



**IMPLEMENTASI *DEEP LEARNING* PADA SISTEM
KLASIFIKASI PENYAKIT PARU BERDASARKAN FOTO
RONTGEN MUNGGUNAKAN METODE *CONVOLUTIONAL
NEURAL NETWORK (CNN)***

SKRIPSI

Oleh

Muhammad Noor Dwi Eldianto

NIM 152410101090

PROGRAM STUDI SISTEM INFORMASI

UNIVERSITAS JEMBER

2019



**IMPLEMENTASI *DEEP LEARNING* PADA SISTEM
KLASIFIKASI PENYAKIT PARU BERDASARKAN FOTO
RONTGEN MUNGUNAKAN METODE *CONVOLUTIONAL
NEURAL NETWORK (CNN)***

SKRIPSI

diajukan guna melengkapi tugas akhir dan memenuhi salah satu syarat untuk menyelesaikan Pendidikan Sarjana (S1) Program Studi Sistem Informasi Universitas Jember dan mencapai gelar Sarjana Komputer

Oleh

Muhammad Noor Dwi Eldianto

NIM 152410101090

PROGRAM STUDI SISTEM INFORMASI

UNIVERSITAS JEMBER

2019

PERSEMBAHAN

Skripsi ini saya persembahkan untuk:

1. Allah SWT yang senantiasa memberikan rahmat dan hidayah-Nya untuk mempermudah dan melancarkan dalam mengerjakan skripsi;
2. Ibunda Elok Nurnahdiah dan Ayahanda Didiek Pantja;
3. Saudaraku Eldira Rizka dan Aditya Muslim;
4. Keluarga besar;
5. Sahabat-sahabatku dengan dukungan dan doanya;
6. Guru-guruku baik dari pendidikan formal maupun informal;
7. Almamater Program Studi Sistem Informasi, Fakultas Ilmu Komputer, Universitas Jember.

MOTTO

“Sesungguhnya urusannya-Nya apabila Dia menghendaki sesuatu hanyalah berkata kepadanya, Jadilah, Maka jadilah ia”.

- QS. Yasin:82 -

“The journey of a thousand miles begins with one step”

- Lao Tzu -

“Try Not to Become a Man of Success. Rather Become a Man of Value.”

- Albert Einstein -

“Jadilah orang yang tersenyum diakhir episode”

- Anonymous -

PERNYATAAN

Saya yang bertanda tangan dibawah ini:

Nama : Muhammad Noor Dwi Eldianto

NIM : 152410101090

Menyatakan dengan sesungguhnya bahwa karya ilmiah yang berjudul “IMPLEMENTASI DEEP LEARNING PADA SISTEM KLASIFIKASI PENYAKIT PARU BERDASARKAN FOTO RONTGEN MUNGGUNAKAN METODE CONVOLUTIONAL NEURAL NETWORK (CNN)”, adalah benar-benar hasil karya sendiri, kecuali jika dalam pengutipan substansi disebutkan sumbernya, belum pernah diajukan pada institusi mana pun, dan bukan karya jiplakan. Saya bertanggung jawab atas keabsahan dan kebenaran isinya sesuai dengan sikap ilmiah yang harus dijunjung tinggi.

Demikian pernyataan ini saya buat dengan sebenarnya, tanpa adanya tekanan dan paksaan dari pihak manapun serta bersedia mendapat sanksi akademik jika di kemudian hari pernyataan ini tidak benar.

Jember, 14 Juni 2019

Yang menyatakan,

Muhammad Noor Dwi Eldianto

NIM 152410101090

SKRIPSI

**IMPLEMENTASI *DEEP LEARNING* PADA SISTEM
KLASIFIKASI PENYAKIT PARU BERDASARKAN FOTO
RONTGEN MUNGUNAKAN METODE *CONVOLUTIONAL
NEURAL NETWORK* (CNN)**

Oleh

Muhammad Noor Dwi Eldianto

NIM 152410101090

Pembimbing

Dosen Pembimbing Utama : Prof. Dr.Saiful Bukhori, ST., M.Kom.

Dosen Pembimbing Pendamping : Januar Adi Putra S.Kom., M.Kom.

PENGESAHAN PEMBIMBING

Skripsi yang berjudul “IMPLEMENTASI DEEP LEARNING PADA SISTEM KLASIFIKASI PENYAKIT PARU BERDASARKAN FOTO RONTGEN MUNGUNAKAN METODE CONVOLUTIONAL NEURAL NETWORK (CNN)”, telah diuji dan disahkan pada:

hari, tanggal : Jum’at, 28 Juni 2019

tempat : Fakultas Ilmu Komputer Universitas Jember.

Disetujui oleh:

Pembimbing I

Pemimbing II

Prof. Dr.Saiful Bukhori, ST., M.Kom.

196811131994121001

Januar Adi Putra S.Kom., M.Kom.

760017015

PENGESAHAN

Skripsi berjudul “Implementasi Deep Learning pada Sistem Klasifikasi Penyakit Paru Berdasarkan Foto Rontgen Menggunakan Metode Convolutional Neural Network (CNN)”, telah diuji dan disahkan pada:

hari, tanggal : Jum’at, 28 Juni 2019

tempat : Fakultas Ilmu Komputer Universitas Jember

Disetujui oleh :

Penguji I,

Penguji II,

Prof. Drs.Slamin, M.Comp.Sc., Ph.D.

NIP 196704201992011001

Gayatri Dwi Santika S.SI., M.Kom

NIP 760017013

Mengesahkan

Dekan Fakultas Ilmu Komputer,

Prof. Dr. Saiful Bukhori, ST., M.Kom

NIP 196811131994121001

RINGKASAN

Implementasi Deep Learning pada Sistem Klasifikasi Penyakit Paru Berdasarkan Foto Rontgen Menggunakan Metode Convolutional Neural Network (CNN); Muhammad Noor Dwi Eldianto, 152410101090; 2019, 80 halaman; Program Studi Sistem Informasi, Fakultas Ilmu Komputer, Universitas Jember.

Indonesia merupakan salah satu negara penderita penyakit paru terbesar di dunia. Setiap tahunnya penderita penyakit paru di Indonesia selalu mengalami peningkatan. Permasalahan yang terjadi adalah diagnosa penyakit paru masih dilakukan secara manual, hal ini menyebabkan akurasi yang kurang baik dan subjektif. Perbedaan persepsi antar dokter spesialis paru dapat menyebabkan hasil diagnosa yang berbeda. Sehingga dibutuhkan teknologi yang dapat membantu dokter spesialis paru untuk menganalisa foto rontgen dengan cepat dan akurat dengan penggunaan teknologi informasi berbasis komputer dan data.

Teknologi yang dapat digunakan untuk mendiagnosa penyakit paru diantaranya adalah dengan menggunakan *computer vision* dan *deep learning*. Dalam penelitian ini menggunakan salah satu metode *deep learning* yaitu *Convolutional Neural Network* yang akan diimplementasikan ke dalam sistem yang dapat mengklasifikasikan foto rontgen paru ke dalam tiga kelas : Normal, *Pneumonia* dan *Tuberculosis*. Sistem yang dibangun berbasis *web apps*, *web* dipilih agar pengguna dapat dengan mudah untuk menggunakannya tanpa harus *install* aplikasi tersebut dahulu.

Dalam penelitian ini menggunakan 300 data *training* yang terdiri dari 100 data di masing-masing kelas, selanjutnya data training akan dibagi kedalam 3 skenario data yang berbeda. Data *test* yang digunakan sebanyak 57 data. Berdasarkan hasil pembahasan akurasi model terbaik sebesar 85.96% didapatkan setelah melakukan *epoch* sebanyak 50 dan menggunakan skenario data 90:10. Akurasi model didapatkan setelah dilakukan *testing* pada model menggunakan *confusion matrix*.

PRAKATA

Puji syukur kehadiran Allah SWT atas limpahan rahmat dan karunia-Nya sehingga penulis dapat menyelesaikan skripsi dengan judul “Implementasi Deep Learning pada Sistem Klasifikasi Penyakit Paru Berdasarkan Foto Rontgen Menggunakan Metode Convolutional Neural Network (CNN)”. Skripsi ini disusun untuk memenuhi salah satu syarat menyelesaikan pendidikan Strata Satu (S1) pada Program Studi Sistem Informasi Universitas Jember:

Penyusunan skripsi ini tidak lepas dari dukungan berbagai belah pihak. Oleh karena itu penulis menyampaikan terima kasih kepada:

1. Prof. Dr.Saiful Bukhori, ST., M.Kom selaku dosen pembimbing utama dan dekan Fakultas Ilmu Komputer;
2. Januar Adi Putra S.Kom., M.Kom selaku dosen pembimbing pendamping yang telah meluangkan waktu, pikiran, dan perhatian dalam penulisan skripsi ini;
3. Seluruh Bapak dan Ibu dosen beserta staf karyawan di Fakultas Ilmu Komputer Universitas Jember;
4. dr. Angga Mardro Raharjo, Sp.P selaku dokter spesialis paru dan dosen Fakultas kedokteran Universitas Jember yang telah memberikan saran-saran terbaiknya dalam pemilihan penyakit paru.
5. Ibunda Elok Nurnahdiah dan Ayahanda Didiek Pantja yang selalu mendukung serta mendoakan dan menjadi orang tua yang luar biasa;
6. Saudaraku Eldira Rizka dan Aditya Muslim yang telah memberikan dukungan semangat dan selalu menginspirasi;
7. Keluarga besar yang telah mendukung dan mendoakan;
8. Keluarga HIMASIF (Himpunan Mahasiswa Sistem Informasi) yang selalu menjadi rumah untuk mengembangkan diri dan berproses bersama.
9. Keluarga SELECTION angkatan 2015 atas motivasi yang selalu mengalir;

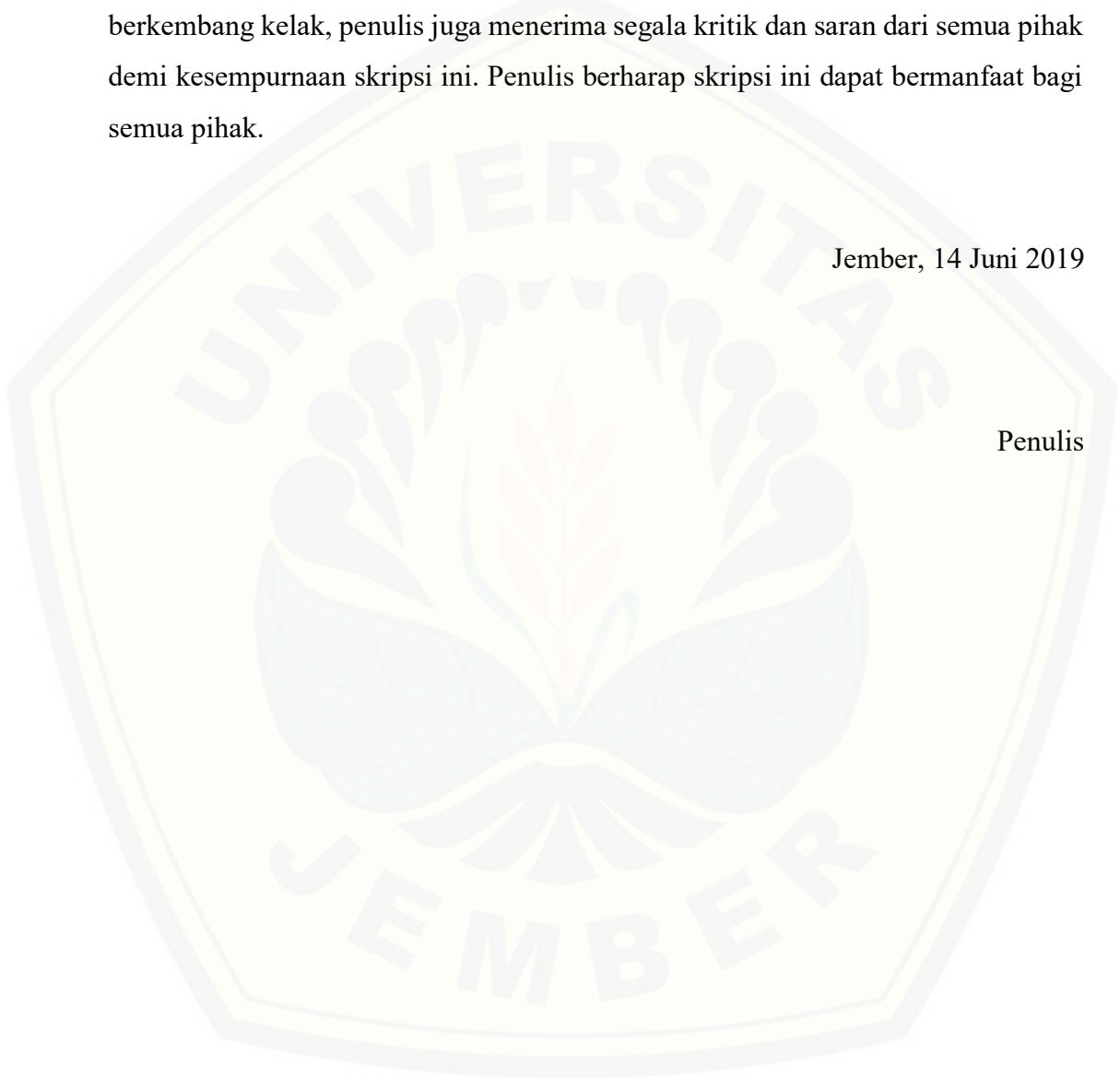
10. Teman-teman Fakultas Ilmu Komputer di semua angkatan atas bantuan dan dukungannya;

11. Semua pihak yang tidak bisa disebutkan satu persatu.

Dengan harapan bahwa penelitian ini nantinya akan terus berlanjut dan berkembang kelak, penulis juga menerima segala kritik dan saran dari semua pihak demi kesempurnaan skripsi ini. Penulis berharap skripsi ini dapat bermanfaat bagi semua pihak.

Jember, 14 Juni 2019

Penulis



DAFTAR ISI

SKRIPSI.....	i
SKRIPSI.....	ii
PERSEMBAHAN.....	iii
MOTTO.....	iv
PERNYATAAN.....	v
SKRIPSI.....	vi
PENGESAHAN PEMBIMBING.....	vii
PENGESAHAN.....	viii
RINGKASAN.....	ix
PRAKATA.....	x
DAFTAR ISI.....	xii
DAFTAR TABEL.....	xv
DAFTAR GAMBAR.....	xvi
BAB 1. PENDAHULUAN.....	1
1.1. Latar Belakang.....	1
1.2. Rumusan Masalah.....	3
1.3. Tujuan Penelitian.....	3
1.4. Batasan Masalah.....	4
1.5. Sistematika Penulisan.....	4
BAB 2. TINJAUAN PUSTAKA.....	6
2.1. Penelitian Terdahulu.....	6
2.2. Penyakit Paru.....	7
2.3. Foto Rontgen.....	10

2.4.	<i>Computer Vision</i>	11
2.5.	<i>Deep Learning</i>	12
2.6.	Convolutional Neural Network (CNN).....	12
BAB 3.	METODOLOGI PENELITIAN	19
3.1.	Tempat Penelitian.....	19
3.2.	Waktu Penelitian	19
3.3.	Jenis Penelitian	19
3.4.	Tahapan Penelitian	19
3.4.1.	Studi Literatur	20
3.4.2.	Pengumpulan Data	20
3.4.3.	Perancangan Arsitektur CNN.....	20
3.4.4.	Implementasi Arsitektur CNN	20
3.4.5.	Pengujian Model	21
3.4.6.	Integrasi Sistem.....	21
3.4.7.	Pengujian Sistem.....	21
BAB 4.	DESAIN DAN IMPLEMENTASI	22
4.1.	Analisa Kebutuhan Sistem	22
4.2.	Desain Sistem	22
4.2.1.	Desain Arsitektur CNN	23
4.2.2.	Business Process Model Notation (BPMN).....	25
4.2.3.	<i>Flowchart</i>	27
4.2.4.	<i>Class Diagram</i>	31
4.2.5.	<i>Entity Relationship Diagram (ERD)</i>	33
4.3.	Desain Tampilan Integrasi Sistem.....	33
4.4.	Implementasi	35

4.1.1.	Implementasi <i>frontend</i>	36
4.1.2.	Implementas <i>backend</i>	36
4.5.	Pengujian Sistem	36
BAB 5.	HASIL DAN PEMBAHASAN	40
5.1.	Hasil Implementasi Arsitektur CNN	40
5.2.	Hasil Pengujian Model	44
5.3.1.	Perangkat Pengujian.....	44
5.3.2.	Hasil <i>Training</i> data Berdasarkan Epoch dan Skenario Data.....	44
5.3.3.	Hasil <i>Testing</i> dengan Data Baru.....	56
5.3.	Hasil Integrasi Sistem.....	61
5.4.1.	Tampilan <i>Dashboard</i>	63
5.4.2.	Tampilan Menu <i>Classification</i>	64
5.4.3.	Tampilan Menu <i>History</i>	65
5.4.4.	Tampilan Menu <i>Labeling</i>	66
BAB 6.	PENUTUP	67
6.1.	Kesimpulan.....	67
6.2.	Saran.....	67
DAFTAR PUSTAKA	68
LAMPIRAN	70
A.	<i>Business Process Model Notation (BPMN)</i>	70
B.	<i>Class Diagram</i>	71
C.	<i>NLM-ChinaCXSet-ReadMe</i>	72
D.	<i>NIHCC README_CHESTXRAY</i>	73

DAFTAR TABEL

Tabel 2.1 Penelitian terdahulu.....	6
Tabel 4.1 Pengujian <i>black box</i>	36
Tabel 5.1 Hasil <i>training</i> data dengan <i>epoch 17</i>	48
Tabel 5.2 Hasil <i>Training</i> data dengan <i>epoch 30</i>	52
Tabel 5.3 Hasil <i>Training</i> data dengan <i>epoch 50</i>	55
Tabel 5.4 <i>Confusion matrix</i> , <i>epoch 17</i> , skenario data 70:30	56
Tabel 5.5 <i>Confusion matrix</i> , <i>epoch 17</i> , skenario data 80:20	57
Tabel 5.6 <i>Confusion matrix</i> , <i>epoch 17</i> , skenario data 90:10	57
Tabel 5.7 <i>Confusion matrix</i> , <i>epoch 30</i> , skenario data 70:30	57
Tabel 5.8 <i>Confusion matrix</i> , <i>epoch 30</i> , skenario data 80:20	58
Tabel 5.9 <i>Confusion matrix</i> , <i>epoch 30</i> , skenario data 90:10	58
Tabel 5.10 <i>Confusion matrix</i> , <i>epoch 50</i> , skenario data 70:30	59
Tabel 5.11 <i>Confusion matrix</i> , <i>epoch 50</i> , skenario data 80:20	59
Tabel 5.12 <i>Confusion matrix</i> , <i>epoch 50</i> , skenario data 90:10	59
Tabel 5.13 Hasil pengujian seluruh model.....	61

DAFTAR GAMBAR

Gambar 2.1 Foto <i>rontgen</i> penyakit tuberculosis (http://openi.nlm.nih.gov)	8
Gambar 2.2 abnormal CXR dari MC dataset	9
Gambar 2.3 Foto <i>rotngen</i> penyakit <i>pneumonia</i> (https://nihcc.app.box.com)	10
Gambar 2.4 Foto <i>Rontgen</i> (https://www.alodokter.com,2018).....	11
Gambar 2.5 Arsitektur <i>convolutional neural network</i>	13
Gambar 2.6 <i>Convolution Layer</i> (Medium Samuel Sena, 2017).....	16
Gambar 2.7 Operasi <i>Max Pooling</i> (Medium Samuel Sena, 2017).....	17
Gambar 2.8 <i>Fully connected layer</i>	18
Gambar 3.1 Tahapan penelitian.....	19
Gambar 4.1 Desain Arsitektur CNN.....	23
Gambar 4.2 Proses <i>convolution</i> pertama	24
Gambar 4.3 Proses <i>convolution</i> kedua	24
Gambar 4.4 Proses <i>convolution</i> ketiga	25
Gambar 4.5 <i>Business Process Model Notation</i>	26
Gambar 4.6 <i>Flowchart</i> arsitektur CNN	27
Gambar 4.7 <i>Flowchart</i> proses <i>convolution</i>	28
Gambar 4.8 <i>Flowchart</i> proses <i>ReLU</i>	29
Gambar 4.9 <i>Flowchart</i> proses <i>Max Pooling</i>	30
Gambar 4.10 <i>Flowchart</i> proses <i>flatten</i>	31
Gambar 4.11 <i>Class diagram</i>	32
Gambar 4.12 <i>Entity Relationship Diagram</i>	33
Gambar 4.13 <i>Mockup form upload</i> foto <i>rontgen</i>	34
Gambar 4.14 <i>Mockup</i> hasil prediksi sistem.....	35
Gambar 5.1 Struktur <i>folder</i> data <i>train</i>	40
Gambar 5.2 Fungsi <i>load train</i>	41
Gambar 5.3 inisialisasi variabel.....	42
Gambar 5.4 Pembuatan jaringan CNN	42
Gambar 5.5 <i>Validation accuracy, epoch 17</i> , skenario data 70:30.....	45
Gambar 5.6 <i>Validation Loss, epoch 17</i> , skenario data 70:30	45

Gambar 5.7 <i>Validation accuracy, epoch 17, skenario data 80:20</i>	46
Gambar 5.8 <i>Validation Loss, epoch 17, skenario data 80:20</i>	46
Gambar 5.9 <i>Validation accuracy, epoch 17, skenario data 90:10</i>	47
Gambar 5.10 <i>Validation Loss, epoch 17, skenario data 90:10</i>	47
Gambar 5.11 <i>Validation accuracy, epoch 30, skenario data 70:30</i>	49
Gambar 5.12 <i>Validation Loss, epoch 30, skenario data 70:30</i>	49
Gambar 5.13 <i>Validation accuracy, epoch 30, skenario data 80:20</i>	50
Gambar 5.14 <i>Validation loss, epoch 30, skenario data 80:20</i>	50
Gambar 5.15 <i>Validation accuracy, epoch 30, skenario data 90:10</i>	51
Gambar 5.16 <i>Validation loss, epoch 30, skenario data 90:10</i>	51
Gambar 5.17 <i>Validation accuracy, epoch 50, skenario data 70:30</i>	52
Gambar 5.18 <i>Validation loss, epoch 50, skenario data 70:30</i>	53
Gambar 5.19 <i>Validation accuracy, epoch 50, skenario data 80:20</i>	53
Gambar 5.20 <i>Validation loss, epoch 50, skenario data 80:20</i>	54
Gambar 5.21 <i>Validation accuracy, epoch 50, skenario data 90:10</i>	54
Gambar 5.22 <i>Validation loss, epoch 50, skenario data 90:10</i>	55
Gambar 5.23 <i>fungsi proses integrasi</i>	62
Gambar 5.24 <i>output proses klasifikasi</i>	63
Gambar 5.25 <i>Tampilan menu dashboard</i>	64
Gambar 5.26 <i>Tampilan form upload foto rontgen paru</i>	64
Gambar 5.27 <i>Tampilan hasil prediksi sistem</i>	65
Gambar 5.28 <i>Tampilan menu history</i>	66
Gambar 5.29 <i>Tampilan menu labeling</i>	66

BAB 1. PENDAHULUAN

1.1. Latar Belakang

Paru-paru merupakan salah satu organ penting pada sistem pernapasan yang berfungsi sebagai tempat bertukarnya oksigen dengan karbondioksida di dalam darah (Andini & Upahita, 2018). Permasalahan yang sering terjadi adalah kualitas udara yang telah tercemar akibat gas buang dari kendaraan maupun asap rokok sehingga udara yang dihirup menjadi tidak sehat dan banyak mengandung bibit penyakit yang akan menyebabkan terjadinya berbagai macam penyakit, khususnya penyakit paru-paru (Ismiyati, Marlita, & Saidah, 2014). Penyakit paru-paru merupakan penyakit yang berdampak serius terhadap sistem pernapasan pada manusia yang dapat berakibat fatal apabila tidak segera ditangani dengan serius.

Menurut Departemen Kesehatan Republik Indonesia, penyakit paru termasuk salah satu penyakit yang kritis hingga saat ini. Jumlah penderita penyakit paru pada tahun 2009 sejumlah 1,7 juta orang, tahun 2010 sejumlah 2,3 juta orang, dan tahun 2011 sejumlah 4,7 juta orang. Data tersebut meliputi berbagai macam penyakit paru-paru diantaranya yang terbesar adalah *Tuberkulosis* (TB paru), kanker paru dan *Pneumonia* (radang paru). Jumlah kasus untuk penyakit TB paru pada tahun 2016 dan 2017 masing-masing sebesar 298.128 dan 360.770, sedangkan jumlah kasus pada penyakit radang paru pada tahun 2016 dan 2017 masing-masing sebesar 503.738 dan 447.431, sedangkan menurut WHO pada tahun 2014 untuk kasus kanker paru, kanker paru menjadi penyebab kematian nomor satu pada laki-laki di Indonesia, dan 1 dari 13 orang laki-laki di Indonesia terdiagnosis kanker paru. Untuk melakukan proses diagnosa paru dibutuhkan proses yang cepat dan akurat. Permasalahan yang terjadi adalah diagnosa penyakit paru masih dilakukan secara manual, hal ini menyebabkan akurasi yang kurang baik dan subjektif. Perbedaan persepsi antar dokter spesialis paru dapat menyebabkan hasil diagnosa yang berbeda. Sehingga dibutuhkan teknologi yang dapat membantu dokter

spesialis paru untuk menganalisa foto *rontgen* dengan cepat dan akurat dengan penggunaan teknologi informasi berbasis komputer dan data.

Teknologi yang dapat digunakan untuk mendiagnosa penyakit paru diantaranya adalah dengan menggunakan *computer vision*. Penelitian untuk mengklasifikasikan penyakit paru dengan *computer vision* telah dilakukan oleh Rahmadewi dan Kurnia pada tahun 2016, penelitian tersebut menggunakan metode segmentasi sobel atau pendeteksian tepi pada foto *rontgen*. Metode ini membandingkan jumlah piksel putih pada deteksi tepi dengan jumlah piksel paru pada segmentasi, sehingga dapat diketahui klasifikasi penyakit paru dari foto *rontgen* tersebut. Selain menggunakan segmentasi sobel, penelitian yang lain, adalah penelitian yang dilakukan oleh Rao dan Pereira pada tahun 2016, penelitian tersebut menggunakan *Convolutional Neural Network (CNN)* dengan *CanNet* sebagai arsitekturnya untuk mengklasifikasikan penyakit kanker paru-paru melalui *CT Scan*.

Berdasarkan permasalahan yang telah dipaparkan di atas, maka penulis melakukan penelitian tentang pengimplementasian *deep learning* untuk mengklasifikasikan penyakit paru dengan menggunakan metode CNN, peneliti ingin mengetahui hasil dari implementasi tersebut dan mencari tingkat akurasi terbaik dengan melakukan beberapa perubahan terhadap jumlah epoch dan scenario datanya. *Deep Learning* adalah salah satu bagian dari metode *Machine Learning* yang mencoba untuk belajar dalam berbagai tingkatan, sesuai dengan tingkat abstraksi yang berbeda, *Deep Learning* menggunakan aliran yang berlapis-lapis dari unit pemrosesan yang non linear, setiap lapisan berturut-turut menggunakan output dari lapisan sebelumnya sebagai input (Deng & Yu, 2014). Sedangkan CNN adalah salah satu metode dari *Deep Learning* yang terdiri dari beberapa modul penting yaitu *convolutional layer* dan *pooling layer*. Modul-modul tersebut disusun antara satu dengan yang lain hingga membentuk arsitektur jaringan dan model yang mendalam. *Convolutional layer* terdiri dari banyak *weights* dan *pooling layer* mengambil sampel *output* dari *convolutional layer* untuk mengurangi laju data dari lapisan sebelumnya (Deng & Yu, 2014).

CNN dipilih karena dari penelitian sebelumnya yakni segmentasi sobel hanya menggunakan satu filter yaitu sobel, sedangkan CNN dalam *convolutional layer* akan membentuk filter menggunakan *gaussian random* dengan jumlah yang telah ditentukan oleh peneliti untuk melakukan proses *feature learning*. Sehingga diharapkan dengan menggunakan *filter* lebih dari satu, akurasi yang dihasilkan oleh CNN akan lebih baik. Semakin banyak filter, pada umumnya akan menambah informasi yang diekstrak, sehingga akan mempengaruhi akurasinya (Thoma, 2017). Pada penelitian ini penulis akan mengklasifikasikan paru-paru kedalam tiga kelas, yaitu : *Tuberculosis, Pneumonia* (Radang paru), dan paru-paru normal. Penulis berharap dengan adanya sistem ini dapat membantu meningkatkan hasil Analisa dokter spesialis paru dalam mendiagnosa penyakit paru, dan diharapkan akan memberikan kontribusi dalam penelitian dibidang *Computer Vision* dan *Deep Learning*.

1.2. Rumusan Masalah

Dari uraian latar belakang diatas dapat diidentifikasi masalah-masalah sebagai berikut:

1. Bagaimana implementasi *Deep Learning* pada Sistem Klasifikasi Penyakit Paru Berdasarkan Foto Rontgen Menggunakan Metode *Convolutional Neural Network* (CNN) dapat mengklasifikasikan penyakit paru?
2. Bagaimana hasil implementasi *Deep Learning* pada Sistem Klasifikasi Penyakit Paru Berdasarkan Foto Rontgen Menggunakan Metode *Convolutional Neural Network* (CNN)?

1.3. Tujuan Penelitian

Tujuan dari penelitian ini adalah untuk mengklasifikasi penyakit paru melalui foto rontgen paru-paru menggunakan metode *Convolutional Neural Network*.

1.4. Batasan Masalah

Terdapat beberapa batasan masalah yang diangkat sebagai parameter dalam melakukan penelitian ini, antara lain:

1. Penelitian ini mengimplementasikan *Deep Learning* pada Sistem berbasis *Web*.
2. Data yang digunakan, menggunakan Citra digital foto *rontgen* dari url berikut :
 - a. <https://nihcc.app.box.com/v/ChestXray-NIHCC>
 - b. http://openi.nlm.nih.gov/imgs/collections/ChinaSet_AllFiles.zip
3. Citra digital hasil foto *rontgen* akan dibagi kedalam tiga kelas yaitu : *Tuberkulosis*, *Pneumonia* (Radang paru), dan paru-paru normal.
4. Jumlah data yang digunakan 300 citra digital foto *rontgen*, dengan rincian sebagai berikut :
 - a. 100 citra digital hasil foto *rontgen* paru-paru normal.
 - b. 100 citra digital hasil foto *rontgen Pneumonia*.
 - c. 100 citra digital hasil foto *rontgen Tuberkulosis*.

1.5. Sistematika Penulisan

Adapun sistematika penulisan dijelaskan sebagai berikut:

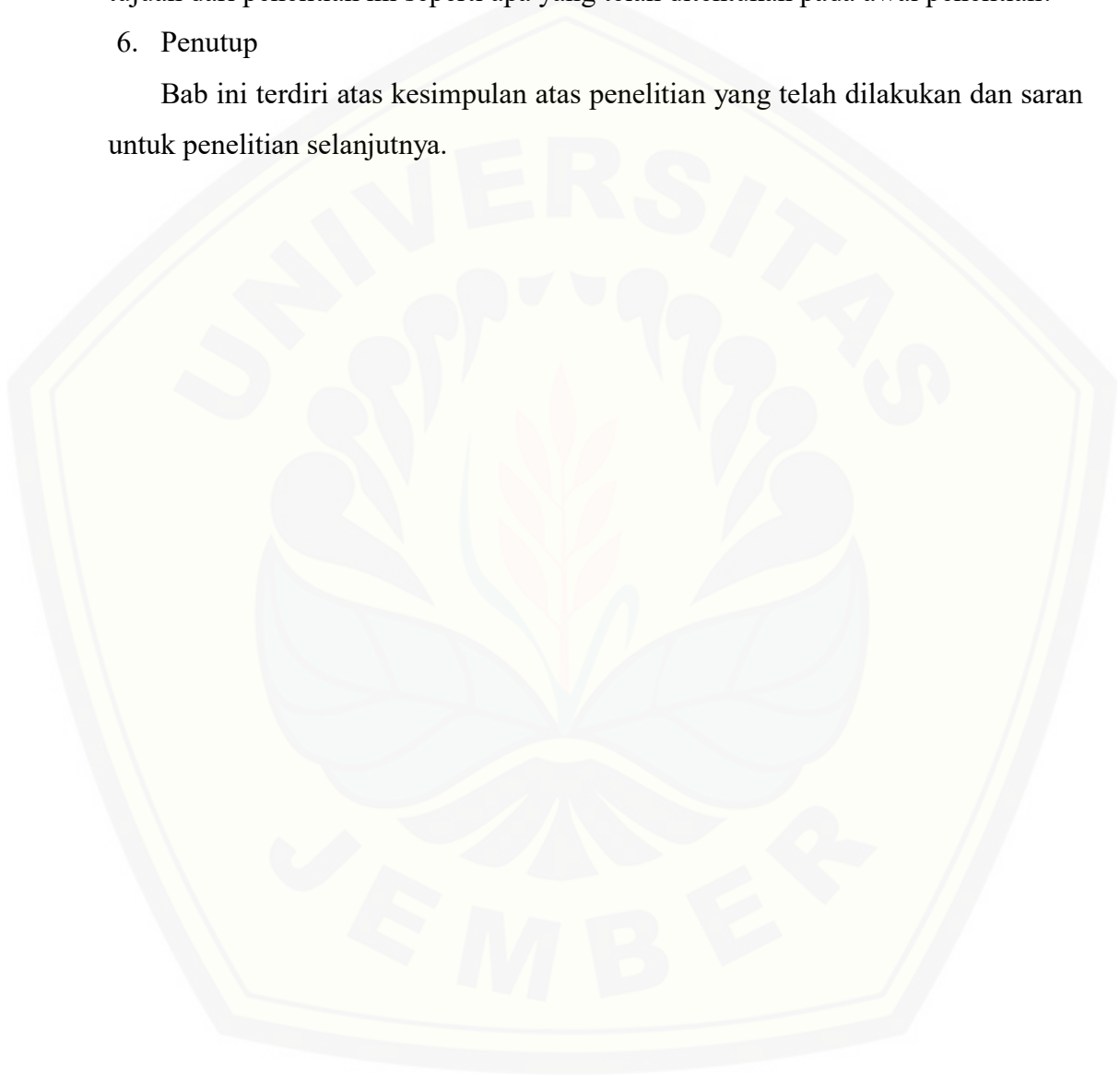
1. Pendahuluan
Bab ini memuat uraian tentang latar belakang, rumusan masalah, tujuan, batasan masalah, dan sistematika penulisan terkait penelitian yang dilakukan.
2. Tinjauan Pustaka
Bab ini menjelaskan tentang kajian-kajian teori terkait dengan penelitian yang dapat mendukung penelitian yang dilakukan.
3. Metodologi Penelitian
Bab ini menjelaskan tentang jenis penelitian, pengembangan sistem dan pengujian metode metode yang digunakan
4. Desain dan Implementasi
Bab ini menjelaskan proses perancangan sistem dimulai dari analisis kebutuhan fungsional dan non-fungsional sistem, pembuatan desain sistem, implementasi sistem dan pengujian.

5. Hasil dan Pembahasan

Bab ini menjelaskan mengenai hasil penelitian yang telah dilakukan serta pembahasan sistem yang telah dibuat. Pembahasan dilakukan guna menjelaskan dan memaparkan bagaimana penelitian ini menjawab perumusan masalah serta tujuan dari penelitian ini seperti apa yang telah ditentukan pada awal penelitian.

6. Penutup

Bab ini terdiri atas kesimpulan atas penelitian yang telah dilakukan dan saran untuk penelitian selanjutnya.



BAB 2. TINJAUAN PUSTAKA

2.1. Penelitian Terdahulu

Penelitian terdahulu tentang *computer vision* dan *convolutional neural network (CNN)* untuk melakukan klasifikasi terhadap gambar telah banyak dilakukan, seperti yang ditunjukkan pada tabel 2.1

Tabel 2.1 Penelitian terdahulu

No.	Peneliti	Tahun	Judul	Penjelasan
1.	1. LeCun 2. Bengio	1995	Convolutional networks for images, speech, and time-series	Penelitian ini menggunakan CNN untuk melakukan klasifikasi pada gambar <i>handwriting</i> , dan <i>speech recognition</i> untuk mendapatkan pola yang terbentuk.
2.	1. Rao 2. Pereira	2016	Convolutional neural networks for lung cancer screening in computed tomography (CT) scans	penelitian ini menggunakan <i>Convolutional Neural Network (CNN)</i> dengan <i>CanNet</i> sebagai arsitekturnya untuk mengklasifikasikan penyakit kanker paru-paru melalui <i>CT Scan</i> .
3.	1. Rahmadewi 2. Kurnia	2016	Klasifikasi Penyakit Paru Berdasarkan Citra Rontgen	penelitian ini menggunakan metode segmentasi sobel atau

			Dengan Metoda Segmentasi Sobel	pendeteksian tepi. Metode ini membandingkan jumlah piksel putih pada deteksi tepi dengan jumlah piksel paru pada segmentasi, sehingga dapat diketahui klasifikasi penyakit paru dari foto <i>rontgen</i> tersebut.
4.	1. Triano Nurhikmat 2. Tuti Purwaningsih	2018	Implementasi Deep Learning untuk Image Classification Menggunakan Algoritma Convolutional Neural Network (CNN) Pada Citra Wayang Golek	Penelitian ini menggunakan metode CNN dengan jumlah 2 <i>convolution layer</i> untuk melakukan klasifikasi pada gambar wayang golek yang didapat dari melakukan teknik <i>scrapping</i> terhadap <i>website google images</i> .

Dari Tabel 2.1 diketahui bahwa, CNN dapat digunakan untuk mengklasifikasikan gambar. Perbedaan yang ada antara penelitian terdahulu seperti pada Tabel 2.1 dengan penelitian ini adalah arsitektur CNN dan data yang digunakan. Arsitektur CNN yang digunakan dalam penelitian ini, menggunakan 3 *convolutional layer* dan data yang digunakan adalah citra digital hasil foto *rontgen*.

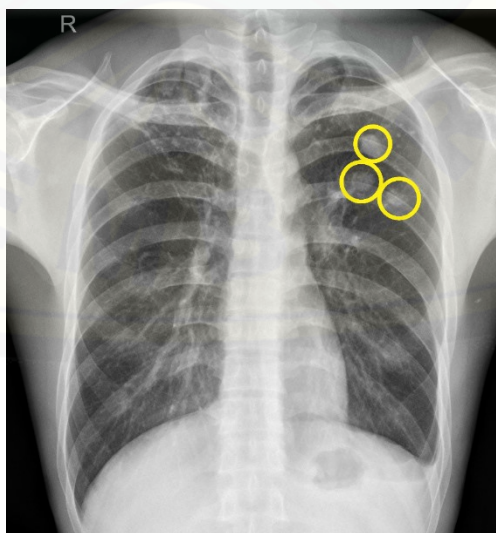
2.2. Penyakit Paru

Penyakit paru ada banyak macamnya, diantaranya yang umum dan banyak terjadi di Indonesia antara lain :

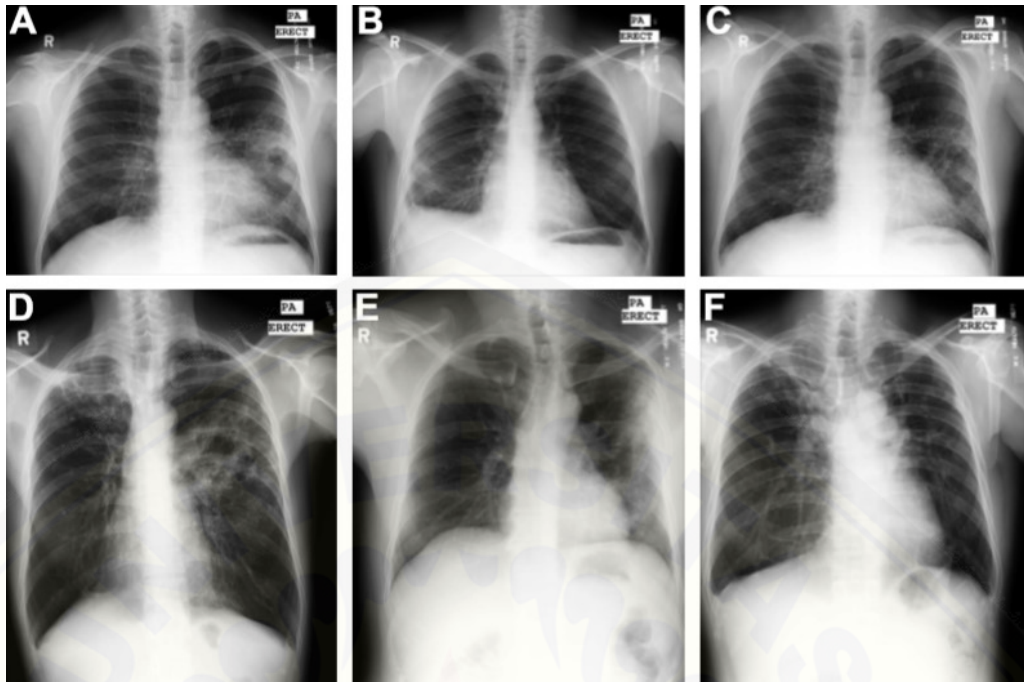
1) Tuberkulosis

Tuberkulosis (Tuberculosis, disingkat Tbc), atau Tb (singkatan dari "Tubercle bacillus") merupakan penyakit menular yang umum, dan dalam banyak kasus bersifat mematikan. Penyakit ini disebabkan oleh berbagai strain mikobakteria, umumnya *Mycobacterium tuberculosis* (disingkat "MTb" atau "MTbc"). Tuberkulosis biasanya menyerang paru-paru, namun juga bisa berdampak pada bagian tubuh lainnya. Tuberkulosis menyebar melalui udara ketika seseorang dengan infeksi TB aktif batuk, bersin, atau menyebarkan butiran ludah mereka melalui udara. Infeksi TB umumnya bersifat asimtomatik dan laten. Namun hanya satu dari sepuluh kasus infeksi laten yang berkembang menjadi penyakit aktif. Bila Tuberkulosis tidak diobati maka lebih dari 50% orang yang terinfeksi bisa meninggal (Muchtar, Herman, & Yulistini, 2018).

Foto *rontgen* pasien dengan penyakit Tuberkulosis dapat diidentifikasi dengan cara mencari adanya *coin lesion* atau luka yang berbentuk bulat akibat adanya infeksi bakteri, ukuran diameter dari *coin lesion* berkisar antara 1 sampai 5cm (Smithuis, 2014). *Coin lesion* pada foto *rontgen* paru ditandai dengan area yang berwarna keabu-abuan dan berbentuk bulat seperti yang ditunjukkan pada gambar 2.1



Gambar 2.1 Foto *rontgen* penyakit tuberculosi
(<http://openi.nlm.nih.gov>)



Gambar 2.2 abnormal CXR dari MC dataset

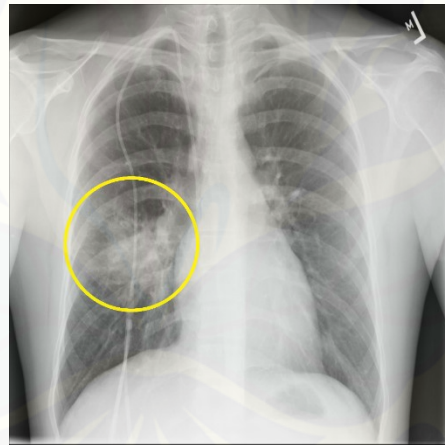
Gambar 2.2 merupakan contoh-contoh foto *rontgen* abnormal dalam dataset MC. Foto *rontgen* A memiliki *infiltrat kavitas* di sebelah kiri dan *infiltrat* halus di paru kanan bawah. Foto *rontgen* B adalah contoh TB *pleura*. Perhatikan bahwa sudut *costophrenic* kanan tumpul menunjukkan *efusi* sedang. Foto *rontgen* C memiliki *infiltrat* di kedua paru-paru. Foto *rontgen* D menunjukkan *infiltrat* yang tidak teratur di paru-paru kiri dengan *kavitasi* dan jaringan parut pada *apex* kanan. Foto *rontgen* E menunjukkan *infiltrat perifer* di paru-paru kiri. Foto *rontgen* F menunjukkan tanda-tanda TB, yang ditunjukkan oleh *retraksi hila bilateral superior*, yang lebih jelas di sebelah kanan (Jaeger, dkk., 2013).

2) Pneumonia

Radang paru-paru atau Pneumonia adalah penyakit infeksi yang menyerang paru, sehingga menyebabkan kantung udara di dalam paru meradang dan membengkak. Kondisi kesehatan ini sering kali disebut dengan paru-paru basah, sebab paru bisa saja dipenuhi dengan air atau cairan lendir (Efni, Machmud, & Pertiwi, 2016). Kondisi paru-paru basah ini dapat dialami oleh siapa pun. Gejala khasnya meliputi batuk, nyeri dada, demam, dan kesulitan bernapas. Alat diagnostik mencakup rontgen dan pengambilan kultur dari

sputum. Vaksin untuk mencegah jenis pneumonia tertentu kini sudah tersedia. Pengobatan yang dilakukan bergantung pada penyebab dasarnya. Dugaan *pneumonia* bakterial diobati dengan antibiotik. Jika pneumonianya parah, penderita biasanya dirujuk ke rumah sakit.

Foto *rontgen* pasien dengan penyakit *Pneumonia* dapat diidentifikasi dengan cara mencari adanya *consolidation*. *Consolidation* adalah hasil penggantian udara dalam alveoli oleh transudat, nanah, darah, sel atau zat lainnya. Penyakit ini biasanya dimulai di dalam alveoli dan menyebar dari satu alveolus ke alveolus lainnya (Smithuis, 2014). *Consolidation* pada foto *rontgen* paru ditandai dengan area yang berwarna abu-abu buram, hal ini terjadi karena isi dari area tersebut tidak hanya udara, melainkan juga terdapat cairan seperti yang ditunjukkan pada gambar 2.3



Gambar 2.3 Foto *rotngen* penyakit *pneumonia*
(<https://nihcc.app.box.com>)

2.3. Foto Rontgen

Foto *rontgen* adalah prosedur pemeriksaan dengan menggunakan radiasi gelombang elektromagnetik guna menampilkan gambaran bagian dalam tubuh. Gambaran dari benda padat seperti tulang atau besi ditampilkan sebagai area berwarna putih, sedangkan udara yang terdapat pada paru-paru akan tampak berwarna hitam, dan gambaran dari lemak atau otot ditampilkan dengan warna abu-

abu. Dalam beberapa jenis foto *rontgen*, digunakan tambahan zat pewarna (kontras) yang diminum atau disuntikkan, misalnya iodine atau barium, untuk menghasilkan gambaran yang lebih detail. Contoh foto rongen dapat dilihat pada Gambar 2.4



Gambar 2.4 Foto *Rontgen*
(<https://www.alodokter.com>,2018)

Prosedur foto *rontgen* ini dilakukan di rumah sakit oleh dokter atau petugas radiologi yang sudah terlatih. Meski radiasi memiliki risiko memicu pertumbuhan sel kanker, paparan radiasi dari foto Rontgen terbilang sangat kecil dan dianggap aman, terutama jika dibandingkan dengan manfaat yang didapat.

2.4. Computer Vision

Computer Vision adalah salah satu cabang ilmu di bidang komputer yang berfokus dan mempelajari tentang bagaimana mesin dapat mengekstraksi informasi yang ada di dalam sebuah gambar statis maupun dinamis (Forsyth & Ponce, 2012). Berbeda dengan cabang ilmu *Image Processing*, jika *image processing* mengolah citra digital atau gambar untuk memperbaiki atau mentransformasikan citra digital, sedangkan *computer vision* mengubah gambar menjadii model, sehingga mesin dapat mengenali gambar tersebut. *Computer vision* dapat dikatakan sebagai gabungan dari *image processing* dan *pattern recognition*.

Computer vision saat ini telah banyak dikombinasikan dengan banyak Teknik, tidak hanya *image processing* dan *pattern recognition* saja, *computer vision* juga sering dikombinasikan dengan metode-metode yang ada pada *machine learning* dan *deep learning*, sehingga computer tidak hanya dapat melihat, melainkan dapat belajar dari apa yang dilihat (Szeliski, 2010).

Terdapat banyak pendapat tentang faktor apa saja yang menyebabkan *computer vision* dapat berkerja dengan baik, setidaknya ada 3 faktor diantaranya adalah *computational theory, representations and algorithms, hardware implementation* (Szeliski, 2010).

2.5. *Deep Learning*

Deep Learning yang juga dikenal sebagai *deep structured learning* atau *hierarchical learning* merupakan bagian dari metode *Machine Learning*. Berdasarkan cara pembelajarannya dibagi menjadi 2 yaitu : *Supervised Learning* (missal : klasifikasi) dan *Unspervised Learning* (missal : analisis pola). Arsitektur *Deep Learning* banyak digunakan dalam bidang *Computer Vision, speech recognition, natural language processing, audio recognition, social network filtering, machine translation, bioinformatics*, dan lain-lain (LeCun, Bengio, & Hinton, 2015).

Perkembangan teknologi dan ilmu pengetahuan saat ini mendorong para peneliti untuk dapat menyelesaikan permasalahan yang ada, permasalahan yang ada diantaranya adalah sulitnya untuk mendapatkan informasi dan mengekstraksi fitur abstrak dari data mentah. *Deep learning* dapat melakukan ekstraksif fitur abstrak yang didapat dari data mentah dengan baik (Goodfellow, Bengio, & Courville, 2016)

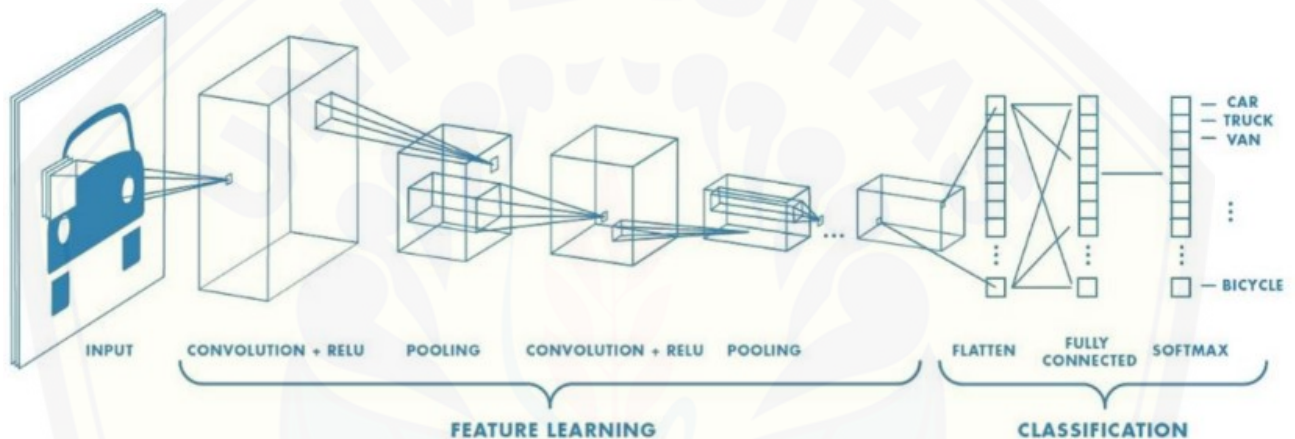
Deep Learning menggunakan aliran yang berlapis-lapis dari unit pemrosesan yang non linear, setiap lapisan berturut-turut menggunakan output dari lapisan sebelumnya sebagai input (Deng & Yu, 2014). Namun *Deep Learning* membutuhkan performa hardware yang sangat baik, dikarenakan banyaknya perhitungan yang harus di proses dan data yang besar.

2.6. *Convolutional Neural Network (CNN)*

Convolutional Neural Networks adalah jenis khusus dari jaringan saraf *multi-layer*. Hampir seperti setiap jaringan saraf lainnya, mereka dilatih dengan versi algoritma *back-propagation*. Perbedaannya adalah dalam arsitekturnya. *Convolutional Neural Networks* dirancang untuk mengenali pola visual langsung

dari piksel gambar dengan *preprocessing* minimal. Mereka dapat mengenali pola dengan variabilitas ekstrim (seperti karakter tulisan tangan), dan dengan kekokohan terhadap distorsi dan transformasi geometris sederhana (LeCun & Bengio, 1995).

CNN adalah sebuah arsitektur yang dapat dilatih dan terdiri dari beberapa tahapan. *Input* dan *output* dari setiap tahap adalah terdiri dari array yang biasa disebut *feature map*. Setiap tahap terdiri dari tiga layer yaitu konvolusi, fungsi aktivasi layer dan *pooling layer*. Jaringan arsitektur *Convolutional Neural Network* dapat dilihat pada Gambar 2.5



Gambar 2.5 Arsitektur *convolutional neural network*

Berdasarkan gambar 2.4, Tahap pertama pada arsitektur CNN adalah tahap konvolusi. Tahap ini dilakukan dengan menggunakan sebuah kernel dengan ukuran tertentu. Perhitungan jumlah kernel yang dipakai tergantung dari jumlah fitur yang dihasilkan. Kemudian dilanjutkan menuju fungsi aktivasi, biasanya menggunakan fungsi aktivasi ReLU (*Rectifier Linear Unit*), Selanjutnya setelah keluar dari proses fungsi aktivasi kemudian melalui proses *pooling*. Proses ini diulang beberapa kali sampai didapatkan peta fitur yang cukup untuk dilanjutkan ke *fully connected neural network*, dan dari *fully connected network* adalah *output class*. Penjelasan setiap tahapan *Convolutional Neural Network* :

a) Convolution Layer

Convolution layer merupakan bagian dari tahap pada arsitektur CNN. Tahap ini melakukan operasi konvolusi pada output dari layer sebelumnya. Layer tersebut adalah proses utama yang mendasari jaringan arsitektur CNN. Konvolusi adalah istilah matematis dimana pengaplikasian sebuah fungsi pada output fungsi lain secara berulang. Operasi konvolusi merupakan operasi pada dua fungsi argumen bernilai nyata. Operasi ini menerapkan fungsi output sebagai *Feature Map* dari input citra. Input dan output ini dapat dilihat sebagai dua argumen bernilai riil. Operasi konvolusi dapat dituliskan seperti persamaan 2.1

$$s(t) = (x * t)(t) = \sum_{\alpha=-\infty}^{\infty} x(\alpha) * w(t - \alpha) \quad (2.1)$$

Keterangan :

S(t) = Fungsi hasil operasi konvolusi

X = Input

W = bobot (kernel)

Fungsi s(t) memberikan output tunggal berupa *feature map*. Argumen pertama adalah input yang merupakan x dan argumen kedua w sebagai kernel atau filter. Apabila dilihat input sebagai citra dua dimensi, maka bisa dikatakan t sebagai piksel dan menggantinya dengan i dan j. Maka dari itu, operasi untuk konvolusi ke input dengan lebih dari satu dimensi dapat ditulis seperti pada persamaan 2.2 dan 2.3

$$s(i, j) = (K * I)(i, j) = \sum_{\infty} \sum_{\infty} I(i - m, j - n)K(m, n) \quad (2.2)$$

$$s(i, j) = (K * I)(i, j) = \sum_{\infty} \sum_{\infty} I(i - m, j + n)K(m, n) \quad (2.3)$$

Keterangan :

$s(i,j)$ = Fungsi hasil operasi konvolusi

I = Input

K = Kernel

i,j = pixel dari Input

m,n = pixel dari Kernel

Berdasarkan persamaan 2.2 dan 2.3 merupakan perhitungan dasar dalam operasi konvolusi, dengan i dan j adalah sebuah piksel dari citra. Perhitungan tersebut bersifat komulatif dan muncul saat K sebagai kernel, kemudian I sebagai input dan kernel yang dapat dibalik relatif terhadap input. Sebagai alternatif operasi konvolusi dapat dilihat sebagai perkalian perkalian matriks antara citra masukan dan kernel dimana keluarannya dihitung dengan *dot product*. Selain itu, penentuan volume output juga dapat ditentukan dari masing-masing lapisan dengan hyperparameters. Hyperparameter yang digunakan pada persamaan 4 digunakan untuk menghitung banyaknya neuron aktivasi dalam sekali output. Perhatikan persamaan 2.4

$$(W - F + 2P)/(S + 1) \quad (2.4)$$

Keterangan :

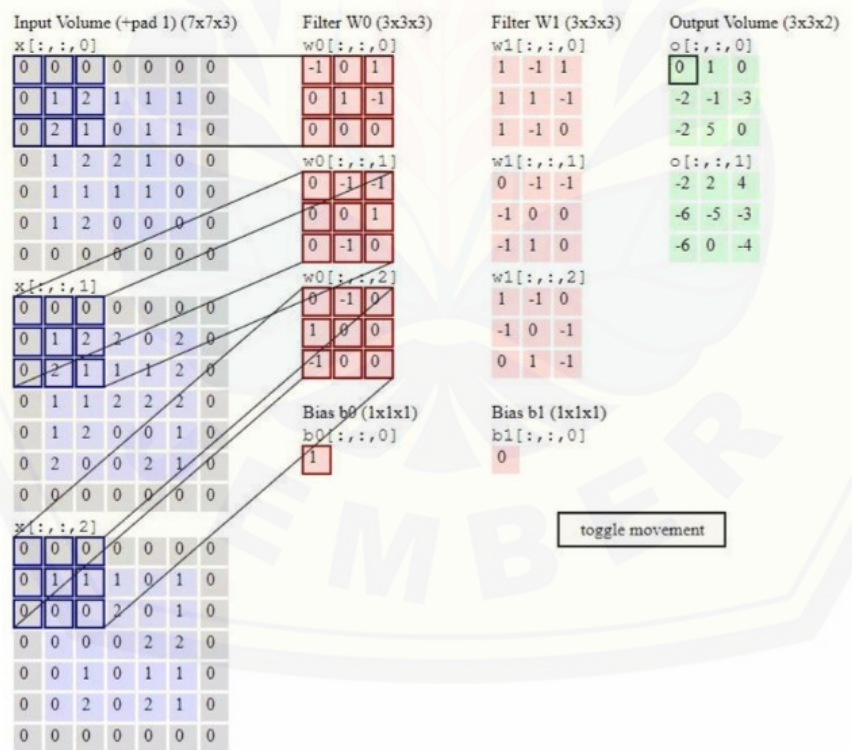
W = Ukuran volume gambar

F = Ukuran Filter

P = Nilai Padding yang digunakan

S = Ukuran Pergeseran (Stride)

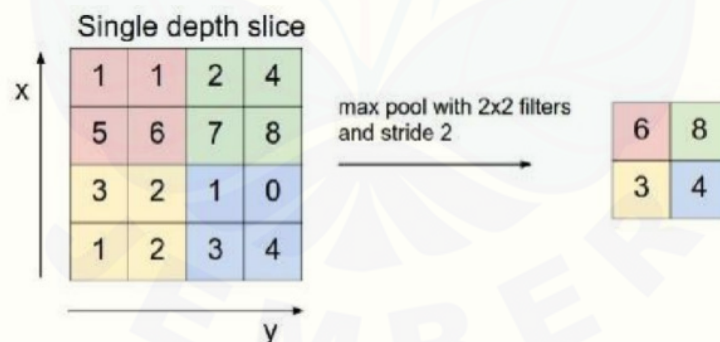
Berdasarkan persamaan 2.4, dapat dihitung ukuran spasial dari volume output dimana hyperparameter yang dipakai adalah ukuran volume (W), filter (F), Stride yang diterapkan (S) dan jumlah padding nol yang digunakan (P). Stride merupakan nilai yang digunakan untuk menggeser filter melalui input citra dan Zero Padding adalah nilai untuk mendapatkan angka nol di sekitar border citra. Berikut adalah operasi Convolutional Layer terdiri dari neuron yang tersusun sedemikian rupa sehingga membentuk sebuah filter dengan panjang dan tinggi (pixels). Sebagai contoh, layer pertama pada feature extraction layer biasanya adalah conv. Layers dengan ukuran 5x5x3. Panjang 5 pixels, tinggi 5 pixels dan tebal/jumlah 3 buah sesuai dengan channel dari image tersebut. Ketiga filter ini akan digeser keseluruhan bagian dari gambar. Setiap pergeseran akan dilakukan operasi “dot” antara input dan nilai dari filter tersebut sehingga menghasilkan sebuah output atau biasa disebut sebagai *activation map* atau *feature map* seperti yang ditunjukkan pada Gambar 2.6



Gambar 2.6 Convolution Layer
(Medium Samuel Sena, 2017)

b) Operasi Pooling

Pooling merupakan pengurangan ukuran matriks dengan menggunakan operasi *pooling*. *Pooling Layer* biasanya berada setelah *convolution*. Pada dasarnya *pooling layer* terdiri dari sebuah filter dengan ukuran dan stride tertentu yang akan secara bergantian bergeser pada seluruh area *feature map*. Dalam *pooling layer* terdapat dua macam *pooling* yang biasa digunakan yaitu *average pooling* dan *max pooling*. Nilai yang diambil pada *average pooling* adalah nilai rata-rata, sedangkan pada *max pooling* adalah nilai maksimal. Lapisan *pooling* yang dimasukkan diantara lapisan konvolusi secara berturut-turut dalam arsitektur model CNN dapat secara progresif mengurangi ukuran *volume output* pada *feature map*, sehingga mengurangi jumlah parameter dan perhitungan di jaringan, untuk mengendalikan *overfitting*. Lapisan *pooling* bekerja di setiap tumpukan *feature map* dan melakukan pengurangan pada ukurannya. Bentuk lapisan *pooling* umumnya dengan menggunakan filter dengan ukuran 2x2 yang diaplikasikan dengan langkah sebanyak dua dan beroperasi pada setiap irisan dari inputnya. Contoh gambar operasi *max pooling* dapat dilihat pada Gambar 2.7 :



Gambar 2.7 Operasi *Max Pooling*
(Medium Samuel Sena, 2017)

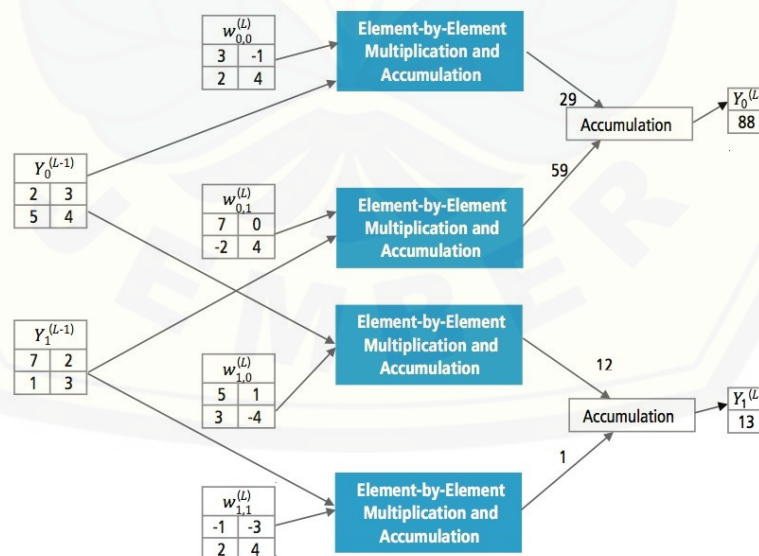
Berdasarkan gambar 2.6 menunjukkan proses dari *max pooling*. Output dari proses *pooling* adalah sebuah matriks dengan dimensi yang lebih kecil dibandingkan dengan citra awal. Lapisan *pooling* pada gambar 2.6 beroperasi pada setiap irisan kedalaman *volume input* secara bergantian. Jika dilihat dari gambar 2.6 operasi *max pooling* dengan menggunakan ukuran filter 2x2. Masukan pada proses tersebut

berukuran 4×4 , dari masing-masing 4 angka pada input operasi tersebut diambil nilai maksimalnya kemudian dilanjutkan membuat ukuran output baru menjadi ukuran 2×2 .

c) Fully-Connected Layer

Fully-Connected Layer adalah sebuah lapisan dimana semua neuron aktivasi dari lapisan sebelumnya terhubung semua dengan neuron di lapisan selanjutnya sama seperti halnya dengan neural network biasa. Pada dasarnya lapisan ini biasanya digunakan pada MLP (Multi Layer Perceptron) yang mempunyai tujuan untuk melakukan transformasi pada dimensi data agar data dapat diklasifikasikan secara linear. Perbedaan antara lapisan Fully-Connected dan lapisan konvolusi biasa adalah neuron di lapisan konvolusi terhubung hanya ke daerah tertentu pada input, sementara lapisan Fully-Connected memiliki neuron yang secara keseluruhan terhubung.

Namun, kedua lapisan tersebut masih mengoperasikan *dot product*, sehingga fungsinya tidak begitu berbeda. Proses fully-connected dapat dilihat pada Gambar 2.8



Gambar 2.8 Fully connected layer

d) Backpropagation

Backpropagation merupakan algoritma pembelajaran yang digunakan oleh *perceptron* dengan banyak lapisan untuk mengubah setiap bobot yang terhubung dengan *neuron* pada *hidden layers*. Penggunaan *error* di dalam *backpropagation* bertujuan untuk mengubah nilai setiap bobot dalam arah mundur (backward). Sebelum mendapatkan *error* ini, terdapat tahap awal yang harus dilakukan yaitu tahap perambatan maju (forward). Pelatihan *backpropagation* meliputi tiga tahap. Tahap pertama adalah tahap maju (forward). tahap ini menghitung maju tahap layer input sampai tahap layer output dengan menggunakan fungsi aktivasi yang telah ditentukan. Tahap kedua adalah tahap mundur (backward), pada tahap ini selisih antara output jaringan dengan target yang diinginkan merupakan kesalahan yang terjadi. Kesalahan tersebut dipropagasikan mundur, mulai dari garis yang terhubung langsung dengan setiap unit pada layer output. Kemudian tahap yang ketiga adalah tahap yang akan memodifikasi bobot untuk menurunkan tingkat kesalahan yang terjadi (Jumarwanto, Hartanto, & Prastiyanto, 2009). Berikut adalah langkah dari algoritma Backpropagation:

1) Tahap pertama (*Feed Forward*)

- Inisiasi nilai bobot dengan *gaussian random*. Rentang nilai yang dihasilkan adalah -1 hingga 1
- Lakukan perkalian *dot product* menggunakan persamaan 2.1 hingga mencapai *output layer*.

2) Tahap kedua (*Backward*)

- Hitung nilai *loss* menggunakan *cross entropy* dengan menggunakan persamaan 2.5

$$E = - \sum_i (target_i * \ln(prediction_i) + (1 - target_i) * \ln(1 - prediction_i)) \quad (2.5)$$

- Setelah mendapatkan nilai *loss* lakukan perubahan bobot

3) Perubahan bobot

- Perubahan bobot dilakukan dengan menggunakan persamaan 2.6

$$w(\text{baru}) = w(\text{lama}) + \text{learning_rate} * \text{Error} \quad (2.6)$$



BAB 3. METODOLOGI PENELITIAN

3.1. Tempat Penelitian

Penelitian ini dilakukan di Laboratorium Rekayasa Perangkat Lunak (RPL), Fakultas Ilmu Komputer, Universitas Jember.

3.2. Waktu Penelitian

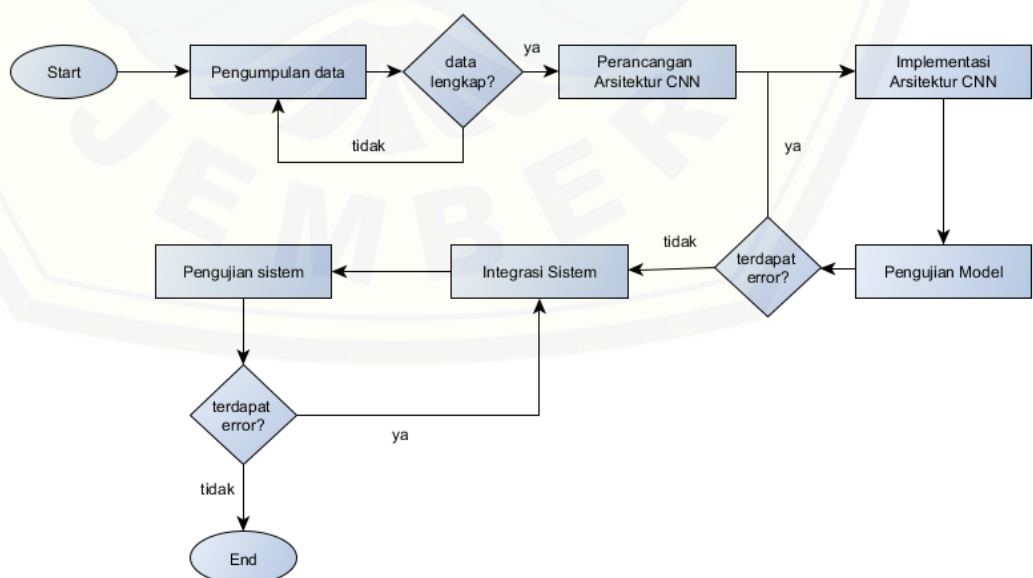
Waktu penelitian ini berlangsung sekitar 4 bulan, mulai bulan Maret 2019 sampai bulan Juli 2019.

3.3. Jenis Penelitian

Jenis penelitian yang akan dilakukan adalah penelitian terapan (*applied research*). Merancang, mendesain, dan membangun sebuah sistem untuk mengklasifikasikan penyakit paru berdasarkan foto *rontgen* dengan penerapan metode *Convolutional Neural Network (CNN)* berbasis *web*.

3.4. Tahapan Penelitian

Tahapan penelitian merupakan urutan langkah penelitian yang dilakukan. Gambaran tahapan penelitian dapat dilihat pada gambar 3.1



Gambar 3.1 Tahapan penelitian

Berdasarkan gambar 3.1, alur tahapan penelitian tersebut akan dijelaskan kedalam sub bab dibawah ini :

3.4.1. Studi Literatur

Studi literatur merupakan tahapan awal dalam penelitian ini, yang bertujuan untuk penyusunan dasar teori, dengan cara mengumpulkan dan membaca buku, jurnal, artikel ilmiah serta sumber lain yang mendukung penelitian ini.

3.4.2. Pengumpulan Data

Pengumpulan data adalah tahapan dimana peneliti mengumpulkan data-data yang dibutuhkan untuk penelitian. Penelitian ini menggunakan data yang diperoleh dari *National Institute of Health Clinical Center (NIHCC)* dan *Shenzhen Hospital*. NIHCC merupakan rumah sakit yang didedikasikan untuk penelitian klinis yang bertempat di Bethesda, Maryland, USA (Wang, dkk., 2017). Sedangkan *Shenzhen Hospital* adalah salah satu rumah sakit terbesar di Shenzhen, Guangdong, China (Jaeger, dkk., 2013). Data tersebut didapatkan dengan cara mengunduhnya di alamat *url* berikut :

d. <https://nihcc.app.box.com/v/ChestXray-NIHCC>

e. http://openi.nlm.nih.gov/imgs/collections/ChinaSet_AllFiles.zip

3.4.3. Perancangan Arsitektur CNN

Perancangan arsitektur CNN adalah tahap dimana peneliti merancang arsitektur CNN yang akan digunakan. Dalam penelitian ini arsitektur yang digunakan adalah CNN dengan 3 lapisan konvolusi.

3.4.4. Implementasi Arsitektur CNN

Implementasi arsitektur CNN adalah proses menuliskan arsitektur CNN yang telah dibuat kedalam program komputer. Peneliti disini menggunakan *Integrated Development Environment (IDE) PyCharm*, bahasa pemrograman *Python 3.6.5*.

3.4.5. Pengujian Model

Pengujian model adalah tahap dimana peneliti melakukan pengujian terhadap model dari arsitektur CNN yang telah dibangun, pengujian dilakukan dengan melakukan perubahan terhadap beberapa parameter agar dapat memberikan hasil yang terbaik dalam mengklasifikasikan foto *rontgen* paru, adapun parameter yang dimaksud adalah jumlah *epoch* dan skenario data. Dalam penelitian ini peneliti menggunakan 3 *epoch* yang berbeda, yaitu : *epoch* 17, *epoch* 30 dan *epoch* 50, sedangkan untuk skenario datanya, peneliti disini menggunakan 3 skenario data yang berbeda di setiap *epoch* yang digunakan, masing-masing skenario tersebut yaitu : skenario data 70:30, skenario data 80:20 dan skenario data 90:10. Pengujian model selanjutnya akan membandingkan akurasi dari masing-masing model yang nantinya akan dipilih model dengan akurasi tertinggi. Perhitungan untuk menghitung akurasi model yang sebenarnya adalah seperti yang ditunjukkan dengan persamaan 3.1

$$akurasi = \frac{\sum \text{prediksi benar}}{\sum \text{data}} \quad (3.1)$$

3.4.6. Integrasi Sistem

Integrasi sistem adalah tahap dimana peneliti melakukan integrasi antara model yang telah dibuat, dengan *web apps* untuk memudahkan pengguna dalam menggunakan sistem klasifikasi penyakit paru. Dalam penelitian ini, peneliti menggunakan Bahasa pemrograman *PHP* dengan *Laravel* sebagai *framework*. Manajemen basis data yang digunakan dalam pembuatan *web apps* adalah *MySQL*.

3.4.7. Pengujian Sistem

Pengujian sistem adalah tahap dimana peneliti melakukan pengujian terhadap sistem yang telah dibuat, yaitu dengan menggunakan pengujian *black box*. Pengujian *black box* adalah pengujian dengan cara melakukan pengujian terhadap sistem yang telah berjalan. Pengujian dilakukan dengan mencoba sistem dengan berbagai kemungkinan kesalahan yang ada.

BAB 4. DESAIN DAN IMPLEMENTASI

Bab ini akan membahas tentang perancangan dan penerapan metode *Convolutional Neural Network* (CNN) pada sistem klasifikasi penyakit paru berdasarkan foto *rontgen*. Proses perancangan sistem dimulai dari analisis kebutuhan fungsional dan non-fungsional sistem, pembuatan desain sistem, dan pengujian.

4.1. Analisa Kebutuhan Sistem

Analisa kebutuhan sistem dalam penelitian ini dilakukan dengan cara mengidentifikasi permasalahan dalam melakukan klasifikasi foto *rontgen* dengan metode CNN, kemudian dicatat dan dinalisa untuk mendapatkan kebutuhan fungsional dan kebutuhan non fungsional.

Kebutuhan fungsional sistem pada penelitian ini adalah sebagai berikut :

1. Sistem dapat menyimpan dan mengolah foto *rontgen* yang telah dimasukkan
2. Sistem dapat melakukan *training data* dari dataset yang telah dimasukkan
3. Sistem dapat menyimpan model yang telah dihasilkan dari proses training
4. Sistem dapat melakukan *testing* terhadap model yang telah dibuat.
5. Sistem dapat menampilkan hasil klasifikasi dari foto *rontgen* yang telah diproses menggunakan metode *Convolutional Neural Network* (CNN).

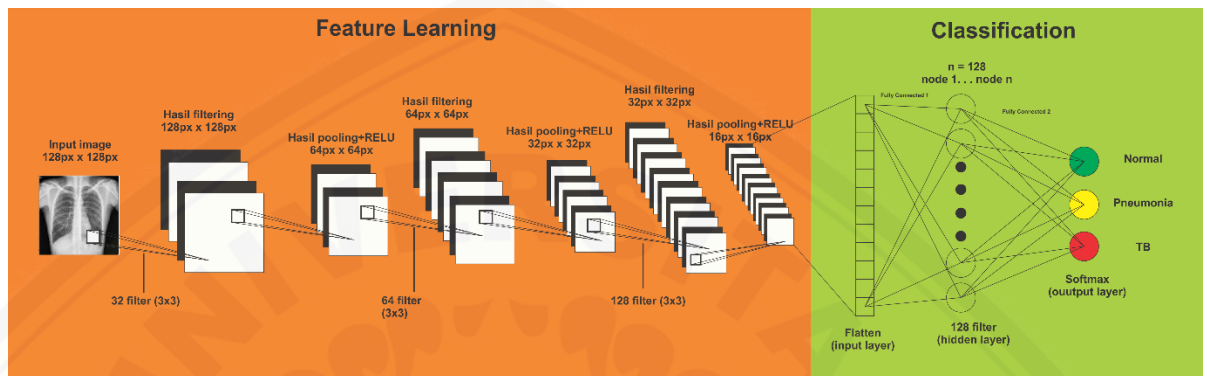
Sedangkan untuk kebutuhan non fungsional sistem pada penelitian adalah tampilan yang mudah digunakan agar pengguna tidak kesulitan dalam menggunakan sistem tersebut.

4.2. Desain Sistem

Desain sistem yang dibuat meliputi desain arsitektur CNN, *flowchart*, *Business Process Model Notation* (BPMN), *class diagram* dan *Entity Relationship Diagram* (ERD).

4.2.1. Desain Arsitektur CNN

Arsitektur CNN dalam penelitian ini menggunakan tiga lapisan *convolution* dengan masing-masing terdiri dari 32 filter, 64 filter dan 128 filter, seperti yang digambarkan pada gambar 4.1

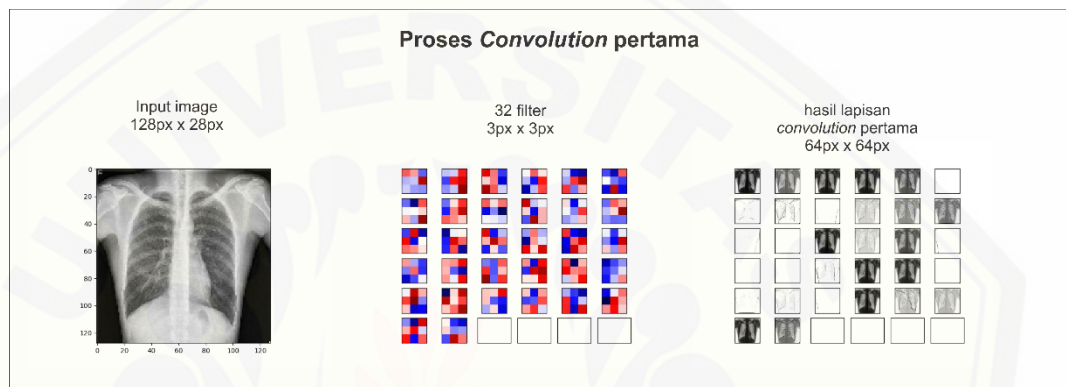


Gambar 4.1 Desain Arsitektur CNN

Gambar 4.1 merupakan arsitektur CNN yang digunakan dalam proses training untuk menghasilkan model yang optimal. Penjelasan gambar 4.1 :

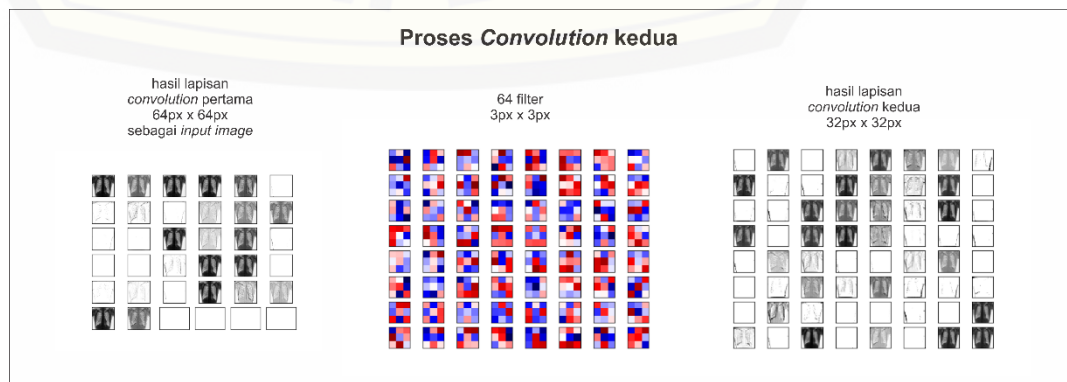
1. *Input image* yang digunakan adalah foto *rontgen* yang telah dilakukan penyesuaian dengan kriteria sebagai berikut :
 - a) Ukuran 128px x 128px.
 - b) Setiap pixelnya diubah kedalam bentuk desimal (dikalikan 1/255).
 - c) 3 *channels Red, Green, Blue* (RGB).
2. Proses pada lapisan *convolution* pertama meliputi *filtering*, *pooling* dan menggunakan fungsi aktivasi *Rectified Linear Unit* (ReLU). Dalam proses *filtering* ini menggunakan *filter* atau *kernel* dengan ukuran 3px x 3px sebanyak 32 filter. *Stridenya* sebesar 1 sehingga pergeseran dari *filter* tersebut adalah 1px secara *horizontal* maupun *vertical*. *Paddingnya* adalah 0 sehingga *output* dari proses *filtering* ini adalah sebuah gambar yang berukuran sama seperti *input imagenya* yaitu 128px x 128px. Selanjutnya adalah proses *pooling*, dalam proses ini peneliti menggunakan *kernel pooling* dengan ukuran 2px x 2px. *Stridenya* sebesar 2 sehingga pergeseran dari *pooling* tersebut adalah 2px secara *horizontal*

maupung *vertical*. *Pooling* ini bertugas untuk mereduksi ukuran *input* dengan cara mengambil nilai *pixel* terbesar berdasarkan pergeseran *kernel*nya, sehingga menghasilkan matriks baru dengan ukuran yang lebih kecil dari *input* yaitu 64px x 64px. Selanjutnya adalah fungsi aktivasi ReLU, fungsi ini mengubah nilai *pixel* negative menjadi 0. Proses yang terjadi dalam lapisan *convolution* yang pertama digambarkan pada gambar 4.2



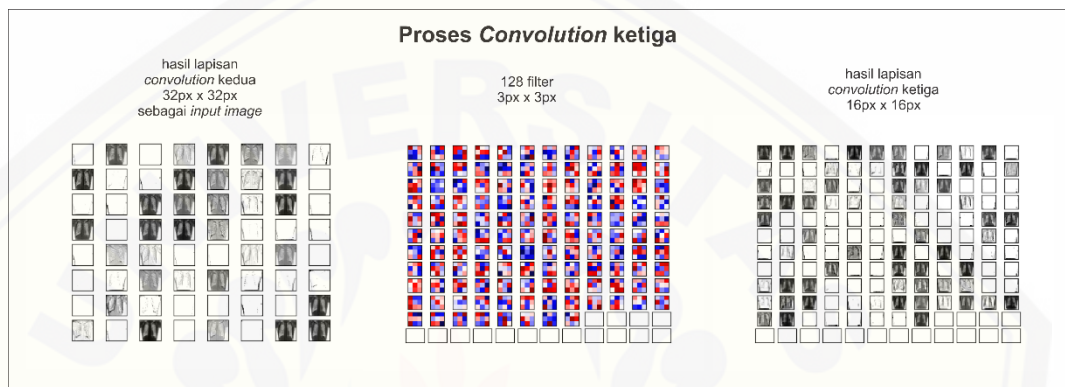
Gambar 4.2 Proses *convolution* pertama

3. Proses pada lapisan *convolution* kedua hampir sama dengan proses *convolution* pertama, namun yang membedakan adalah *input* dan jumlah filter yang digunakan. *Input* yang digunakan adalah hasil dari proses lapisan *convolution* pertama sedangkan jumlah filter yang digunakan dalam proses *convolution* kedua ini adalah sebanyak 64 filter. *Output* yang dihasilkan adalah *image* dengan ukuran 32px x 32px. Proses yang terjadi dalam lapisan *convolution* yang kedua digambarkan pada gambar 4.3



Gambar 4.3 Proses *convolution* kedua

- Proses pada lapisan *convolution* ketiga, *Input* yang digunakan adalah hasil dari proses lapisan *convolution* kedua sedangkan jumlah filter yang digunakan dalam proses *convolution* ketiga ini adalah sebanyak 128 filter. *Output* yang dihasilkan adalah *image* dengan ukuran 16px x 16px. Proses yang terjadi dalam lapisan *convolution* yang kedua digambarkan pada gambar 4.4

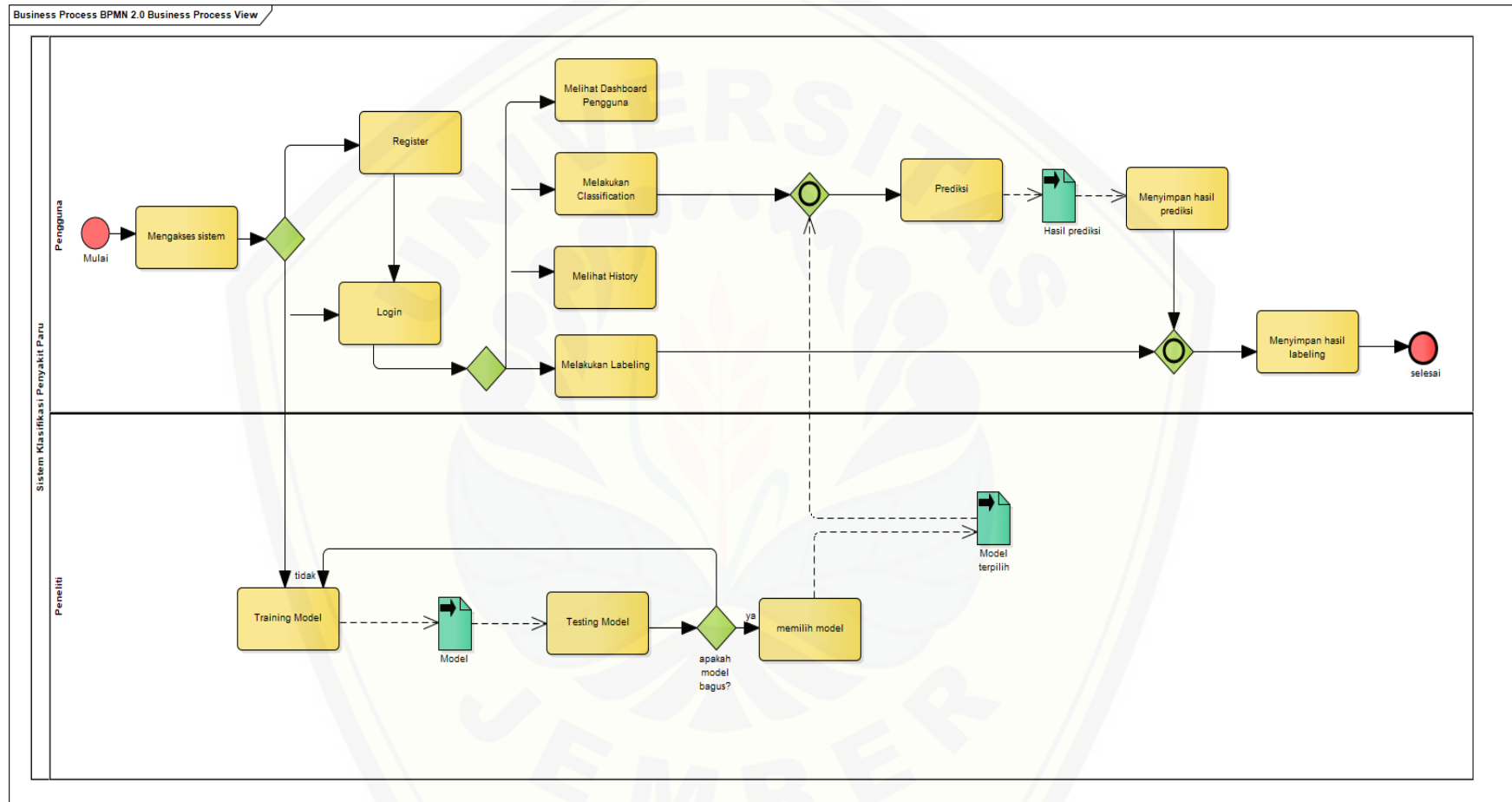


Gambar 4.4 Proses *convolution* ketiga

- Proses selanjutnya adalah *flatten*, proses ini mengubah *pixel* hasil dari proses lapisan *convolution* ketiga menjadi sebuah vector atau matriks 1 dimensi.
- Proses selanjutnya adalah membuat *fully connected layer* dengan input dari *flatten layer* dengan output sebanyak 128.
- Proses terakhir adalah *fully connected layer* kedua dengan input dari layer sebelumnya, menggunakan fungsi aktivasi *softmax*.

4.2.2. Business Process Model Notation (BPMN)

Business Process Model Notation (BPMN) merupakan representasi grafis untuk menentukan proses bisnis dari sebuah model proses bisnis, BPMN dibuat agar memudahkan pembaca dalam mengetahui proses bisnis yang terjadi. Berikut adalah BPMN dari sistem klasifikasi penyakit paru dalam penelitian ini, yang dapat dilihat pada gambar 4.5



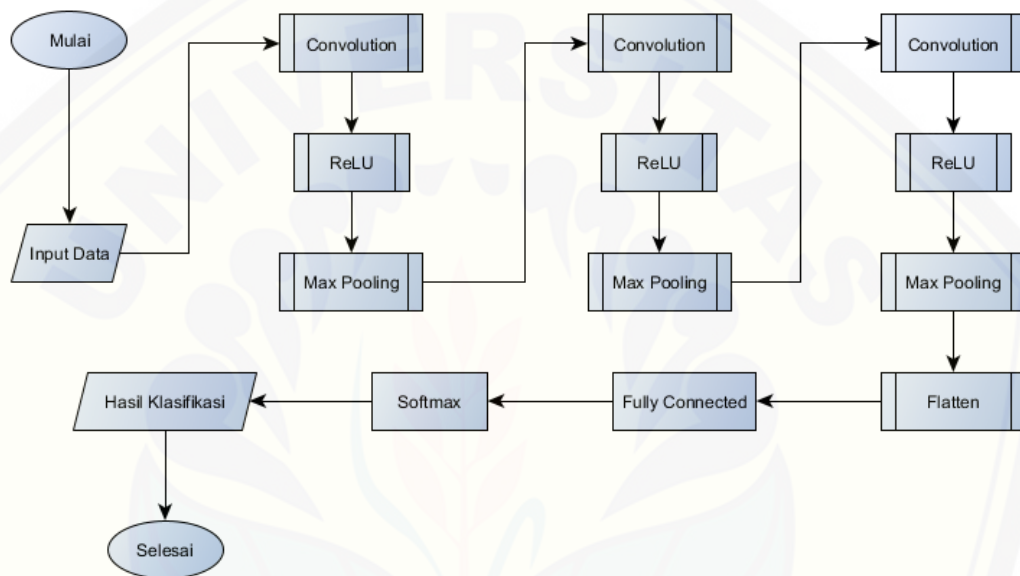
Gambar 4.5 Business Process Model Notation

4.2.3. Flowchart

Flowchart merupakan suatu bagan yang digambarkan dengan simbol-simbol tertentu yang bertujuan untuk menggambarkan sebuah alur proses dari sebuah algoritma.

1. *Flowchart* arsitektur CNN

Flowchart arsitektur CNN menggambarkan alur proses algoritma CNN secara global, *Flowchart* arsitektur CNN dapat dilihat pada gambar 4.6

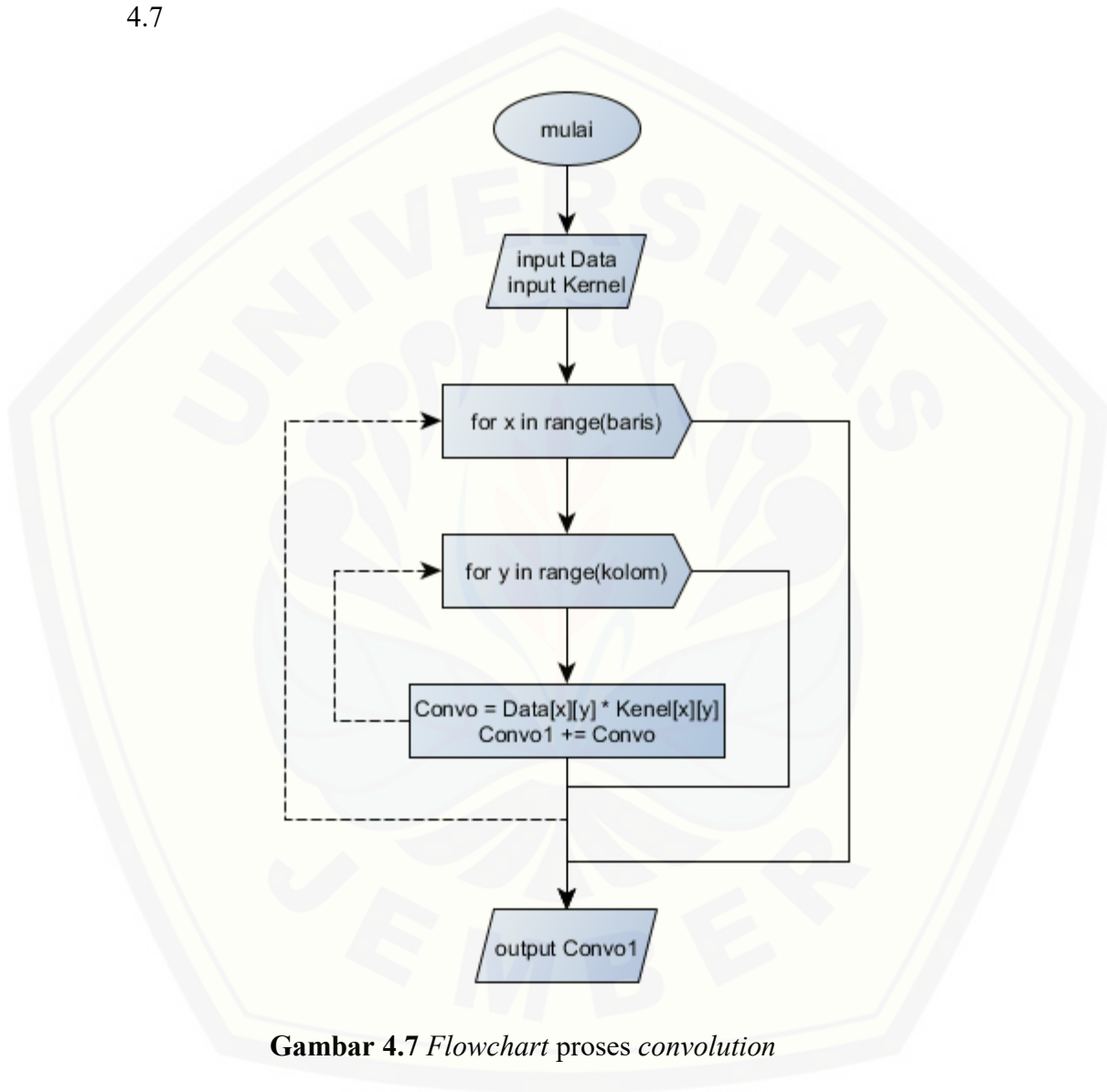


Gambar 4.6 *Flowchart* arsitektur CNN

Berdasarkan gambar 4.6 terdapat proses utama dalam CNN yaitu *Convolution*, *ReLU*, *Max Pooling* yang nantinya akan digambarkan dalam *flowchart* proses yang lebih detail.

2. Flowchart Proses Convolution

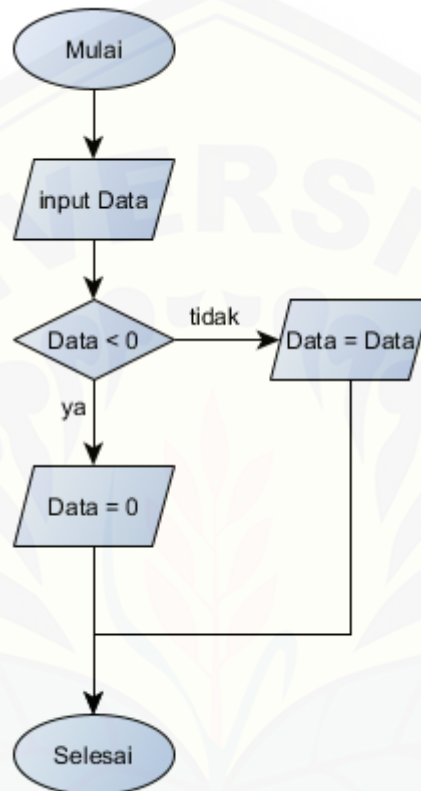
Flowchart Proses Convolution menggambarkan, proses *convolution* yang lebih detail, inti dari proses ini adalah melakukan perkalian *dot matrix* antara data input (pixel) dengan *kernel* yang outputnya berupa matriks baru yang lebih kecil. *Flowchart Proses Convolution* dapat dilihat pada gambar 4.7



Gambar 4.7 *Flowchart proses convolution*

3. Flowchart Proses ReLU

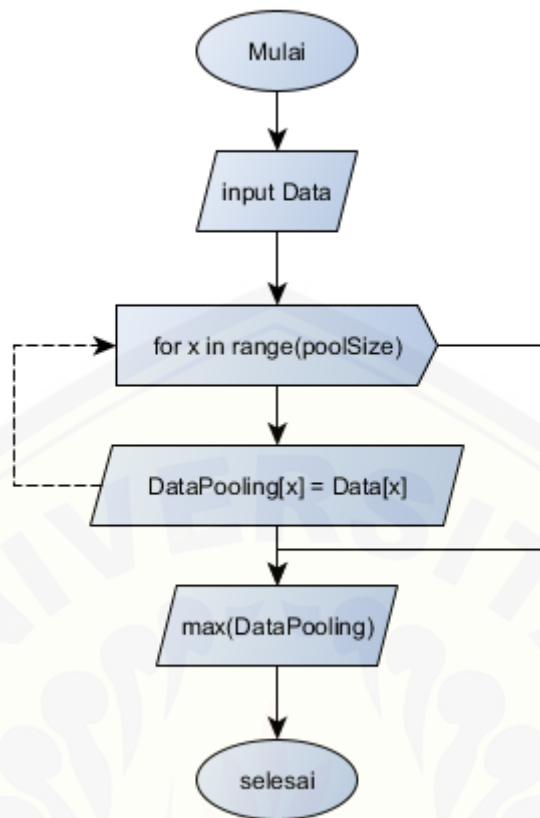
Flowchart Proses ReLU menggambarkan proses *Rectified Linear Unit* (ReLU) yang merupakan *activation function* yang bertugas untuk mengubah nilai negative menjadi 0. Flowchart Proses ReLU dapat dilihat pada gambar 4.8



Gambar 4.8 Flowchart proses ReLU

4. Flowchart Proses Max Pooling

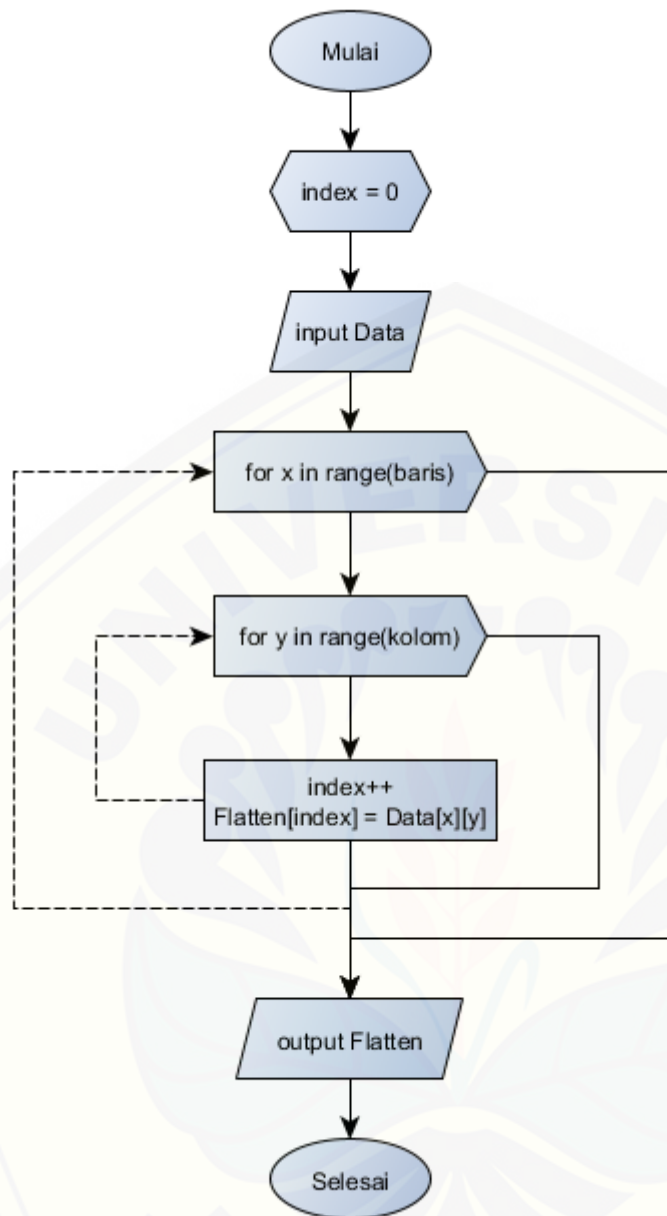
Flowchart Proses *Max Pooling* menggambarkan proses untuk mengambil nilai terbesar dari suatu matriks. Pooling bertujuan untuk melakukan pengurangan ukuran matriks dengan menggunakan operasi pooling. Operasi *pooling* yang digunakan dalam penelitian ini adalah operasi *max pooling*. Flowchart Proses *Max Pooling* dapat dilihat pada gambar 4.9



Gambar 4.9 Flowchart proses Max Pooling

5. Flowchart Proses Flatten

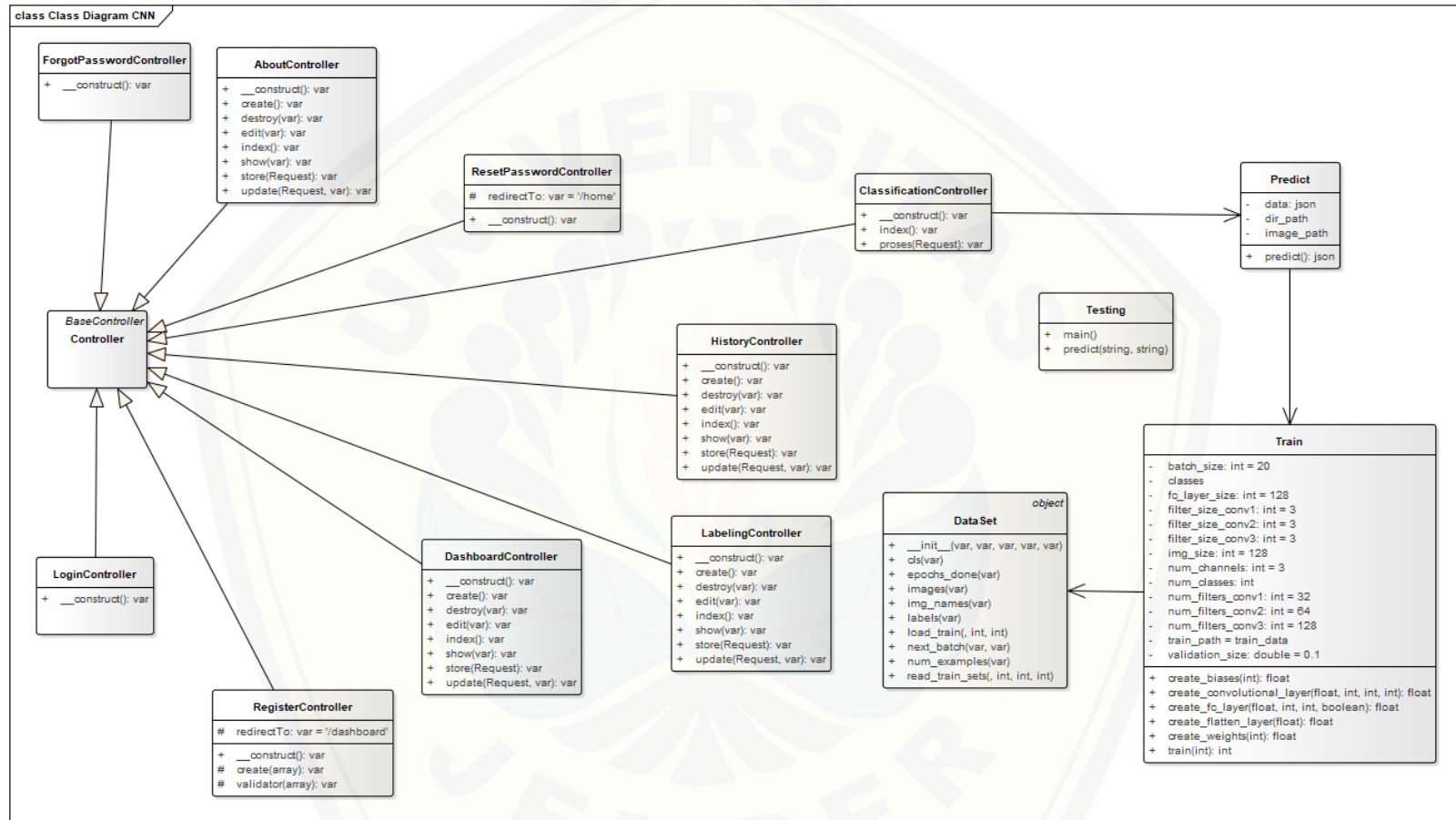
Flowchart Proses Flatten menggambarkan proses *flatten* yang lebih detail, proses *flatten* adalah proses untuk mengubah suatu matriks ke dalam bentuk *vector* atau kedalam bentuk 1 dimensi. Flowchart Proses Flatten dapat dilihat pada gambar 4.10



Gambar 4.10 Flowchart proses flatten

4.2.4. Class Diagram

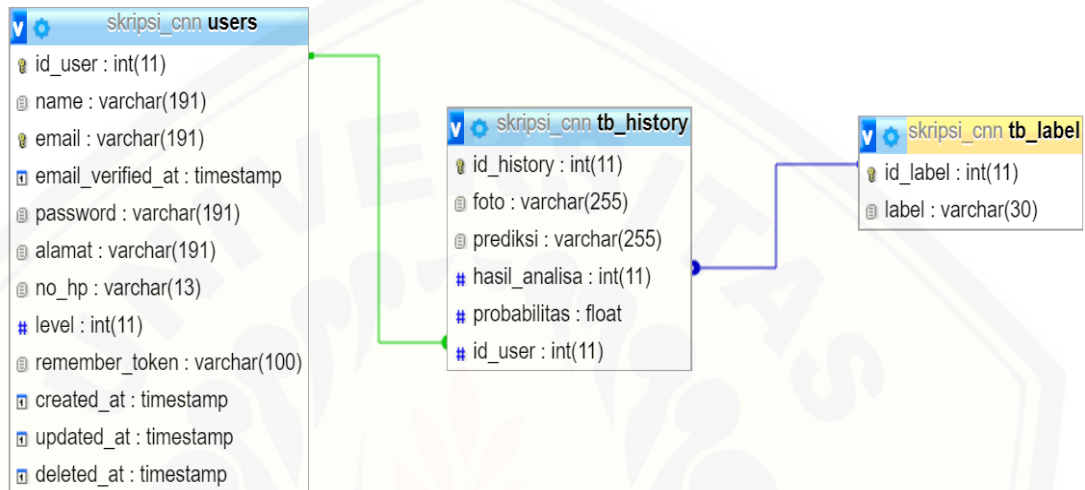
Class diagram merupakan diagram yang menggambarkan hubungan antar kelas yang akan digunakan untuk membangun sistem klasifikasi penyakit paru pada penelitian ini, *class diagram* tersebut dapat dilihat pada gambar 4.11



Gambar 4.11 Class diagram

4.2.5. Entity Relationship Diagram (ERD)

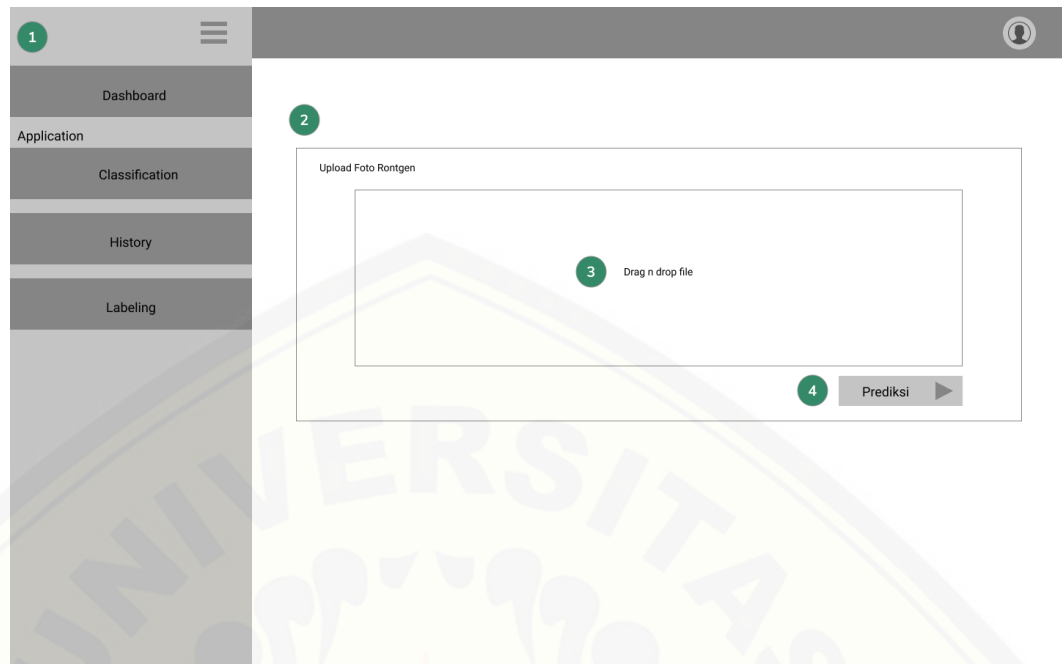
Entity Relationship Diagram merupakan *diagram* yang digunakan untuk pemodelan kebutuhan basis data, ERD dari sistem klasifikasi penyakit paru pada penelitian ini dapat dilihat pada gambar 4.12



Gambar 4.12 Entity Relationship Diagram

4.3. Desain Tampilan Integrasi Sistem

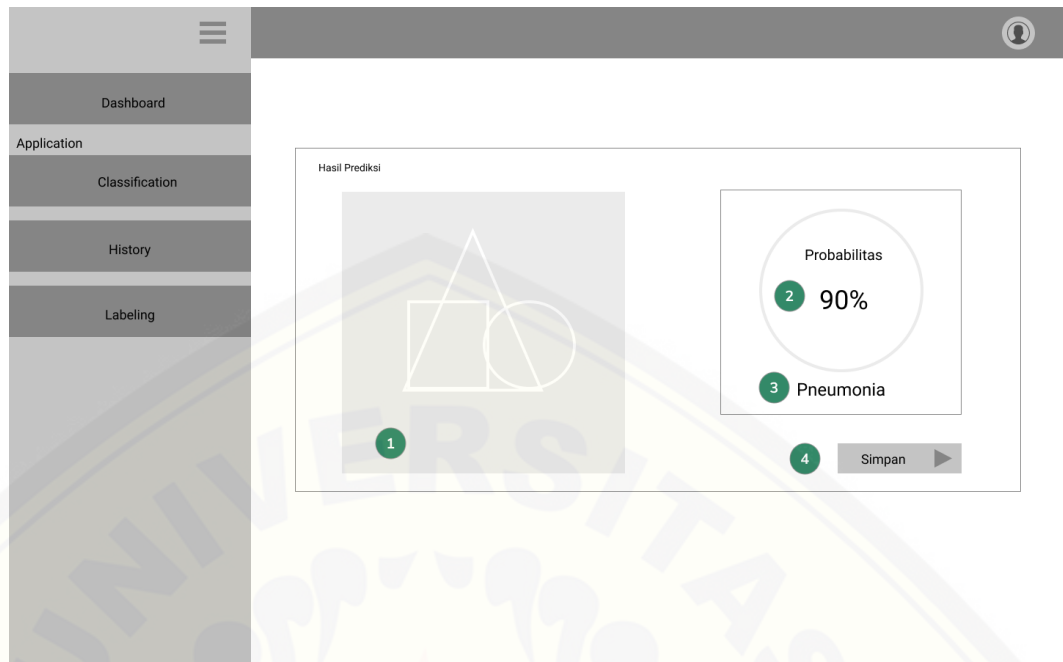
Desain tampilan integrasi sistem berupa *mockup* dari *web apps* yang nantinya akan dibuat kedalam tampilan *web apps* yang sebenarnya, *mockup* tersebut sebagai acuan peneliti untuk membuat tampilan *web apps*. Desain tampilan integrasi system dapat dilihat pada gambar 4.13 dan 4.14



Gambar 4.13 *Mockup form upload foto rontgen*

Keterangan :

1. *Side menu* : Menu apa saja yang dapat diakses oleh pengguna
2. *Form upload foto rontgen*
3. Pengguna dapat melakukan *drag n drop* foto rontgen pada tempat yang telah disediakan
4. Tombol prediksi berfungsi untuk *mengupload* foto rontgen dan menjalankan fungsi prediksi



Gambar 4.14 *Mockup* hasil prediksi sistem

Keterangan :

1. Foto *rontgen* yang telah berhasil diupload oleh pengguna
2. Tingkat kemungkinan hasil klasifikasi
3. Penyakit yang diprediksi oleh system
4. Tombol simpan berfungsi untuk menyimpan foto *rontgen* yang telah *diupload* dan hasil prediksinya

4.4. Implementasi

Implementasi adalah tahap dimana peneliti melakukan pengimplementasian desain sistem yang telah dibuat kedalam bahasa pemrograman agar menjadi sistem yang dapat mudah digunakan oleh pengguna. Dalam tahap implementasi ini dibagi kedalam dua bagian yaitu implementasi bagian *frontend* dan bagian *backend*. *Frontend* adalah bagian yang berbasis *web* bertugas untuk melakukan pengelolaan *user interface*, *frontend* dalam penelitian ini menggunakan bahasa pemrograman *HTML*, *CSS*, *Javascript* dan *PHP*, sedangkan bagian *backend* adalah bagian yang

bertugas untuk mengolah logika klasifikasi, *training data*, dan *testing model*, *backend* dalam penelitian ini menggunakan bahasa pemrograman *python*.

4.1.1. Implementasi *frontend*

Implementasi *frontend* dalam penelitian ini adalah proses penulisan kode program *HTML*, *CSS*, *Javascript* dan *PHP* untuk menghasilkan sebuah *user interface* dalam bentuk *web apps* yang dapat digunakan oleh dokter paru untuk mengakses system dengan mudah. Fitur yang terdapat dalam *web apps* tersebut antara lain *Dashboard*, *Classification*, *History*, *Labeling*.

4.1.2. Implementas *backend*

Implementasii *backend* dalam penelitian ini adalah proses penulisan kode program untuk menghasilkan sebuah program yang dapat digunakan peneliti untuk melakukan *training data*, *testing model*, *setup* variabel seperti *epoch* dan skenario data, serta melakukan pemilihan terhadap model yang akan digunakan, program tersebut ditulis menggunakan bahasa pemrograman *python*.

4.5. Pengujian Sistem

Pengujian dilakukan untuk mengevaluasi aplikasi yang dibuat telah sesuai dengan yang diharapkan atau tidak. Proses pengujian dilakukan dengan pengujian *black box*. Hasil pengujian *black box* dapat dilihat pada Tabel 4.1

Tabel 4.1 Pengujian *black box*

No	Menu	Fungsi	Kasus	Hasil	Ket
1	<i>Training Data</i>	Membuka dataset	Ketika <i>training path</i> tidak ditemukan	Menampilkan <i>exception</i> “training path not exist”	OK
			Ketika <i>training path</i> ditemukan	Memulai membaca data <i>training</i> dan memberi label	OK

				sesuai dengan nama foldernya	
		Melakukan proses <i>training</i>	Ketika memasukkan jumlah <i>epoch</i>	Proses <i>training</i> akan berhenti sesuai dengan jumlah <i>epoch</i> yang dimasukkan	OK
2	<i>Testing data</i>	Melakukan proses <i>testing</i> data secara otomatis	Ketika <i>testing path</i> tidak ditemukan	Menampilkan <i>exception</i> “testing path not exist”	OK
			Ketika <i>model path</i> tidak ditemukan	Menampilkan <i>exception</i> “model path not exist”	OK
			Ketika <i>testing path</i> dan <i>model path</i> ditemukan	Melakukan proses <i>testing</i> dan menampilkan hasilnya	OK
3	Menu <i>dashboard</i>	Melihat menu <i>dashboard</i>	Ketika belum <i>login</i>	<i>Redirect</i> ke halaman <i>login</i>	OK
			Ketika sudah <i>login</i>	Menampilkan menu <i>dashboard</i>	OK
4	Menu <i>classification</i>	Melakukan proses klasifikasi	Ketika belum <i>login</i>	<i>Redirect</i> ke halaman <i>login</i>	OK
			Ketika sudah <i>login</i>	Menampilkan menu <i>classification</i>	OK

			Ketika pengguna melakukan <i>drag n drop</i> foto rontgen	Foto rontgen dapat ditampilkan ke dalam <i>form upload</i> foto rontgen	OK
			Ketika pengguna mengklik prediksi	Sistem menampilkan hasil prediksi	OK
			Ketika pengguna mengklik <i>submit</i>	Sistem menyimpan hasil prediksi system	OK
5	Menu <i>History</i>	Melihat <i>history</i> , hasil prediksi dan Analisa dari dokter lain	Ketika belum <i>login</i>	<i>Redirect</i> ke halaman <i>login</i>	OK
			Ketika sudah <i>login</i>	Menampilkan menu <i>history</i>	OK
			Ketika pengguna mengklik detail	Sistem menampilkan detail dari hasil prediksi sistem dan Analisa dokter	OK
6	Menu <i>Labeling</i>	Melakukan proses <i>labeling</i>	Ketika belum <i>login</i>	<i>Redirect</i> ke halaman <i>login</i>	OK

			Ketika sudah <i>login</i>	Menampilkan menu <i>history</i>	OK
			Ketika pengguna mengklik detail lalu memilih label dan klik submit	Sistem menyimpan foto rontgen yang telah diberi label ke dalam database	OK
7	<i>Testing</i>	Melakukan <i>testing</i> terhadap model yang telah dibuat	Ketika fungsi <i>testing</i> dijalankan	Melakukan perhitungan dengan menggunakan persamaan 3.1, dan menampilkan hasil akurasinya	OK

BAB 6. PENUTUP

Pada bab ini merupakan bagian akhir dari penulisan skripsi, yang berisikan kesimpulan dan saran atas penelitian yang telah dilakukan. Kesimpulan berisikan hasil yang didapat dari hasil pengujian penelitian menggunakan metode yang sedang diteliti, dan saran untuk penelitian kedepannya .

6.1. Kesimpulan

Kesimpulan dari penelitian yang telah dilakukan adalah sebagai berikut :

1. Sistem klasifikasi penyakit paru ini mampu mengklasifikasikan penyakit paru berdasarkan foto rontgen yang *diupload* oleh dokter paru dengan akurasi yang baik. Sistem ini mampu mengklasifikasikan ke tiga jenis kondisi paru yaitu Normal, *Pneumonia* dan TB.
2. Berdasarkan hasil pengujian, model yang dihasilkan mencapai akurasi tertingginya yaitu 85.96% pada saat menggunakan *epoch* 50 dan skenario data 90:10, sedangkan akurasi terendah sebesar 77.19% dicapai ketika menggunakan *epoch* 17 dan skenario data 80:20.
3. *Epoch* dan skenario data dapat mempengaruhi tingkat akurasi, semakin besar data *training* yang diberikan maka akurasi yang didapat akan semakin tinggi, sedangkan jika menggunakan data *training* yang lebih sedikit, hasil yang didapat akan sangat tidak stabil.

6.2. Saran

Pengembangan lebih lanjut untuk penelitian ini dapat membangun sistem klasifikasi penyakit paru di *platform* lain seperti *mobile android* dan *iOS*. Disarankan untuk melakukan modifikasi terhadap arsitektur CNN yang digunakan atau dapat menggunakan arsitektur CNN yang lain seperti *VGG16* atau dengan menggunakan *Inception* agar dapat membandingkan akurasi antar arsitektur metode CNN.

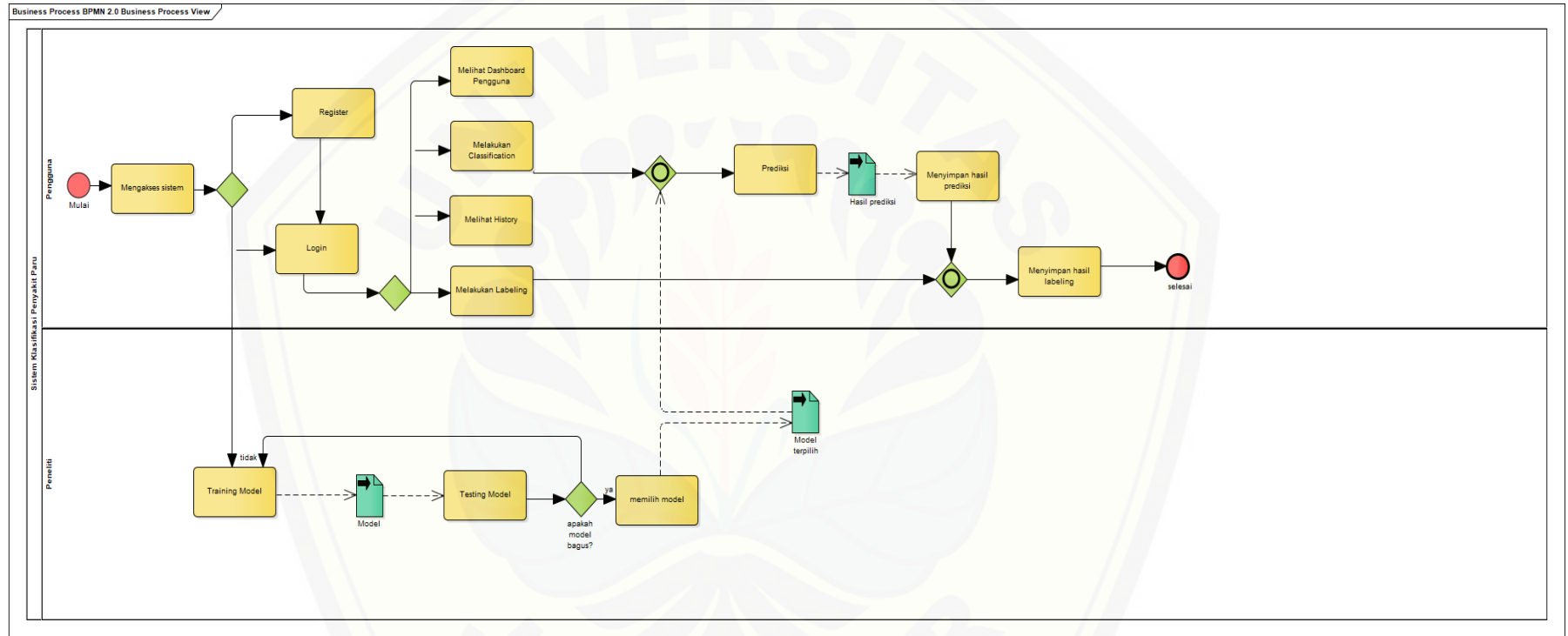
DAFTAR PUSTAKA

- Aan. (2016, 5 26). Retrieved from <http://fk.unila.ac.id: http://fk.unila.ac.id/blog/2016/05/26/persiapan-pembukaan-ppds-paru-di-fk-unila/>
- Andini, W. C., & Upahita, D. (2018, April 11). *hello sehat*. Retrieved from hello sehat: <https://helohehat.com/hidup-sehat/fakta-unik/anatomi-paru/>
- Deng, L., & Yu, D. (2014). Deep Learning: Methods and Applications. In L. Deng, & D. Yu, *Deep Learning: Methods and Applications* (pp. 199-201).
- Efni, Y., Machmud, R., & Pertiwi, D. (2016). Faktor Risiko yang Berhubungan dengan Kejadian Pneumonia pada Balita di Kelurahan Air Tawar Barat Padang. *Jurnal Kesehatan Andalas*, 5(2), 365-370.
- Forsyth, D. A., & Ponce, J. (2012). *Computer Vision A MODERN APPROACH* (2nd ed.). Upper Saddle River, New Jersey: Pearson Education, Inc.
- Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press.
- Ismiyati, Marlita, D., & Saidah, D. (2014). Pencemaran Udara Akibat Emisi Gas Buang. *Jurnal Manajemen Transportasi & Logistik (JMTransLog)*, 1(3), 241-248.
- Jaeger, S., Karargyris, A., Candemir, S., Folio, L., Siegelman, J., Callaghan, F., . . . McDonald, C. J. (2013). Automatic Tuberculosis Screening Using Chest Radiographs. *IEEE Transactions on Medical Imaging*, 2013.
- LeCun, Y., & Bengio, Y. (1995). Convolutional networks for images, speech, and time-series. *The Handbook of Brain Theory and Neural Networks*.
- LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, 436–444.
- Muchtar, N. H., Herman, D., & Yulistini. (2018). Gambaran Faktor Risiko Timbulnya Tuberkulosis Paru pada Pasien yang Berkunjung ke Unit DOTS RSUP Dr. M. Djamil Padang Tahun 2015 . *Jurnal Kesehatan Andalas*, 7(1), 80-87.

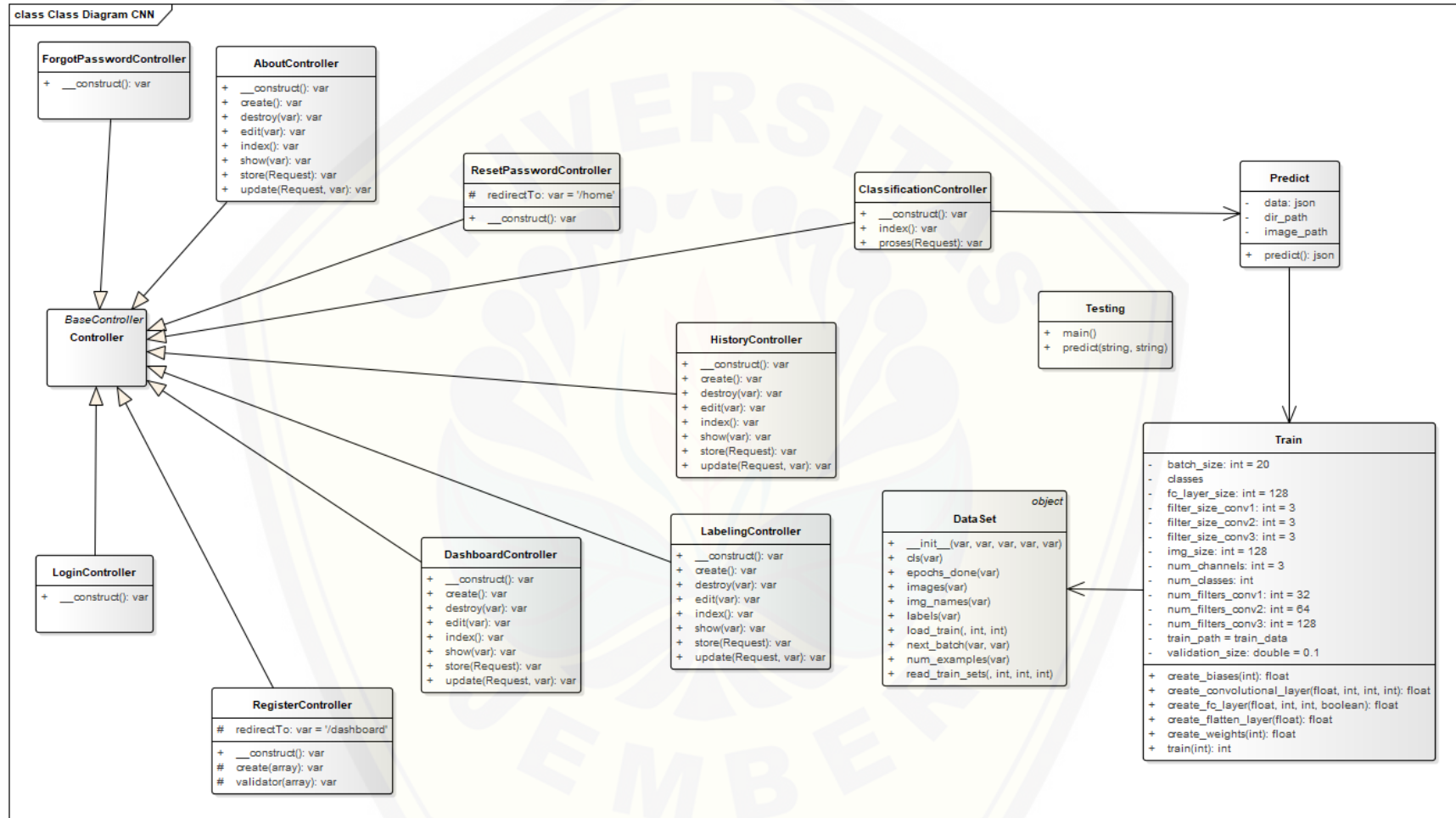
- Rahmadewi, R., & Kurnia, R. (2016). KLASIFIKASI PENYAKIT PARU BERDASARKAN CITRA RONTGEN DENGAN METODA SEGMENTASI SOBEL. *Jurnal Nasional Teknik Elektro*, 8-12.
- Rao, P., Pereira, N. A., & Srinivasan, R. (2016). Convolutional neural networks for lung cancer screening in computed tomography (CT) scans. *2nd International Conference on Contemporary Computing and Informatics (IC3I)*, 489-493.
- Smithuis, R. (2014, January 1). *The Radiology Assistant* . Retrieved from The Radiology Assistant : <http://www.radiologyassistant.nl/en/p50d95b0ab4b90/chest-x-ray-lung-disease.html>
- Szeliski, R. (2010). *Computer Vision: Algorithms and Applications*. Springer.
- Thoma, M. (2017, July 31). *Analysis and Optimization of Convolutional Neural Network*. Retrieved from arxiv.org: <https://arxiv.org/pdf/1707.09725.pdf>
- Wang, X., Peng, Y., Lu, L., Lu, Z., Bagheri, M., & Summers, R. (2017). ChestX-ray8: Hospital-scale Chest X-ray Database and Benchmarks on Weakly-Supervised Classification and Localization of Common Thorax Diseases. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 3462-3471). IEEE CVPR.

LAMPIRAN

A. Business Process Model Notation (BPMN)



B. Class Diagram



C. NLM-ChinaCXRSet-ReadMe



China Set - The Shenzhen set - Chest X-ray Database

Description: The standard digital image database for Tuberculosis is created by the National Library of Medicine, Maryland, USA in collaboration with Shenzhen No.3 People's Hospital, Guangdong Medical College, Shenzhen, China. The Chest X-rays are from out-patient clinics, and were captured as part of the daily routine using Philips DR Digital Diagnose systems.

Number of X-rays:

- 336 cases with manifestation of tuberculosis, and
- 326 normal cases.

Image parameters:

- Format: PNG
- Image size varies for each X-ray. It is approximately 3K x 3K.

Image file names are coded as CHNCXR_####_0/1.png, where '0' represents the normal and '1' represents the abnormal lung.

The clinical readings of all X-rays are saved as text files following the same file format: CHNCXR_####_0/1.txt. Each text file contains the patient's age, gender, and abnormality of the lung. (PTB: pulmonary tuberculosis).

IRB Details: The dataset was de-identified by the data providers and was exempted from IRB review at their institutions. The dataset use and public release were exempted from IRB review (No. 5357) by the NIH Office of Human Research Protections Programs.

Citation and Use:

- Please do not share the dataset outside of your research group/ organization, but forward new dataset requests to us.
- Please inform us if you find errors or inconsistencies in the data.
- It is requested that publications resulting from the use of this data attribute the source (National Library of Medicine, National Institutes of Health, Bethesda, MD, USA and Shenzhen No.3 People's Hospital, Guangdong Medical College, Shenzhen, China) and cite the following publications:
 - 1) Jaeger S, Karargyris A, Candemir S, Folio L, Siegelman J, Callaghan F, Xue Z, Palaniappan K, Singh RK, Antani S, Thoma G, Wang YX, Lu PX, McDonald CJ. Automatic tuberculosis screening using chest radiographs. *IEEE Trans Med Imaging*. 2014 Feb;33(2):233-45. doi: 10.1109/TMI.2013.2284099. PMID: 24108713
 - 2) Candemir S, Jaeger S, Palaniappan K, Musco JP, Singh RK, Xue Z, Karargyris A, Antani S, Thoma G, McDonald CJ. Lung segmentation in chest radiographs using anatomical atlases with nonrigid registration. *IEEE Trans Med Imaging*. 2014 Feb;33(2):577-90. doi: 10.1109/TMI.2013.2290491. PMID: 24239990

D. NIHCC README_CHESTXRAY

NIH Chest X-ray Dataset of 14 Common Thorax Disease Categories:

(1, Atelectasis; 2, Cardiomegaly; 3, Effusion; 4, Infiltration; 5, Mass; 6, Nodule; 7, Pneumonia; 8, Pneumothorax; 9, Consolidation; 10, Edema; 11, Emphysema; 12, Fibrosis; 13, Pleural_Thickening; 14 Hernia)

Background & Motivation: Chest X-ray exam is one of the most frequent and cost-effective medical imaging examination. However clinical diagnosis of chest X-ray can be challenging, and sometimes believed to be harder than diagnosis via chest CT imaging. Even some promising work have been reported in the past, and especially in recent deep learning work on Tuberculosis (TB) classification. To achieve clinically relevant computer-aided detection and diagnosis (CAD) in real world medical sites on all data settings of chest X-rays is still very difficult, if not impossible when only several thousands of images are employed for study. This is evident from [2] where the performance deep neural networks for thorax disease recognition is severely limited by the availability of only 4143 frontal view images [3] (OpenI is the previous largest publicly available chest X-ray dataset to date).

In this database, we provide an enhanced version (with 6 more disease categories and more images as well) of the dataset used in the recent work [1] which is approximately 27 times of the number of frontal chest x-ray images in [3]. Our dataset is extracted from the clinical PACS database at National Institutes of Health Clinical Center and consists of ~60% of all frontal chest x-rays in the hospital. Therefore we expect this dataset is significantly more representative to the real patient population distributions and realistic clinical diagnosis challenges, than any previous chest x-ray datasets. Of course, the size of our dataset, in terms of the total numbers of images and thorax disease frequencies, would better facilitate deep neural network training [2]. Refer to [1] on the details of how the dataset is extracted and image labels are mined through natural language processing (NLP).

Details: ChestX-ray dataset comprises 112,120 frontal-view X-ray images of 30,805 unique patients with the text-mined fourteen disease image labels (where each image can have multi-labels), mined from the associated radiological reports using natural language processing. Fourteen common thoracic pathologies include Atelectasis, Consolidation, Infiltration, Pneumothorax, Edema, Emphysema, Fibrosis, Effusion, Pneumonia, Pleural_thickening, Cardiomegaly, Nodule, Mass and Hernia, which is an extension of the 8 common disease patterns listed in our CVPR 2017 paper. *Note that original radiology reports (associated with these chest x-ray studies) are not meant to be publicly shared for many reasons. The text-mined disease labels are expected to have accuracy >90%. Please find more details and benchmark performance of trained models based on 14 disease labels in our arxiv paper: [1705.02315](https://arxiv.org/abs/1705.02315)*

Contents:

1. 112,120 frontal-view chest X-ray PNG images in 1024*1024 resolution (under images folder)
2. Meta data for all images (Data_Entry_2017.csv): Image Index, Finding Labels, Follow-up #, Patient ID, Patient Age, Patient Gender, View Position, Original Image Size and Original Image Pixel Spacing.