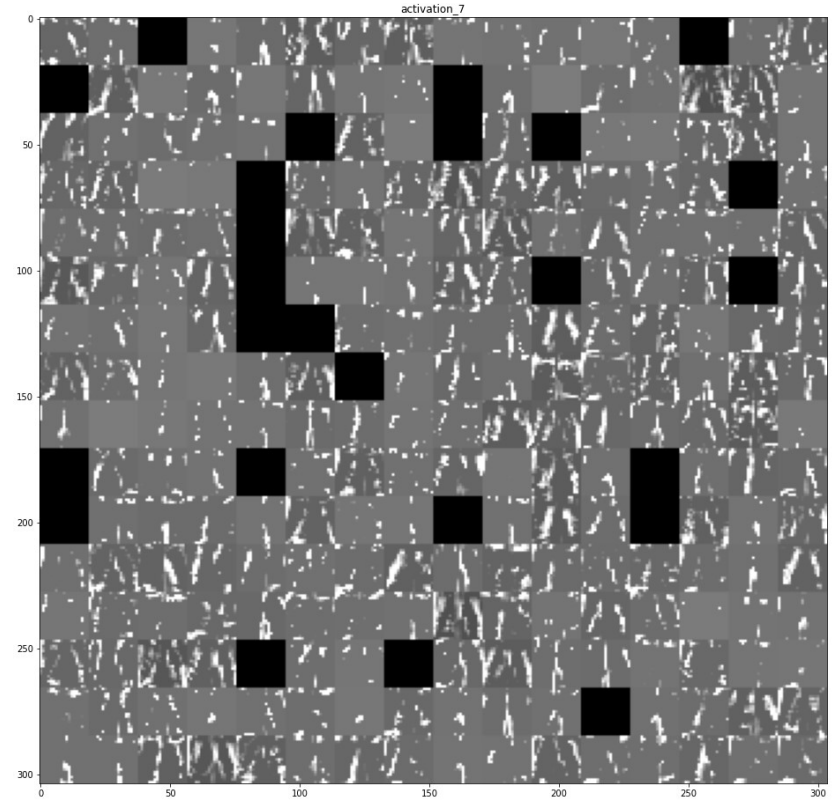

conv2d_5
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600
700

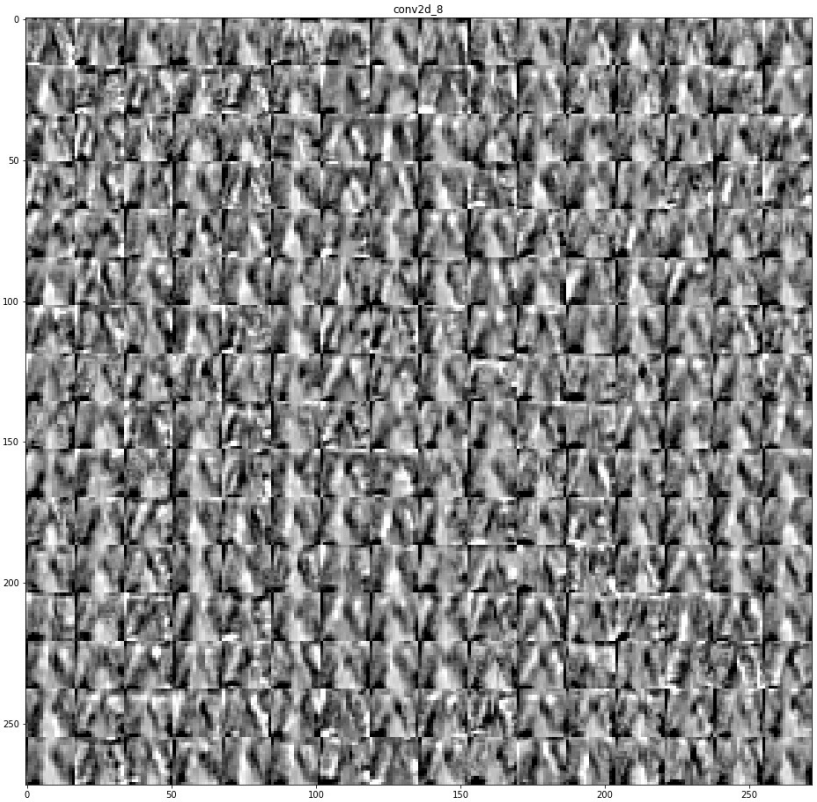

activation_5
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600
700

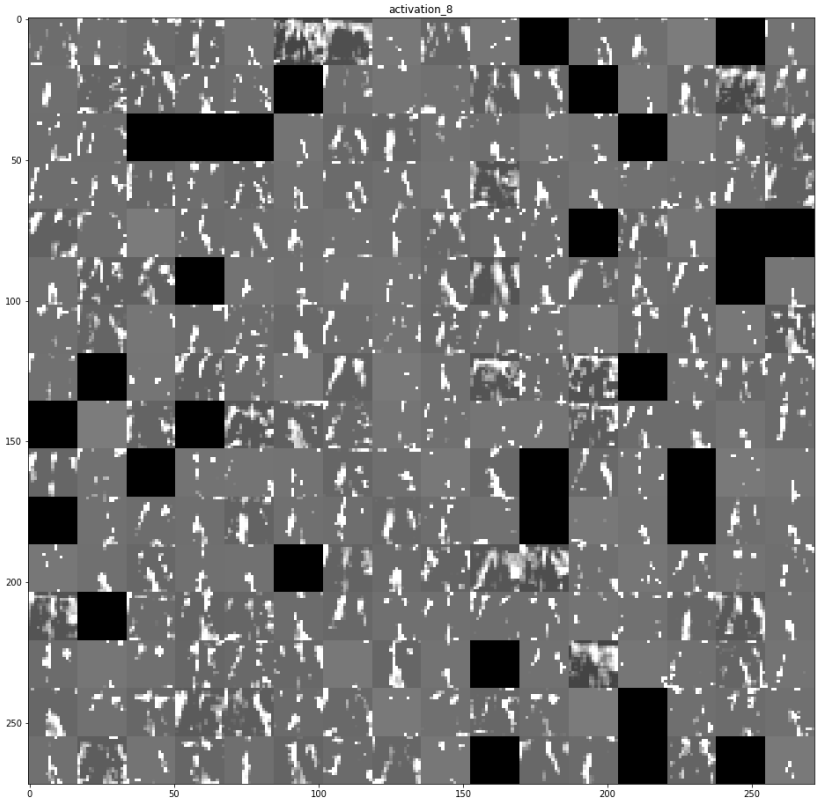

conv2d_6
0
50
100
150
200
250
300
0
100
200
300
400
500
600

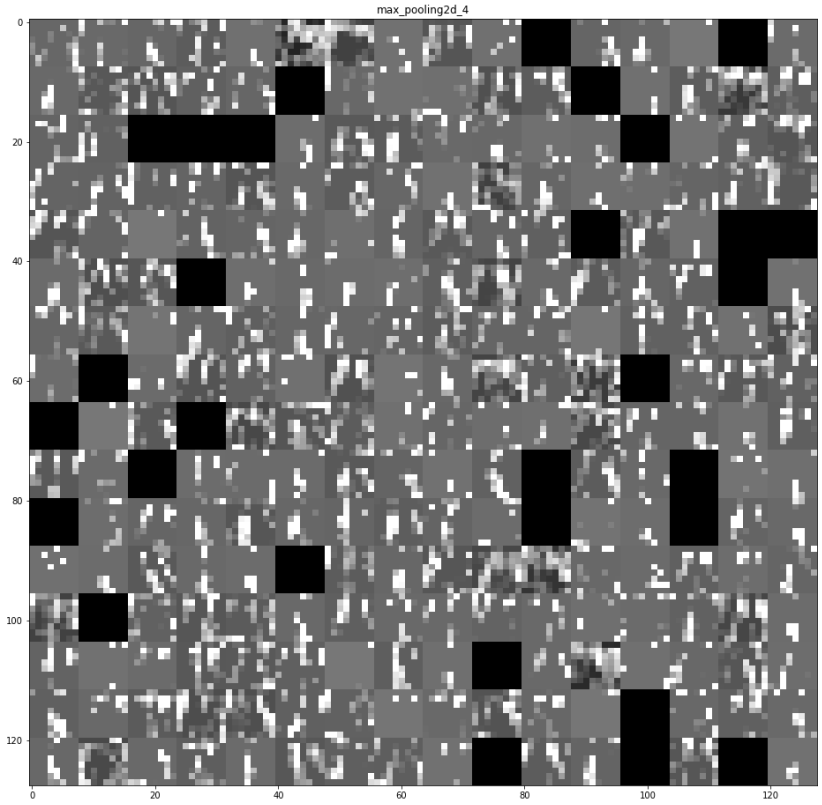

activation_6
max_pooling2d_3


conv2d_7
0
50
100
150
200
250
300
0
50
100
150
200
250
300


activation_7
0
50
100
150
200
250
300
0
50
100
150
200
250
300


conv2d_8
0
50
100
150
200
250
0
50
100
150
200
250


activation_8
0
50
100
150
200
250
0
50
100
150
200
250



Gambar 5. 3 Hasil f*eature maps* pada bagian *feature learning*

5.3 Penentuan Parameter Model

Untuk mendapatkan performa model CNN yang baik, harus dicari parameter-parameter yang terbaik dalam model CNN. Parameter yang di maksud dalam penelitian ini adalah jumlah lapisan konvolusi, jumlah *epoch*, dan *learning rate*. Tujuan dalam penentuan parameter ini adalah untuk mendapatkan performa algoritma CNN yang baik dengan membandingkan nilai setiap parameter.

5.3.1 Pengaruh Jumlah Lapisan Konvolusi

Lapisan konvolusi merupakan bagian yang penting dalam *Convolutional Neural Network*. Lapisan ini berfungsi untuk melakukan ekstraksi fitur dalam citra. Jumlah lapisan konvolusi yang digunakan dalam suatu model CNN mempengaruhi akurasi dari model klasifikasi. Tabel 5.3 merupakan perbandingan akurasi antara model CNN 4 lapisan konvolusi dengan model CNN dengan 8 lapisan konvolusi.

Hasil akurasi yang didapatkan merupakan akurasi CNN dengan *dataset* yang sudah dilakukan *preprocessing* sebelumnya. Berdasarkan Tabel 5.3 *learning rate* yang digunakan untuk melatih model adalah 0,0001 dan jumlah *epoch* yang digunakan adalah 100. Dengan penambahan lapisan konvolusi dari 4 *layer*, 6 *layer*, hingga 8 *layer* terlihat bahwa ada peningkatan nilai akurasi pada model CNN,

Tabel 5. 2 Keluaran model CNN

Layer	***Type***	***Output Shape***	***Parameter***
1	*Convolutional layer 1*	(*None*, 198, 198, 32)	896
2	*Convolutional layer 2*	(*None*, 196, 196, 32)	9248
3	*Maxpooling layer 1*	(*None*, 98, 98, 32)	0
4	*Convolutional layer 3*	(*None*, 96, 96, 64)	18496
5	*Convolutional layer 4*	(*None*, 94, 94, 64)	36928
6	*Maxpooling layer 2*	(*None*, 47, 47, 64)	0
7	*Convolutional layer 5*	(*None*, 45, 45, 128)	73856
8	*Convolutional layer 6*	(*None*, 43, 43, 128)	147584
9	*Maxpooling layer 3*	(*None*, 21, 21, 128)	0
10	*Convolutional layer 7*	(*None*, 19, 19, 256)	295168
11	*Convolutional layer 8*	(*None*, 17, 17, 256)	590080
12	*Maxpooling layer 4*	(*None*, 8, 8, 256)	0
13	*Flatten*	(*None*, 16384)	0
14	*Dropout layer*	(*None*, 16384)	0
15	*Dense layer 1*	(*None*, 512)	8389120
16	*Dense layer 2*	(*None*, 1)	513
Total Parameter			9.561.889

Tabel 5. 3 Perbandingan akurasi berdasarkan jumlah lapisan konvolusi

Jumlah Lapisan Konvolusi	***Train Accuracy***	***Validation Accuracy***	***Precision***	***Recall***	***Specificity***
4 *layer*	95,35 %	94,33 %	98,25 %	94,42 %	93,99 %
6 *layer*	95,38 %	95,03 %	98,15 %	95,44 %	93,56 %
8 *layer*	96,18 %	96,02 %	97,83 %	97,12 %	92,27 %
10 *layer*	96,51 %	95,22 %	98,1 %	95,74 %	93,35 %

meskipun peningkatan akurasi tersebut tidak terlalu signifikan. Namun saat lapisan konvolusi ditambah menjadi 10 *layer* konvolusi akurasi validasinya menurun sehingga citra yang diprediksi dengan benar juga menurun. Sehingga arsitektur

CNN dengan jumlah lapisan konvolusi sebanyak 8 lapisan dipilih karena performa yang dihasilkan sedikit lebih baik dibandingkan dengan yang lain.

5.3.2 Pengaruh Jumlah *Epoch*

Seperti yang sudah dijelaskan pada subsubbab 3.4.2, bahwa *epoch* merupakan suatu kondisi ketika seluruh *dataset* sudah melewati proses pembelajaran yaitu melalui proses *forward* dan *backward* dalam sekali putaran. Dalam CNN jumlah *epoch* merupakan parameter yang mempengaruhi hasil akurasi pada suatu algoritma CNN. Jumlah *epoch* pada setiap model tidak dapat disamaratakan karena tidak ada jumlah pasti untuk *epoch* yang dapat menentukan performa CNN yang benar-benar baik. Tabel 5.4 merupakan perbandingan performa model berdasarkan jumlah *epoch*.

Proses pelatihan dengan mengubah jumlah *epoch* ini digunakan model CNN dengan lapisan konvolusi sebanyak delapan *layer* dan *learning rate* 0.0001. Berdasarkan Tabel 5.4 dapat dilihat bahwa semakin banyak jumlah *epoch* yang digunakan untuk melatih model, semakin meningkat nilai akurasinya. Untuk nilai presisi, *recall*, dan spesifisitas juga mengalami peningkatan saat jumlah *epoch* yang digunakan bertambah. Namun dengan penambahan jumlah *epoch* juga memiliki kekurangan yaitu lamanya proses pelatihan model. Sehingga semakin banyak jumlah *epoch* proses pelatihan akan semakin lama.

5.3.3 Pengaruh *Learning Rate*

Learning rate atau tingkat pembelajaran merupakan salah satu parameter dalam proses pelatihan data untuk menghitung nilai koreksi bobot atau *weight* selama proses pelatihan. Penggunaan *learning rate* juga mempengaruhi akurasi pada model CNN. Tabel 5.5 merupakan perbandingan performa model berdasarkan nilai *learning rate*.

Proses *training* dengan mengubah parameter *learning rate* dilakukan pada *dataset* yang sudah di *preprocessing*, dengan arsitektur CNN yang terdiri dari 8 lapisan konvolusi dengan jumlah *epoch* 100. Berdasarkan Tabel 5.5 dapat dilihat bahwa penggunaan tingkat pembelajaran 0,00001 akurasi yang didapatkan lebih kecil dibandingkan dengan tingkat pembelajaran 0,0001. Hal ini dapat terjadi

Tabel 5. 4 Perbandingan akurasi berdasarkan jumlah *epoch*

Jumlah *Epoch*	*Train Accuracy*	*Validation Accuracy*	*Precision*	*Recall*	*Specificity*
25	92,24 %	94,56 %	96,80 %	96,22 %	88,63 %
75	95,73 %	95,83 %	97,65 %	97 %	91,63 %
100	96,18 %	96,06 %	97,83 %	97,12 %	92,27 %

Tabel 5. 5 Perbandingan akurasi berdasarkan perubahan *learning rate*

Learning Rate	*Train Accuracy*	*Validation Accuracy*	*Precision*	*Recall*	*Specificity*
0,001	94,38 %	93,67 %	98,79 %	93,05 %	95,92 %
0,0001	96,18 %	96,06 %	97,83 %	97,12 %	92,27 %
0,00001	93,23 %	93,81 %	97,94 %	94,06 %	92,92 %

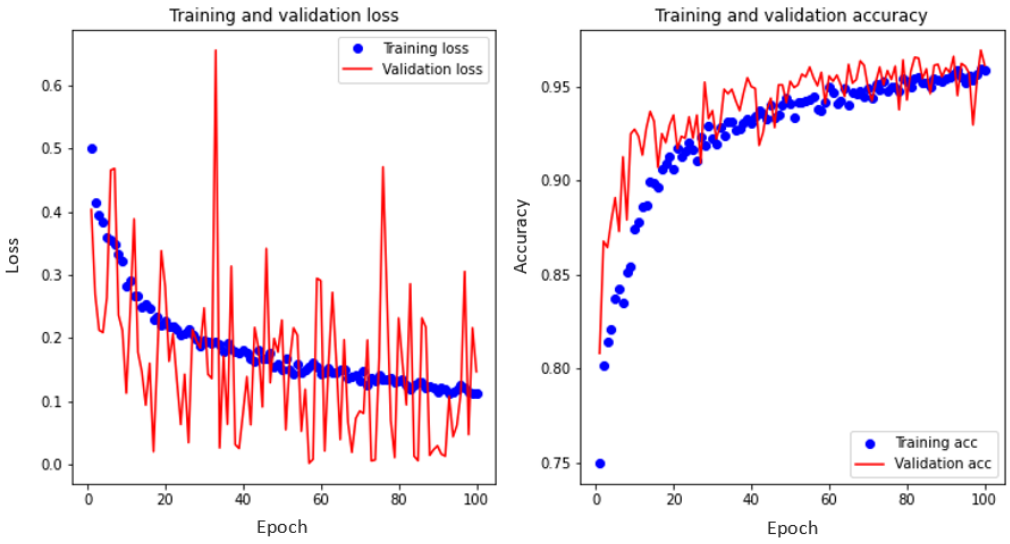
karena lambatnya proses konvergensi nilai *loss* pada saat pelatihan data. Nilai *recall* yang dihasilkan dengan *learning rate* 0,0001 juga lebih besar, namun untuk presisi dan spesifisitas model dengan *learning rate* sebesar 0,00001 lebih unggul. Kemudian nilai pembelajaran yang lebih besar yaitu 0,001 menghasilkan akurasi yang tidak optimal, hal tersebut terjadi karena nilai *loss* semakin meningkat ketika proses iterasi berjalan pada saat *training*. Nilai presisi dan spesifisitas yang dihasilkan dengan *learning rate* 0,001 justru lebih bagus dibandingkan saat *learning rate* 0,0001, namun untuk nilai *recall* lebih unggul saat *learning rate* 0,0001.

5.4 Model Hasil *Training*

Setelah melalui proses pelatihan atau *training* dalam algoritma CNN, didapatkan akurasi dari model yang diusulkan. Proses pelatihan yang digunakan dalam alrogitma CNN ini jumlah *epoch* yang digunakan sebanyak 100 *epoch* dengan *learning rate* 0,0001. Gambar 5.4 merupakan grafik akurasi dan *loss* dari *training* dan *validation*. Berdasarkan Gambar 5.4 akurasi dari *training* dan *validation* mencapai sekitar 96% dengan nilai *loss* keduanya sekitar 0,1. Dengan jumlah *epoch* yang cukup banyak dibutuhkan waktu yang cukup lama juga yaitu sekitar 2 – 3 jam untuk melakukan proses *training*.

Tabel 5.6 merupakan perbandingan performa model yang dilakukan dalam penelitian sebelumnya dengan model yang digunakan dalam tugas akhir ini. Perbandingan tersebut dibuat dengan hasil yang diperoleh oleh O. Stephen *et al* [9] karena *dataset* yang digunakan sama. Pembagian data latih dan data validasi yang dilakukan O. Stephen *et al* [9] juga dilakukan dalam tugas akhir ini dengan pembagian jumlah data yang sama. Arsitektur yang diusulkan juga digunakan untuk melatih *dataset* dalam tugas akhir ini, namun arsitektur tersebut dimodifikasi untuk mendapatkan hasil yang terbaik. Dalam penelitian yang dilakukan dalam penelitian tersebut tidak melakukan *preprocessing* pada *dataset* yang digunakan. Sedangkan dalam tugas akhir ini tahap *preprocessing* citra ditambahkan untuk meningkatkan kualitas citra dan meningkatkan performa model.

Berdasarkan Tabel 5.6, terlihat perbedaan performa yang dihasilkan dalam tugas akhir ini dengan penelitian-penelitian sebelumnya. Performa model dalam tugas akhir ini menghasilkan akurasi *validation* yang lebih tinggi dibandingkan



(a) (b)

Gambar 5. 4 (a) Grafik *training and validation accuracy*; (b) Grafik *training and validation loss*

dengan performa model dalam penelitian O. Stephen *et al* [9], E. Ayan [19], dan G. Verma [20]. Nilai akurasi validasi pada penelitian ini sebenarnya sama seperti nilai akurasi tes pada penelitian V. Chouhan *et al* [21], nilai akurasi tes pada penelitian V. Chouhan *et al* merupakan nilai akurasi pengujian yang diambil dari folder tes, jika pada penelitian ini akurasi pengujiannya diambil dari berkas validasi karena diawal *dataset* hanya dibagi menjadi dua set yaitu *train* dan val. Nilai akurasi tes yang diperoleh oleh V. Chouhan *et al* (2020) lebih tinggi yaitu 96,39%, akurasinya lebih tinggi karena algoritma yang digunakan merupakan *ensemble model* yaitu menggunakan beberapa arsitektur secara bersamaan untuk mengklasifikasikan satu gambar lalu hasilnya diperoleh dari nilai keluaran yang terbanyak. Namun jika dilihat berdasarkan performa masing-masing arsitektur modelnya hasil yang didapatkan lebih rendah dibandingkan dengan model dalam penelitian ini.

Selanjutnya untuk akurasi *train* jika dibandingkan dengan penelitian yang dilakukan O. Stephen *et al* (2019) dan G. Verma (2020) hasilnya lebih unggul pada penelitian ini yaitu sekitar 2% lebih unggul dari G. Verma dan 1% lebih unggul dari O. Stephen. Untuk nilai presisi jika dibandingkan dengan penelitian yang dilakukan V. Chouhan *et al* (2020), pada penelitian ini lebih unggul sekitar 4%. Namun nilai *recall* yang diperoleh oleh V. Chouhan *et al* (2020) lebih tinggi yaitu sekitar 2% dibandingkan penelitian ini. Sedangkan nilai *recall* yang diperoleh E. Ayan (2019) dengan arsitektur Xception sekitar 12% lebih rendah dan untuk arsitektur VGG16 sekitar 15% lebih rendah dibandingkan nilai *recall* dari model pada penelitian ini. Untuk spesifisitas pada penelitian ini lebih unggul dibandingkan penelitian-penelitian sebelumnya seperti yang tertera pada Tabel 5.6. Jadi dapat dikatakan bahwa penelitian ini lebih unggul dalam akurasi *train*, validasi (untuk beberapa

penelitian), presisi, dan spesifisitas dibandingkan dengan penelitian sebelumnya. Namun untuk nilai *recall* yang diperoleh lebih rendah dibandingkan penelitian sebelumnya.

Tabel 5. 6 Perbandingan performa model dengan penelitian sebelumnya

Model	***Accuracy (%)***			***(%)***		
	Train	***Test***	***Validation***	***Presicion***	***Recall***	***Specificity***
Model CNN 16 layer (Penelitian ini, 2020)	96,18	-	96,06	97,83	97,12	92,27
O. Stephen et al (2019)	95,31	-	93,73	-	-	-
Xception (E. Ayan, 2019)	-	-	82	-	85	76
VGG16 (E. Ayan, 2019)	-	-	87	-	82	91
G. Verma (2020)	94,36	-	92,89	-	-	-
V. Chouhan et al (2020)	-	96,39	-	93,28	99,62	-

BAB VI KESIMPULAN DAN SARAN

6.1 Kesimpulan

Pada tugas akhir ini, diajukan sebuah sistem untuk mendeteksi citra pneumonia dari sebuah modalitas pencitraan X-ray. Tahapan yang dilakukan dalam tugas akhir ini adalah menyiapkan *dataset*, *preprocessing*, dan deteksi menggunakan *Convolutional Neural Network*. Pada tahapan *preprocessing*, dilakukan pemfilteran pada citra menggunakan median filter serta peningkatan kontras pada citra menggunakan metode *Contrast Limited Adaptive Histogram Equalization* (CLAHE). Setelah tahap *preprocessing,* citra masuk ke dalam tahap *Convolutional Neural Network* (CNN) untuk dilakukan proses ekstraksi fitur dan klasifikasi. Pada tahap CNN akan dideteksi citra normal dan citra pneumonia. Dari serangkaian kerangka kerja dalam sistem ini kemudian dilakukan pengujian menggunakan *dataset* yang sudah diolah. Hasil yang didapatkan dari pengujian sistem ini diperoleh akurasi validasi sebesar 96,06 % dan akurasi pelatihan sebesar 96,18 %. Adanya tahap *preprocessing* meningkatkan performa model dalam mengklasifikasian citra. Pengujian juga dilakukan dengan menghitung nilai presisi, *recall,* spesifisitas serta menganalisa berdasarkan kurva ROC. Nilai presisi, *recall,* dan spesifisitas yang didapatkan berturut-turut adalah 97,83 %, 97,12 %, dan 92,27 %. Kurva ROC yang dihasilkan menunjukkan bahwa grafik mendekati titik (0,1) dengan *Area Under Curve* (AUC) sebesar 0,95.

Dibandingkan dengan penelitian-penelitian sebelumnya penelitian ini lebih unggul dalam akurasi *train*, validasi (untuk beberapa penelitian), presisi, dan spesifisitas dibandingkan dengan penelitian sebelumnya. Namun untuk nilai *recall* yang diperoleh lebih rendah dibandingkan penelitian sebelumnya. Adanya modifikasi arsitektur CNN dan tahapan *preprocessing* pada citra masukan menyebabkan performa keseluruhan sistem lebih baik.

Pengujian dengan *balancing data* dilakukan untuk membandingkan performa algoritma CNN berdasarkan kondisi *dataset* yang digunakan untuk proses *training*. Hasil yang didapatkan ketika data latih dalam kondisi seimbang akurasinya memang lebih kecil dibandingkan saat data latih dalam kondisi seimbang. Namun nilai presisi dan spesifisitas lebih besar saat datanya dalam kondisi yang seimbang. Nilai presisi yang rendah menunjukkan bahwa nilai *False Positive* (FP) lebih tinggi. Karena tingginya nilai FP maka akan berpengaruh juga pada nilai spesifisitas. Hal ini dapat terjadi karena adanya *overfitting* pada salah satu kelas ketika kondisi *dataset* tidak seimbang.

6.2 Saran

Untuk meningkatkan performa dan mendapat nilai akurasi yang lebih tinggi lagi dalam mengklasifikasikan citra, maka bisa ditambahkan metode lain pada tahap *preprocessing*. Dengan adanya modifikasi pada algoritma CNN serta dengan

penentuan *hyperparameter* yang lebih variatif, diharapkan bisa mendapatkan akurasi yang lebih tinggi. Aplikasi *transfer learning* dapat diterapkan untuk meningkatkan performa model. Penerapan berbagai teknik augmentasi data yang relevan juga dapat dilakukan agar data yang digunakan lebih bervariasi. Serta dengan penggunaan *dataset* yang berasal dari Rumah Sakit juga bisa dilakukan untuk menambah data dan menghindari adanya *overfitting*, data dari Rumah Sakit tersebut juga dapat digunakan untuk mempelajari mengenai ciri-ciri citra pneumonia khususnya pada masyarakat di Indonesia.

DAFTAR PUSTAKA

[1] World Health Organization, Pneumonia [Fact Sheet], WHO, 2019, https://www.who.int/news-room/fact-sheets/detail/pneumonia.

[2] J. A. G. Scott, W. A. Brooks, J. S. M. Peiris, D. Holtzman, and E. K. Mulhollan, "Pneumonia Research to Reduce Childhood Mortality in the Developing World.".

[3] Franquet T, "Imaging of pneumonia: trends and algorithms." Eur Respir J.2001; 18:196-208.

[4] K. F. O'Grady, P. J. Torzillo, K. Frawley, and A. B. Chang, "The Radiological Diagnosis of Pneumonia in Children.", 2014.

[5] M. Young, T. J. Marrie, "Interobserver variability in the interpretation of chest roentgenograms of patients with possible pneumonia.", Arch Intern Med 154, 1994.

[6] A. Enes and U. M. Halil, "Diagnosis of Pneumonia from Chest X-Ray Images using Deep Learning.", IEEE, 2019.

[7] H.H. Aung, A. Sivakumar, S.K. Gholami, S.P. Venkateswaran, B. Gorain, Shadab Md, "Nanotechnology-Based Targeted Drug Delivery Systems for Lung Cancer", 2019.

[8] Kelly MS, Sandora TJ. Community-Acquired Pneumonia. In: Kliegman RM, editor. Nelson Textbook of Pediatrics. 20th ed. Philadelphia; 2016.

[9] O. Stephen, M. Sain, U. J. Maduh, and D. Jeong, " An Efficient Deep Learning Approach to Pneumonia Classification in Healthcare.", Journal of Healthcare Engineering Volume 2019.

[10] Z. A. Matondang, "Penerapan Metode Contrast Limited Adaptive Histogram Equalization (CLAHE) Pada Citra Digital Untuk Memperbaiki Gambar X-Ray.", Publikasi Ilmiah Teknologi Informasi Neumann, 2018.

[11] J. Wang, Y. Ma, L. Zhang, R. X. Gao, D. Wu, "Deep Learning for Smart Manufacturing : Methods and Applications", Journal of Manufacturing Systems Vol. 48, Part C, July 2018.

[12] A. Hidaka, and T. Kurita, "Consecutive Dimensionality Reduction by Canonical Correlation Analysis for Visualization of Convolutional Neural Networks", Proceedings of the ISCIE International Symposium on Stochastic Systems Theory and its Applications, Volume 2017 Pages 160-167, 2017.

[13] I. E. Poletaev, K. S. Pervunin, M. P. Tokarev, "Artificial neural network for bubbles pattern recognition on the images", Journal of Physics: Conference Series 754, 2016.

[14] F. H. Achmad, P. K. Agi, dan A. Tri, "Analisis Perbandingan Performa *Web Service* Menggunakan Bahasa Pemrograman Python, PHP, dan Perl pada *Client* Berbasis Android.", Jurnal Pengembangan Teknologi Informasi dan

Ilmu Komputer Vol. 2, No. 1, Januari 2018.
[15] D. S. Kermany, M. Goldbaum, W. Cai, M. A. Lewis, H. Xia, and K. Zhang, "Identifying Medical Diagnoses and Treatable Diseases by Image-Based Deep Learning", Cell 172, 1122–1131, 2018.
[16] A. Y. Khairul, dan D. S. Ronda, "Penerapan Metode Median Filtering dan Histogram Equalization untuk Meningkatkan Kualitas Citra Radiografi.", Jurnal Riset Komputer (JURIKOM), Vol. 5 No. 3, Juni 2018.
[17] D. P. Kingma, and Ba, J. "Adam : A Method for Stochastic Optimization.", arXiv:1412.6980, 2015.
[18] O. Cici, Indwiarti, dan S. Yulian, "Analisis Prediksi Churn Menggunakan Metode Logistic Regression dan Algoritma Decision Tree.", e-Proceeding of Engineering Vol. 2, No. 2, Agustus 2015.
[19] E. Ayan, H. Murat, " Diagnosis of Pneumonia from Chest X-Ray Images using Deep Learning.", 2019 Scientific Meeting on Electrical-Electronics & Biomedical Engineering and Computer Science (EBBT), IEEE, 2019.
[20] G. Verma, P. Shiva, "Pneumonia Classification using Deep Learning in Healthcare.", International Journal of Innovative Technology and Exploring Engineering (IJITEE) ISSN: 2278-3075, Volume-9 Issue-4, February 2020.
[21] V. Chouchan, S. K. Singh, A. Khamparia, D. Gupta, P. Tiwari, C. Moreira, R. Damasevicius, and V. H. C. Albuquerque, " A Novel Transfer Learning Based Approach for Pneumonia Detection in Chest X-ray Images.", Appl. Sci. 10, 599, 2020.

LAMPIRAN

a. Pengaruh kernel size pada filter median terhadap algoritma CNN

Ukuran kernel	***Train Accuracy***	***Validation Accuracy***	***Precision***	***Recall***	***Specificity***
3x3	96,18 %	96,02 %	97,83 %	97,12 %	92,27 %
5x5	95,14 %	94,66 %	97,96 %	95,14 %	92,92 %
7x7	95,41 %	95,64 %	97,81 %	96,58 %	92,27 %

b. Pengaruh nilai *clip limit* pada CLAHE terhadap algoritma CNN

Clip Limit	***Train Accuracy***	***Validation Accuracy***	***Precision***	***Recall***	***Specificity***
0,005	96,16 %	96,02 %	97,71 %	97,18 %	91,85 %
0,01	96,18 %	96,02 %	97,83 %	97,12 %	92,27 %
0,02	96,08 %	94,28 %	98,19 %	94,42 %	93,78 %
0,5	96,18 %	94,75 %	98,02 %	95,20 %	93,13 %

c. Pengaruh modifkasi arsitektur

Arsitektur CNN	***Train Accuracy***	***Validation Accuracy***	***Precision***	***Recall***	***Specificity***
4 *layer* konvolusi	95,35 %	94,33 %	98,25 %	94,42 %	93,99 %
6 *layer* konvolusi	95,38 %	95,03 %	98,15 %	95,44 %	93,56 %
10 *layer* konvolusi	96,51 %	95,22 %	98,10 %	95,74 %	93,35 %
Layer dense (256)	94,63 %	94,75 %	96,75 %	96,52 %	88,41 %
Penambahan *dropout* setelah *pooling layer*	94,09 %	95,22 %	97,28 %	96,58 %	90,34 %

BIODATA PENULIS

Penulis dilahirkan di Kabupaten Lumajang pada tanggal 5 Juni 1998. Penulis merupakan anak pertama dari dua bersaudara. Penulis menempuh pendidikan sekolah dasar di SDN Wonorejo 01 Lumajang dan meneruskan ke jenjang sekolah menengah pertama di SMPN 02 Lumajang dan melanjutkan ke sekolah menengah atas SMAN 01 Lumajang. Kemudian pada tahun 2016 setelah lulus dari jenjang sekolah menengah atas, penulis melanjutkan ke jenjang pendidikan yang lebih tinggi di Departemen Teknik Biomedik Institut Teknologi Sepuluh Nopember, dan selama masa perkuliahan aktif dalam organisasi dan kepanitiaan yang ada di lingkup Institut Teknologi Sepuluh Nopember. Penulis juga aktif sebagai asisten Laboratorium Instrumentasi dan Pengolahan Sinyal Biomedika Departemen Teknik Biomedik.

E-mail : alif.hilmi16@mhs.bme.its.ac.id

BIOMEDICAL ENGINEERING ITS
2020