

**SKRIPSI**

**KLASIFIKASI CITRA SINAR-X PENYAKIT PARU  
MENGUNAKAN CONVOLUTIONAL NEURAL NETWORK  
BERBASIS ANDROID**

**Disusun dan diajukan oleh:**

**CALVIN RINALDY LEONARD  
D121 20 1079**



**PROGRAM STUDI SARJANA TEKNIK INFORMATIKA  
FAKULTAS TEKNIK  
UNIVERSITAS HASANUDDIN  
GOWA  
2024**

**LEMBAR PENGESAHAN SKRIPSI**

**KLASIFIKASI CITRA SINAR-X PENYAKIT PARU MENGGUNAKAN  
CONVOLUTIONAL NEURAL NETWORK BERBASIS ANDROID**

Disusun dan diajukan oleh

**CALVIN RINALDY LEONARD  
D121201079**

Telah dipertahankan di hadapan Panitia Ujian yang dibentuk dalam rangka  
Penyelesaian Studi Program Sarjana Program Studi Teknik Informatika  
Fakultas Teknik Universitas Hasanuddin  
Pada tanggal 5 Februari 2024  
dan dinyatakan telah memenuhi syarat kelulusan

Menyetujui,

Pembimbing Utama,

Pembimbing Pendamping,

Dr. Ir. Ingrid Nurtanio, M.T.  
Nip. 196108131988112001

Anugrayani Bustamin, S.T., M.T.  
Nip. 199012012018074001

Ketua Program Studi,



Prof. Dr. Ir. Indrabayu, ST., MT., M.Bus.Sys., IPM, ASEAN. Eng.  
Nip. 197507162002121004

## PERNYATAAN KEASLIAN

Yang bertanda tangan dibawah ini:

Nama : Calvin Rinaldy Leonard

NIM : D121201079

Program Studi : Teknik Informatika

Jenjang : S1

Menyatakan dengan ini bahwa karya tulisan saya berjudul

{Klasifikasi Citra Sinar-X Penyakit Paru Menggunakan Convolutional Neural Network Berbasis Android}

Adalah karya tulisan saya sendiri dan bukan merupakan pengambilan alihan tulisan orang lain dan bahwa skripsi yang saya tulis ini benar-benar merupakan hasil karya saya sendiri

Semua informasi yang ditulis dalam skripsi yang berasal dari penulis lain telah diberi penghargaan, yakni dengan mengutip sumber dan tahun penerbitannya. Oleh karena itu semua tulisan dalam skripsi ini sepenuhnya menjadi tanggung jawab penulis. Apabila ada pihak manapun yang merasa ada kesamaan judul dan atau hasil temuan dalam skripsi ini, maka penulis siap untuk diklarifikasi dan mempertanggungjawabkan segala resiko

Segala data dan informasi yang diperoleh selama proses pembuatan skripsi, yang akan dipublikasi oleh Penulis dimasa depan harus mendapat persetujuan dari Dosen Pembimbing.

Apabila dikemudian hari terbukti atau dapat dibuktikan bahwa Sebagian atau keseluruhan isi skripsi ini hasil karya orang lain, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Gowa, 28 Oktober 2023

Yang Menyatakan



Calvin Rinaldy Leonard

## ABSTRAK

**CALVIN RINALDY LEONARD.** *Klasifikasi Citra Sinar-X Penyakit Paru Menggunakan Convolutional Neural Network Berbasis Android* (dibimbing oleh Ingrid Nurtanio dan Anugrayani Bustamin)

Paru-paru adalah organ vital dalam tubuh manusia yang berfungsi sebagai pertukaran udara yaitu masuknya oksigen dan keluarnya karbon dioksida dari dalam tubuh. Dalam mendiagnosis penyakit paru, para dokter akan melakukan analisa dini pada hasil citra *x-ray thorax* untuk mencari perubahan atau kelainan yang terjadi pada paru-paru. Proses analisa yang dilakukan oleh para dokter terkadang menghasilkan diagnosis yang berbeda (*differential diagnosis*).

Penelitian ini bertujuan untuk mengklasifikasikan penyakit paru dari citra sinar-X, dimana terdapat tiga label kondisi paru. Selain itu, penelitian ini juga bertujuan untuk mengukur performa dari sistem yang telah dibuat melalui percobaan lima skenario terkait dengan *tuning* parameter *learning rate*, *batch size*, *epoch*, *convolutional layer* dan *drop out*.

Sistem dikembangkan dengan menggunakan metode *Deep Learning Convolutional Neural Network* (CNN). Arsitektur model yang digunakan adalah CNN *from scratch*. Total data yang digunakan dalam membangun sistem yaitu 812 citra yang terbagi dalam tiga label kelas kondisi paru, yaitu normal, pneumonia dan tuberkulosis. Kemudian, Sistem akan ditanamkan pada aplikasi LungX berbasis Android menggunakan *library TensorflowLite* yang dapat digunakan pada versi android 7 ke atas.

Hasil pelatihan dengan menggunakan 812 citra sinar-X menghasilkan akurasi sebesar 94%. Hasil validasi dengan 173 citra sinar-X dan pengujian dengan 175 citra sinar-X berturut-turut menghasilkan akurasi sebesar 94% dan 90% dengan parameter terbaik *learning rate 0,0001*, *batch size 8*, *epoch 35*, *4 convolutional layer* dan *drop out 0*.

Kata Kunci: Paru, Klasifikasi, Citra Sinar-X, Android, *Differential Diagnosis*, *Deep Learning*, *Convolutional Neural Network from scratch*

## ABSTRACT

**CALVIN RINALDY LEONARD.** *Classification of Lung Diseases X-Ray Image Based On Android Using Convolutional Neural Network* (supervised by Ingrid Nurtanio and Anugrayani Bustamin)

The lungs are vital organs in the human body that function as the site of air exchange, allowing the intake of oxygen and the removal of carbon dioxide from the body. Lungs are highly susceptible to diseases caused by bacteria, viruses, and fungi that can easily enter the body through the process of air exchange. In diagnosing lung diseases, doctors often perform early analysis on the results of chest X-ray images to detect changes or abnormalities in the lungs. The diagnostic process conducted by doctors can sometimes yield different results (differential diagnosis).

This research aims to classify lung diseases from X-ray images, with three labels for lung conditions. Furthermore, this research also aims to measure the performance of the system developed in 5 kinds of experiment for tuning parameters. Those consisted of learning rate, batch size, epoch, convolutional layer and drop out .

The system was developed using the Deep Learning Convolutional Neural Network (CNN) method. The model architecture employed is a CNN from scratch. The total data used in building the system consists of 812 images divided into three label classes of lung conditions: normal, pneumonia, and tuberculosis. Therefore, the system will be embedded on LungX application based on Android system using TensorflowLite library, which can be used on the Android version 7 above.

The training results using 812 X-ray images produced an accuracy of 94%. Validation with 173 X-ray images and testing with 175 X-ray images subsequently resulted in accuracies of 94% and 90%, respectively. In addition, the best parameters are learning rate 0,0001, batch size 8, epoch 35, 4 convolutional layers, and drop out 0.

**Keywords:** Lungs, Classification, X-ray Images, Android, Differential Diagnosis, Deep Learning, Convolutional Neural Network from scratch

## DAFTAR ISI

LEMBAR PENGESAHAN SKRIPSI.....	i
PERNYATAAN KEASLIAN.....	ii
ABSTRAK.....	iii
ABSTRACT.....	iv
DAFTAR ISI.....	v
DAFTAR GAMBAR.....	vi
DAFTAR TABEL.....	viii
DAFTAR SINGKATAN DAN ARTI SIMBOL.....	ix
DAFTAR LAMPIRAN.....	x
KATA PENGANTAR.....	xi
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	2
1.3 Tujuan Penelitian.....	3
1.4 Manfaat Penelitian.....	3
1.5 Ruang Lingkup.....	3
BAB II TINJAUAN PUSTAKA.....	4
2.1 Sinar-X <i>thorax</i> .....	4
2.2 Paru.....	4
2.3 <i>Histogram Equalization</i> .....	6
2.4 Koefisien Variasi.....	7
2.5 <i>Deep Learning</i> .....	8
2.6 <i>Convolutional Neural Network (CNN)</i> .....	9
2.7 <i>Confusion Matrix</i> .....	20
2.8 Akurasi, Presisi, <i>Recall</i> , dan <i>F1-Score</i> .....	21
2.9 <i>TensorFlow Lite</i> .....	22
2.10 Android.....	23
BAB III METODE PENELITIAN.....	24
3.1 Tahapan Penelitian.....	24
3.2 Waktu dan Lokasi Penelitian.....	25
3.3 Instrumen Penelitian.....	26
3.4 Teknik Pengambilan Data.....	26
3.5 Perancangan dan Implementasi Sistem.....	28
BAB IV HASIL DAN PEMBAHASAN.....	54
4.1 Hasil Penelitian.....	51
4.2 Pembahasan.....	75
BAB V KESIMPULAN DAN SARAN.....	86
5.1 Kesimpulan.....	89
5.2 Saran.....	89
DAFTAR PUSTAKA.....	91
LAMPIRAN.....	95

## DAFTAR GAMBAR

Gambar 1 Citra Sinar-X <i>thorax</i> pneumonia bercak <i>basal</i> , Citra Sinar-X <i>thorax</i> pneumonia bercak <i>apex</i> , dan Citra Sinar-X <i>thorax</i> pneumonia Konsolidasi .....	5
Gambar 2 Citra Sinar-X tuberkulosis dengan bercak dibagian <i>apex</i> paru, Citra Sinar-X tuberkulosis dengan konsolidasi .....	6
Gambar 3 Citra Sinar-X <i>thorax</i> sebelum diolah, Citra Sinar-X <i>thorax</i> setelah diolah .....	7
Gambar 4 Perbedaan <i>Machine Learning</i> dan <i>Deep Learning</i> .....	8
Gambar 5 Arsitektur <i>Convolutional Neural Network</i> (CNN) .....	9
Gambar 6 Proses konvolusi dengan <i>filter</i> .....	10
Gambar 7 Proses Konvolusi.....	11
Gambar 8 Grafik Fungsi Aktivasi <i>ReLU</i> .....	17
Gambar 9 Proses fungsi aktivasi <i>ReLU</i> .....	18
Gambar 10 Proses <i>max-pooling</i> dan <i>average-pooling</i> .....	19
Gambar 11 Proses <i>max pooling</i> .....	19
Gambar 12 <i>Fully Connected Layer</i> .....	20
Gambar 13 <i>Confusion Matrix Multiclass</i> .....	20
Gambar 14 <i>Confusion Matrix</i> dua kelas.....	21
Gambar 15 Tahapan Penelitian .....	24
Gambar 16 <i>Flowchart</i> Perancangan Sistem.....	28
Gambar 17 Data Primer Kelas Pneumonia dan Tuberkulosis.....	28
Gambar 18 Sumber Data Sekunder Paru Normal .....	19
Gambar 19 Sumber Data Sekunder Paru Tuberkulosis.....	19
Gambar 20 Perbandingan histogram dan rasio kontras data primer dan data sekunder .....	30
Gambar 21 Perbandingan histogram dan rasio kontras data primer dan sekunder setelah <i>Histogram Equalization</i> .....	31
Gambar 22 Citra sinar-X <i>thorax</i> normal sebelum <i>Histogram Equalization</i> .....	31
Gambar 23 Citra sinar-X <i>thorax</i> normal setelah <i>Histogram Equalization</i> .....	32
Gambar 24 Citra sinar-X <i>thorax</i> tuberkulosis sebelum <i>Histogram Equalization</i> .....	32
Gambar 25 Citra sinar-X <i>thorax</i> tuberkulosis setelah <i>Histogram Equalization</i> .....	32
Gambar 26 Hasil Pengurutan Selisih Kontras Terkecil Citra Normal .....	34
Gambar 27 Hasil Pengurutan Selisih Kontras Terkecil Citra Tuberkulosis.....	34
Gambar 28 Citra Pneumonia yang Direduksi Karena <i>Outlier</i> , Citra Pneumonia yang Direduksi Karena Subjek Anak Kecil.....	35
Gambar 29 Citra Tuberkulosis yang Direduksi Karena Tahap awal, Citra Tuberkulosis yang Direduksi Karena <i>Outlier</i> dan Subjek Anak Kecil .....	36
Gambar 30 Arsitektur CNN yang digunakan.....	38
Gambar 31 Visualisasi Proses <i>training</i> Arsitektur CNN.....	38
Gambar 32 Citra sinar-X Pneumonia Data <i>Training</i> .....	40
Gambar 33 Proses Konvolusi Pada data <i>training</i> .....	40
Gambar 34 Matriks Intensitas Piksel citra pneumonia 64x64 .....	34
Gambar 35 Matriks Intensitas Piksel citra pneumonia 10x10 .....	35

Gambar 36 Proses Konvolusi Pada citra asli .....	42
Gambar 37 Proses Fungsi Aktivasi .....	43
Gambar 38 Proses <i>Max Pooling</i> .....	43
Gambar 39 Proses <i>Flatten</i> .....	44
Gambar 40 <i>Use case diagram</i> sistem.....	46
Gambar 41 Tampilan splash screen LungX, Tampilan beranda, <i>Disclaimer</i> ...	46
Gambar 42 Proses Kerja Sistem di Android .....	47
Gambar 43 Model dengan 3 <i>Hidden Layer</i> (16, 32, 64) .....	51
Gambar 44 Hasil <i>training</i> dengan <i>learning rate</i> 0,001 .....	53
Gambar 45 Hasil <i>training</i> dengan <i>learning rate</i> 0,0001 .....	54
Gambar 46 Hasil <i>training</i> dengan <i>learning rate</i> 0,00001 .....	54
Gambar 47 Grafik <i>loss</i> dan <i>accuracy validation data</i> tiga nilai <i>learning rate</i>	54
Gambar 48 Grafik <i>loss</i> dan <i>accuracy testing data</i> tiga nilai <i>learning rate</i> .....	55
Gambar 49 Hasil <i>training</i> dengan <i>batch size</i> 8 .....	56
Gambar 50 Hasil <i>training</i> dengan <i>batch size</i> 16 .....	56
Gambar 51 Hasil <i>training</i> dengan <i>batch size</i> 32 .....	56
Gambar 52 Grafik <i>loss</i> dan <i>accuracy validation data</i> tiga nilai <i>batch size</i> .....	57
Gambar 53 Grafik <i>loss</i> dan <i>accuracy testing data</i> tiga nilai <i>batch size</i> .....	57
Gambar 54 Hasil <i>training epoch</i> 20 .....	58
Gambar 55 Hasil <i>training epoch</i> 35 .....	58
Gambar 56. Hasil <i>training epoch</i> 50 .....	58
Gambar 57 Grafik <i>loss</i> dan <i>accuracy testing data</i> tiga nilai <i>epoch</i> .....	59
Gambar 58 Grafik waktu <i>training</i> dengan tiga nilai <i>epoch</i> .....	59
Gambar 59 Model dengan 3 <i>Hidden Layer</i> (16, 32, 64) .....	60
Gambar 60 Hasil <i>training</i> dengan 3 (16,32,64) lapisan konvolusi .....	62
Gambar 61 Model dengan 4 <i>Hidden Layer</i> (16, 32, 64, 64) .....	62
Gambar 62 Hasil <i>training</i> dengan 4 <i>Hidden Layer</i> (16,32,64,64) .....	64
Gambar 63 Model dengan 5 <i>Hidden Layer</i> (16, 32, 64, 64, 64).....	65
Gambar 64 Hasil <i>training</i> dengan 5 <i>hidden layers</i> (16,32,64,64,64).....	67
Gambar 65 Grafik <i>loss</i> dan <i>accuracy validation data</i> tiga <i>layer</i> konvolusi ....	67
Gambar 66 Grafik <i>loss</i> dan <i>accuracy testing data</i> tiga <i>layer</i> konvolusi.....	67
Gambar 67 Hasil <i>training</i> tanpa <i>dropout</i> .....	68
Gambar 68 Hasil <i>training</i> dengan <i>dropout</i> 0,3 .....	68
Gambar 69 Hasil <i>training</i> dengan <i>dropout</i> 0,5 .....	69
Gambar 70 Grafik <i>loss</i> dan <i>accuracy validation data</i> tiga nilai <i>dropout</i> .....	69
Gambar 71 Grafik <i>loss</i> dan <i>accuracy testing data</i> tiga nilai <i>dropout</i> .....	69
Gambar 72 Hasil klasifikasi salah untuk label Pneumonia, Tuberkulosis .....	72
Gambar 73 Contoh Hasil Klasifikasi Benar.....	73
Gambar 74 Pneumonia <i>false positives</i> .....	87
Gambar 75 Tuberkulosis <i>false negative</i> .....	87



**DAFTAR TABEL**

Tabel 1 Black Box Aplikasi LungX .....	48
Tabel 2 Hasil Perhitungan Confusion Matrix data testing .....	71
Tabel 3 Hasil Perhitungan Confusion Matrix 60 data testing di Android .....	74
Tabel 4 Perbandingan Percobaan 3 learning rate 0,001, 0,0001, dan 0,00001 .....	75
Tabel 5 Perbandingan Percobaan 3 batch size 8, 16 dan 32.....	78
Tabel 6 Perbandingan Percobaan 3 epoch 20, 35 dan 50 .....	80
Tabel 7 Perbandingan Percobaan 3 macam layer konvolusi .....	82
Tabel 8 Perbandingan Percobaan 3 dropout 0, 0,3 dan 0,5 .....	85

**DAFTAR SINGKATAN DAN ARTI SIMBOL**

Lambang/Singkatan	Arti dan Keterangan
CNN	<i>Convolutional Neural Network</i>
CV	<i>Coefficient of Variation</i>
NPI	Nilai Persebaran Intensitas
$\sigma$	Standar deviasi
$\mu$	Rata-rata menghitung CV
$\bar{x}$	Rata-rata ( <i>mean</i> )
ReLu	<i>Rectified Linear Unit</i>

## DAFTAR LAMPIRAN

Lampiran 1 <i>Dataset</i> .....	96
Lampiran 2 <i>flowchart</i> Reduksi Data .....	97
Lampiran 3 <i>Source Code Program</i> .....	98
Lampiran 4 <i>Testing</i> Aplikasi <i>LungX</i> .....	106

## KATA PENGANTAR

Segala puji syukur penulis panjatkan kepada Tuhan Yang Maha Esa atas karena berkat, anugerah dan kebaikan-Nya sehingga penulis dapat menyelesaikan tugas akhir dengan judul **“Klasifikasi Citra Sinar-X Penyakit Paru Menggunakan Convolutional Neural Network (CNN) Berbasis Android”** sebagai salah satu syarat dalam menyelesaikan jenjang Strata-1 di Departemen Teknik Informatika, Fakultas Teknik, Universitas Hasanuddin.

Penulis menyadari bahwa dalam penyusunan dan penulisan tugas akhir ini tidak dapat terselesaikan dengan baik tanpa adanya bimbingan dan bantuan baik materi maupun non-materi dari berbagai pihak. Bimbingan dan bantuan tersebut adalah alasan penulisan dan penyusunan tugas akhir ini dapat terselesaikan dengan lancar. Oleh karena itu sebagai salah bentuk penghargaan yang setinggi-tingginya, penulis ingin mengucapkan terima kasih yang sebesar-besarnya kepada:

1. Keluarga penulis, Bapak Hendrik dan Ibu Mei Ling yang senantiasa mendoakan, memberikan semangat dan bantuan materi maupun non-materi kepada penulis, Viery Leonard selaku kakak kandung penulis yang selalu mendukung, mendoakan dan memberikan semangat.
2. Ibu Dr. Ir. Ingrid Nurtanio, M.T. selaku pembimbing I dan Ibu Anugrayani Bustamin, S.T., M.T. selaku pembimbing II, yang senantiasa menyempatkan waktu untuk mengarahkan dan berbagi pikiran kepada penulis untuk menyelesaikan tugas akhir.
3. Segenap dosen dan staf Departemen Teknik Informatika, Fakultas Teknik, Universitas Hasanuddin yang telah membantu penulis selama masa perkuliahan.
4. Segenap staf Rumah Sakit Universitas Hasanuddin yang telah membantu penulis selama proses pengumpulan data.
5. Segenap staf Komisi Etik FKUH-RSWS-Unhas yang telah membantu penulis dalam proses pengurusan persetujuan etik.
6. Agung, Shekinah, Rischa, Tasya, Nabila, Rizal dan teman-teman dekat penulis yang tidak dapat penulis sebutkan satu-persatu atas kebersamaan, bantuan dan semangat yang diberikan.

7. Teman-teman Rezolver 20 yang telah kebersamai penulis selama masa perkuliahan, atas waktu bersama dan hiburan-hiburan yang telah diberikan.
8. Zorro dan King selaku kucing kesayangan penulis yang senantiasa menghibur penulis dalam menyelesaikan tugas akhir.
9. Seluruh pihak yang tidak disebutkan dan tanpa sadar telah menjadi inspirasi dan membantu penulis dalam menyelesaikan tugas akhir.

Penulis berharap semoga Tuhan Yang Maha Esa berkenan membalas segala bantuan dan dukungan dari semua pihak yang membantu dalam penyelesaian tugas akhir ini. Penulis menyadari bahwa tugas akhir ini masih jauh dari kata sempurna. Oleh karena itu, penulis mengharapkan segala bentuk saran serta masukan yang membangun dari berbagai pihak. Semoga tugas akhir ini dapat berguna bagi para pembaca.

Makassar, 28 Oktober 2023

Penulis

# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

Teknologi yang semakin berkembang saat ini turut mengambil bagian dalam ranah kesehatan untuk membantu para profesional kesehatan dalam mendiagnosis sebuah penyakit. Salah satu area yang mendapatkan perhatian khusus adalah paru-paru. Paru-paru adalah organ vital dalam tubuh manusia yang berfungsi sebagai pertukaran udara, serta menjaga tubuh mendapatkan oksigen yang cukup dan mengangkut karbon dioksida keluar dari tubuh (sistem respirasi). Paru-paru terletak di dalam rongga dada (*thorax*) (C. pearce, 2016), sehingga untuk mengetahui permasalahan atau mendiagnosis penyakit tertentu pada paru-paru, maka bagian *thorax* akan dilakukan pemeriksaan sinar-X.

Dalam mendiagnosis penyakit paru, para dokter akan melakukan analisa pada hasil citra sinar-X untuk mencari perubahan atau kelainan yang terjadi pada paru-paru. Proses analisa yang dilakukan oleh para dokter terkadang menghasilkan hasil diagnosis yang berbeda (*differential diagnosis*) karena kemampuan setiap dokter dalam menganalisis hasil sinar-X berbeda, sehingga hasil diagnosis masing-masing dokter dapat berbeda (Hartono, 2017). Selain itu citra sinar-X seringkali menunjukkan pola yang mirip antara berbagai penyakit, seperti pneumonia dan tuberkulosis. Hal ini membuat diagnosis menjadi cukup lama dan dokter membutuhkan banyak waktu untuk menganalisis detail hasil sinar-X tersebut.

Dalam membantu mendukung diagnosis para dokter, hasil citra x-ray thorax juga dapat dilakukan proses komputerisasi, yaitu dengan memanfaatkan teknologi *image processing* (pengolahan gambar) karena sebuah mesin dapat menangkap detail-detail terperinci di dalam gambar jika dibandingkan dengan mata manusia secara langsung. Teknologi pengolahan gambar digunakan untuk memproses hasil citra sinar-X tersebut untuk meningkatkan kualitas dan mengekstraksi fitur-fitur dalam gambar citra tersebut sehingga dapat menghasilkan informasi yang berarti (Dendi Maysanjaya, 2020).

Citra sinar-X *thorax* yang telah diolah kemudian akan mengimplementasikan teknologi *deep learning*, sebuah metode pengembangan dari *machine learning* yang sangat cocok untuk mengolah data tidak terstruktur seperti *image*. *Deep learning* mempunyai kemampuan untuk dapat melakukan *feature engineering* secara otomatis (Dendi Maysanjaya, 2020). Metode yang akan digunakan yaitu *Convolutional Neural Networks* (CNN). Secara umum, CNN adalah algoritma dari *deep learning* yang melakukan ekstraksi fitur dengan melakukan konvolusi atau *filtering* pada input citranya dan kemudian akan memasuki lapisan *fully connected* dan dapat melakukan klasifikasi paru (normal, pneumonia dan tuberkulosis).

Dalam era mobilitas yang semakin berkembang, tidak hanya efektivitas yang menjadi faktor penting. Namun, kemudahan penggunaan juga harus menjadi fokus pengembangan. Aplikasi berbasis Android menjadi pilihan platform untuk mengintegrasikan sistem klasifikasi tersebut. Dengan adanya aplikasi Android, para petugas kesehatan dapat dengan mudah mengakses sistem klasifikasi penyakit paru dimanapun, termasuk di rumah sakit atau pun di lokasi terpencil sehingga menghemat sumber daya (Hosseini et al., 2023). Aplikasi Android yang dikembangkan akan menampilkan hasil klasifikasi beserta dengan persentase akurasi (*confidence score*) (Tobias et al., 2020).

Dengan melihat permasalahan tersebut, penulis mengusulkan judul “Klasifikasi Citra Sinar-X Penyakit Paru Menggunakan Convolutional Neural Network Berbasis Android” untuk membangun aplikasi Android yang dapat melakukan klasifikasi kondisi paru (normal, pneumonia dan tuberkulosis) dengan menggunakan metode *deep learning* CNN, sehingga dapat membantu mendukung keputusan diagnosa penyakit paru ketika terjadi *differential diagnosis*.

## 1.2 Rumusan Masalah

Berdasarkan latar belakang masalah yang telah diuraikan, maka rumusan masalah pada penelitian ini yaitu:

1. Bagaimana mengimplementasikan *deep learning* dalam melakukan klasifikasi penyakit paru menggunakan metode CNN?

2. Bagaimana tingkat akurasi yang dihasilkan sistem dalam melakukan klasifikasi penyakit paru menggunakan metode CNN?

### 1.3 Tujuan Penelitian

Dari rumusan masalah dapat diperoleh tujuan dari penelitian ini yaitu :

1. Untuk mengimplementasikan *deep learning* dalam melakukan klasifikasi penyakit paru dengan menggunakan metode CNN.
2. Untuk mengukur performa akurasi sistem dalam melakukan klasifikasi penyakit paru dengan menggunakan metode CNN.

### 1.4 Manfaat Penelitian

Melalui penelitian ini, diharapkan dapat memberikan manfaat antara lain :

1. Manfaat Bagi Tenaga Kesehatan  
Penelitian ini dapat digunakan sebagai deteksi dini dalam upaya penegah terhadap perbedaan diagnosis (*differential diagnosis*) kondisi paru.
2. Manfaat Bagi Peneliti  
Penelitian ini dapat menjadi referensi bagi penelitian-penelitian berikutnya yang relevan dengan topik klasifikasi citra sinar-X dengan metode CNN.

### 1.5 Ruang Lingkup

Ruang lingkup dalam penelitian ini adalah sebagai berikut :

1. Kondisi paru yang diklasifikasi yaitu paru normal, paru pneumonia dan paru tuberkulosis.
2. Objek penelitian difokuskan pada posisi *infiltrat* (bercak) pada paru (bercak bawah untuk pneumonia dan bercak atas untuk tuberkulosis).
3. Parameter yang digunakan yaitu nilai intensitas citra sinar-X *thorax*.
4. Metode yang digunakan yaitu *Convolutional Neural Network* (CNN).
5. Menggunakan *library TensorFlow Lite* untuk mengubah model menjadi *tflite* agar model dapat diimplementasikan pada *smartphone*.
6. Sistem hanya dapat digunakan pada Android.



## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1 Sinar-X Thorax**

Sinar-X *thorax* adalah sebuah teknik dalam diagnosis medis dengan menggunakan sinar-X (radiografi) yang menghasilkan gambar internal bagian rongga dada seseorang. Pada hasil sinar-X pada bagian dada, dapat diperoleh gambaran yang mencakup paru-paru, jantung, pembuluh darah serta tulang-tulang di dalam dada. Sinar-X *thorax* digunakan oleh dokter sebagai penilaian awal dalam diagnosis pasien karena sinar-X *thorax* dapat membantu dokter dalam mengidentifikasi setiap cedera di dalam rongga dada, seperti patah tulang rusuk, pneumotoraks, hemotoraks, kontusi paru, pneumonia maupun tuberkulosis. Namun, pada tahap awal penyakit tertentu pada paru-paru, penyakit tidak selalu terlihat dengan jelas pada sinar-X *thorax*. Sinar-X *thorax* akan selalu menjadi teknik yang berharga untuk memantau dan mengidentifikasi diagnosis dalam triase awal (Tataroglu et al., 2018).

#### **2.2 Paru**

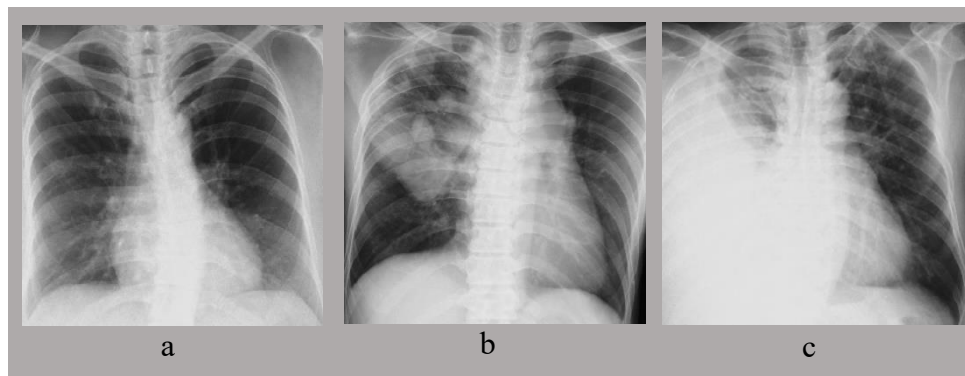
Paru-paru adalah organ vital yang sangat penting dalam sistem pernapasan manusia. Paru-paru terletak di dalam rongga dada, berada di kedua sisi dada, di bawah tulang rusuk. Paru-paru berfungsi dalam proses pertukaran udara yang kita hirup, yaitu masuknya oksigen dari luar dan memasukkan ke dalam aliran darah, dan sebaliknya mengeluarkan karbon dioksida dari darah ke luar saat proses pernafasan terjadi. Paru-paru menerapkan sistem ventilasi-perfusi dari atas ke bawah. Ventilasi yaitu aliran udara yang masuk dan keluar dari paru-paru, sedangkan perfusi mengacu pada darah yang mengalir melalui pembuluh darah diparu-paru.

Struktur paru-paru terdiri dari dua bagian utama yaitu paru-paru kanan dan paru-paru kiri. Masing-masing paru memiliki beberapa lobus, dan setiap lobus terdiri dari unit kecil disebut alveoli. Selain itu terdapat pembuluh darah dalam paru-paru. Arteri pulmonalis mengangkut darah yang sudah tidak mengandung oksigen dari ventrikel kanan jantung ke paru-paru, cabang-cabangnya menyentuh saluran-saluran bronchial, bercabang dan bercabang lagi hingga menjadi arteriola

halus. Arteriola kemudian membelah-belah dan membentuk jaringan kapiler dan kapiler itu akan menyentuh dinding alveoli (C. pearce, 2016).

### 2.2.1 Penyakit Paru Pneumonia

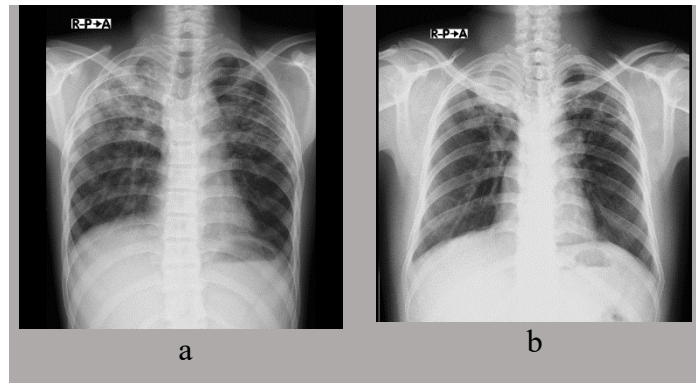
Pneumonia adalah penyakit yang menyerang paru-paru manusia. Pneumonia dapat disebabkan oleh bakteri, virus, dan jamur yang cenderung menyebabkan peradangan pada paru-paru. Pneumonia ditandai dengan adanya infeksi akut parenkim paru yang ditandai dengan gejala demam, batuk dahak, sesak napas, leukopenia atau leukositosis. Dalam kaitan antara pneumonia dan sinar-X yaitu gambar sinar-X sangat penting untuk beberapa aspek manajemen pneumonia. Sinar-X ini memiliki akurasi diagnostik sebesar 75% untuk konsolidasi alveolar dan 47% untuk efusi pleura (Natasya, 2022). Dalam membaca hasil sinar-X *thorax*, pemeriksaan rontgen akan difokuskan pada *infiltrate basal* paru.



**Gambar 1** (a) Citra sinar-X pneumonia dengan bercak di bagian basal paru. (b) Citra sinar-X pneumonia dengan bercak di bagian tengah ke atas. (c) Citra sinar-X pneumonia dengan konsolidasi

### 2.2.2 Penyakit Paru Tuberkulosis

Tuberkulosis adalah penyakit menular yang disebabkan oleh bakteri *Mycobacterium tuberculosis*. Tuberkulosis dapat menyerang semua bagian tubuh termasuk paru-paru. Tuberkulosis paru ditandai dengan infeksi kronis parenkim. Gejala yang ditimbulkan pada pasien dengan tuberkulosis yaitu lemas, mudah berkeringat pada malam hari, nyeri dada jika batuk, susah tidur, demam, batuk berkepanjangan, batuk darah dan nafsu makan menurun (Sari & Setyawati, 2022). Dalam membaca hasil sinar-X *thorax*, pemeriksaan rontgen akan difokuskan pada bagian *apex* paru.

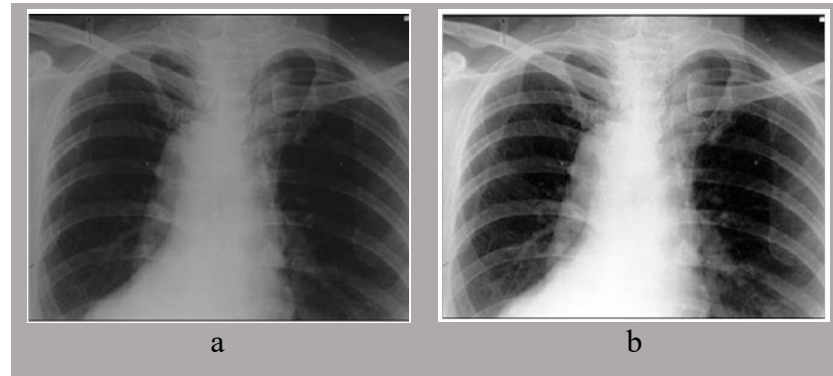


**Gambar 2 (a)** Citra sinar-X tuberkulosis dengan bercak di bagian apex paru. **(b)** Citra sinar-X tuberkulosis dengan konsolidasi.

### 2.3 *Histogram Equalization*

*Histogram Equalization* adalah sebuah teknik dalam pengolahan gambar (*image processing*). *Histogram Equalization* digunakan untuk meningkatkan kontras pada gambar ketika sebaran intensitas tidak merata (Ouledroun et al., 2023). Histogram yang dominan di bagian kiri atau intensitas bawah, maka gambar cenderung lebih gelap, sedangkan jika histogram dominan di bagian kanan atau intensitas atas, maka gambar cenderung lebih terang, dan jika histogram hanya memiliki jarak intensitas yang dekat, maka gambar disebut sebagai *poor contrast* (Sia et al., 2013).

*Histogram Equalization* dalam kaitannya dengan radiografi yaitu terkadang kualitas citra radiografi masih belum terlalu bagus karena disebabkan alat scan yang belum terlalu baik kualitasnya. Oleh karena itu dengan menggunakan *Histogram Equalization* maka citra radiografi tersebut dapat diubah tingkat keabuan-nya merata ke semua tingkat keabuan. *Histogram Equalization* dalam penelitian (I. Akhlis, 2011) berhasil meningkatkan kontras citra yang ditandai dengan standar deviasi tingkat keabuan citra hasil lebih tinggi dari pada standar tingkat keabuan citra asal.



**Gambar 3 (a)** Citra sinar-X thorax sebelum diolah. **(b)** Citra sinar-X thorax setelah diolah.

## 2.4 *Coefficient of Variation*

Koefisien Variasi (CV) adalah pengukuran statistik yang digunakan untuk mengukur keragaman intensitas gambar dalam suatu distribusi relative terhadap rata-rata intensitas gambar tersebut. Dengan menggunakan CV akan memberikan nilai yang dapat mendeskripsikan sejauh mana nilai intensitas dua gambar berbeda dalam hal kontras atau Nilai Persebaran Intensitas (NPI), serta menjadi salah satu metrik gambar untuk melihat sejauh mana perbedaan intensitas gambar tersebut (Higaki et al., 2020). Semakin tinggi nilai CV, maka semakin besar kontras dalam gambar tersebut. Koefisien variasi diperoleh dengan menggunakan persamaan yang merupakan metrik standar menggunakan nilai rata-rata piksel, yang diperoleh melalui perbandingan antara standar deviasi dan rata-rata (Nurkamilah et al., 2023). Sehingga parameter – parameter yang digunakan yaitu:

a. Standar deviasi ( $\sigma$ )

Standar deviasi adalah pengukuran statistik untuk melihat variasi data dalam sampel atau populasi. Standar deviasi menghasilkan sebuah jarak nilai dalam data cenderung bervariasi dari rata-rata.

b. Rata-rata ( $\mu$ )

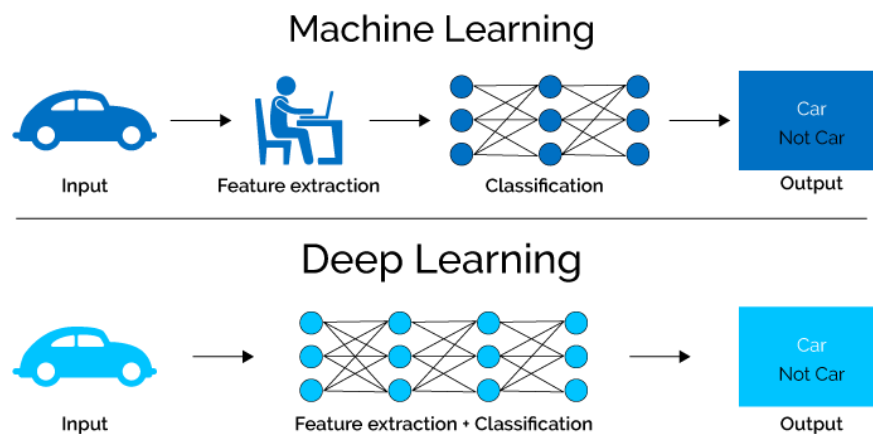
Rata-rata adalah pusat dari seluruh data yang diperoleh dari jumlah seluruh nilai dibagi dengan banyaknya nilai.

Dengan menggunakan dua parameter yaitu standar deviasi dan rata-rata maka menghasilkan persamaan sebagai berikut.

$$\text{Rasio Kontras} = \frac{\sigma}{\mu} \quad (1)$$

## 2.5 Deep Learning

*Deep Learning* adalah sebuah metode pengembangan dari *machine learning*. *Deep learning* berfokus pada penggunaan jaringan saraf tiruan yang sangat dalam (*Deep Neural Networks*) dalam melakukan *feature engineering* yaitu proses mengolah data untuk mengidentifikasi pola secara detail atau kompleks. *Deep learning* belajar secara mendalam dengan mengambil representasi data yang sangat rinci dari *dataset* yang ada (Taye, 2023).



**Gambar 4** Perbedaan machine learning dan deep learning

*Machine learning* menerima sebuah inputan yang sama dengan *deep learning*. Namun, perbedaan utama terletak pada fitur yang diperoleh. *Machine Learning* memerlukan bantuan dari manusia untuk menentukan pada data tertentu, fitur mana yang cocok untuk digunakan. Sedangkan pada *deep learning* tidak lagi membutuhkan bantuan manusia dalam mencari fitur yang penting, karena *deep learning* akan melakukan proses *feature engineering* secara detail dan sangat mendalam (kompleks) sehingga akan muncul banyak variasi fitur yang diekstraksi.

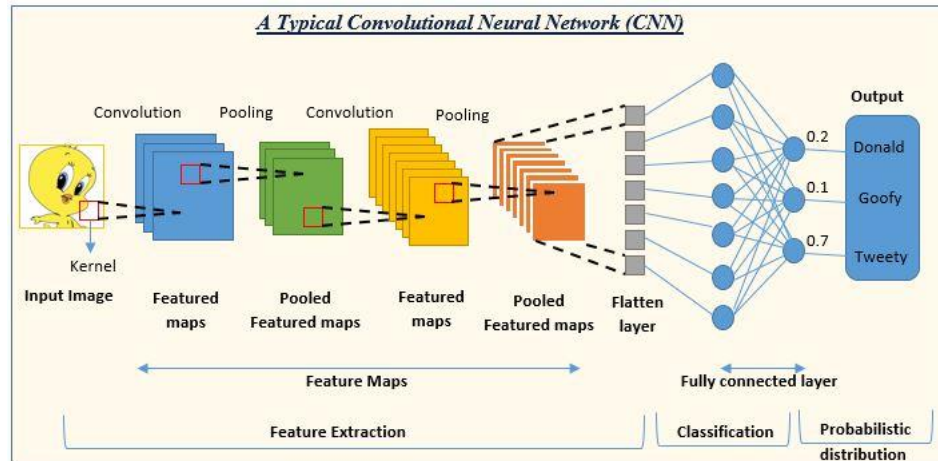
Selain dapat melakukan *feature engineering* secara otomatis, *deep learning* juga dapat sangat baik untuk mengolah data tidak terstruktur seperti gambar, suara dan video dengan hasil akurasi yang tinggi. Namun, dibalik kelebihanannya, *deep learning* membutuhkan data yang banyak, *deep learning* membutuhkan perangkat keras yang bagus untuk mampu memproses banyak parameter, *deep learning* membutuhkan waktu komputasi yang sangat lama dan *deep learning* sangat

memungkinkan untuk terjadinya *overfitting* yaitu keadaan dimana model tidak mampu bekerja secara maksimal pada data baru (J. W. G. Putra, 2020).

## 2.6 Convolutional Neural Network (CNN)

*Convolutional Neural Network* (CNN) atau jaringan saraf konvolusional adalah jenis arsitektur jaringan saraf tiruan yang digunakan dalam pengolahan data citra dan pengenalan pola visual. CNN menerapkan metode *supervised learning*, yaitu menggunakan data yang telah diberi label, kemudian CNN akan belajar untuk mengidentifikasi fitur-fitur yang relevan untuk klasifikasi. CNN yang merupakan arsitektur dari *deep learning* sangat efektif dalam *image classification*, *object detection* dan *segmentation tasks*. Namun, CNN membutuhkan perangkat keras yang bagus, biaya yang besar, waktu komputasi yang lama dan kemungkinan terjadinya *overfit* cukup tinggi.

Secara umum, arsitektur CNN menggunakan beberapa tahapan yaitu *input layers*, *convolutional layers*, *activation layers*, *pooling layers*, *fully connected layer*, *output layers* (R. A. D. I. Putra, 2023).

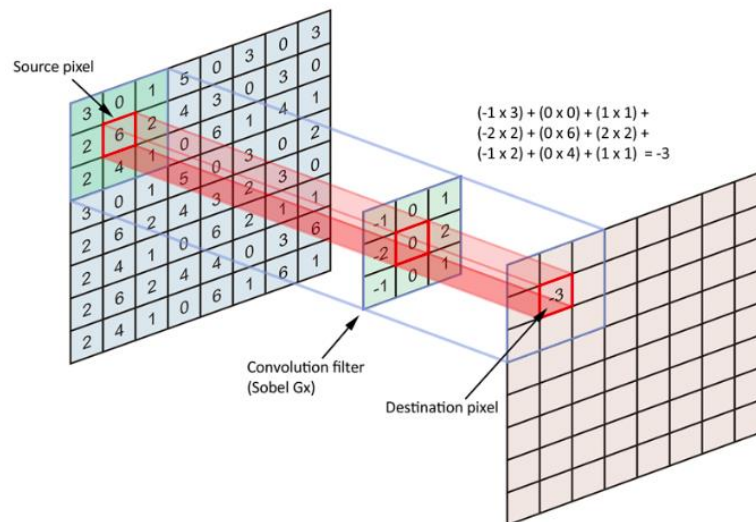


**Gambar 5** Arsitektur Convolutional Neural Network (CNN)

### 2.6.1 Convolutional Layer

*Convolutional Layer* atau lapisan konvolusi adalah komponen kunci dalam *Convolutional Neural Network* (CNN) yang dirancang untuk melakukan pengolahan citra fitur untuk mencari pola tertentu. Lapisan konvolusi dapat dikatakan sebagai pencari pola atau fitur yang bertujuan untuk menemukan fitur-fitur penting dalam gambar, seperti fitur tepi, fitur

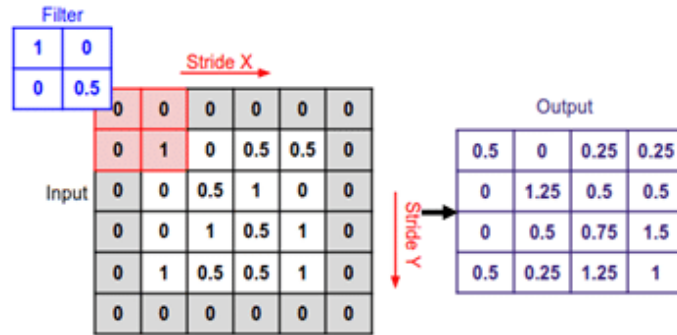
sudut, dan fitur tekstur. Dalam lapisan konvolusi terdapat beberapa parameter yang dapat diatur seperti, jumlah filter, ukuran kernel, perpindahan (*stride*), dan *padding*. *Filter-filter* akan digeser dengan jarak yang telah diatur pada *stride* tertentu (1,2,3) dengan ukuran kernel tertentu (3x3, 5x5, 7x7), dengan *padding same* atau *valid*. Hasil dari lapisan konvolusi adalah *feature map* yaitu hasil ekstraksi fitur pada gambar dengan nilai filter tertentu (Azmi & Defit, 2023).



**Gambar 6** Proses konvolusi dengan filter

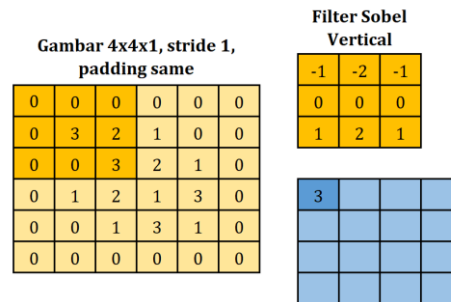
Pada gambar 6. Dapat dilihat bahwa terdapat sebuah *filter sobel gx* dengan ukuran kernel 3x3 dan *padding same* yang berarti ukuran tidak akan tereduksi karena bagian yang kosong akan diganti dengan nol (*zero padding*).

Pada konvolusi, masukan gambar akan di konvolusi atau di *filtering* dengan dengan *filter* yang telah ditetapkan jumlahnya beserta ukuran kernelnya. Proses konvolusi sebenarnya merupakan perkalian dot antara *input* data dengan *filter* kemudian menjumlahkan hasil perkalian tersebut. Proses konvolusi akan membuat ukuran gambar semakin kecil, sehingga hal ini dapat diatasi dengan menggunakan *padding same* yang akan mempertahankan ukuran gambar.



Gambar 7 Proses konvolusi

Pada penelitian ini, masukan gambar memiliki ukuran 64x64x1 yang berarti gambar hanya memiliki 1 *channel grayscale*. Proses konvolusi dapat dihitung dengan ukuran *layer* 4x4 dan ukuran kernel 3x3 dengan *stride* atau perpindahan 1 serta *padding same*. Dengan menerapkan *padding same*, ukuran gambar input tidak akan menjadi kecil karena akan ada satu lapisan di luar matriks yaitu *zero padding*. Contoh Proses perhitungan yang terjadi pada konvolusi adalah sebagai berikut.



$$[1,1] = (0 \times -1) + (0 \times -2) + (0 \times -1) + (0 \times 0) + (3 \times 0) + (2 \times 0) + (0 \times 1) + (0 \times 2) + (3 \times 1) = 3$$



$$[1,2] = (0 \times -1) + (0 \times -2) + (0 \times -1) + (3 \times 0) + (2 \times 0) + (1 \times 0) + (0 \times 1) + (3 \times 2) + (2 \times 1) = 8$$



**Gambar 4x4x1, stride 1,  
padding same**

0	0	0	0	0	0
0	3	2	1	0	0
0	0	3	2	1	0
0	1	2	1	3	0
0	0	1	3	1	0
0	0	0	0	0	0

**Filter Sobel  
Vertical**

-1	-2	-1
0	0	0
1	2	1

3	8	8	

$$[1,3] = (0 \times -1) + (0 \times -2) + (0 \times -1) + (2 \times 0) + (1 \times 0) + (0 \times 0) + (3 \times 1) + (2 \times 2) + (1 \times 1) = 8$$

**Gambar 4x4x1, stride 1,  
padding same**

0	0	0	0	0	0
0	3	2	1	0	0
0	0	3	2	1	0
0	1	2	1	3	0
0	0	1	3	1	0
0	0	0	0	0	0

**Filter Sobel  
Vertical**

-1	-2	-1
0	0	0
1	2	1

3	8	8	4

$$[1,4] = (0 \times -1) + (0 \times -2) + (0 \times -1) + (1 \times 0) + (0 \times 0) + (0 \times 0) + (2 \times 1) + (1 \times 2) + (0 \times 1) = 4$$

**Gambar 4x4x1, stride 1,  
padding same**

0	0	0	0	0	0
0	3	2	1	0	0
0	0	3	2	1	0
0	1	2	1	3	0
0	0	1	3	1	0
0	0	0	0	0	0

**Filter Sobel  
Vertical**

-1	-2	-1
0	0	0
1	2	1

3	8	8	4
-4			

$$[2,1] = (0 \times -1) + (3 \times -2) + (2 \times -1) + (0 \times 0) + (0 \times 0) + (3 \times 0) + (0 \times 1) + (1 \times 2) + (2 \times 1) = -4$$

Gambar 4x4x1, stride 1,  
padding same

0	0	0	0	0	0
0	3	2	1	0	0
0	0	3	2	1	0
0	1	2	1	3	0
0	0	1	3	1	0
0	0	0	0	0	0

Filter Sobel  
Vertical

-1	-2	-1
0	0	0
1	2	1

3	8	8	4
-4	-2		

$$[2,2] = (3 \times -1) + (2 \times -2) + (1 \times -1) + (0 \times 0) + (3 \times 0) + (2 \times 0) + (1 \times 1) + (2 \times 2) + (1 \times 1) = -2$$

Gambar 4x4x1, stride 1,  
padding same

0	0	0	0	0	0
0	3	2	1	0	0
0	0	3	2	1	0
0	1	2	1	3	0
0	0	1	3	1	0
0	0	0	0	0	0

Filter Sobel  
Vertical

-1	-2	-1
0	0	0
1	2	1

3	8	8	4
-4	-2	3	

$$[2,3] = (3 \times -1) + (2 \times -2) + (1 \times -1) + (0 \times 0) + (3 \times 0) + (2 \times 0) + (1 \times 1) + (2 \times 2) + (1 \times 1) = 3$$

Gambar 4x4x1, stride 1,  
padding same

0	0	0	0	0	0
0	3	2	1	0	0
0	0	3	2	1	0
0	1	2	1	3	0
0	0	1	3	1	0
0	0	0	0	0	0

Filter Sobel  
Vertical

-1	-2	-1
0	0	0
1	2	1

3	8	8	4
-4	-2	3	6

$$[2,4] = (1 \times -1) + (0 \times -2) + (0 \times -1) + (2 \times 0) + (1 \times 0) + (0 \times 0) + (1 \times 1) + (3 \times 2) + (0 \times 1) = 6$$

**Gambar 4x4x1, stride 1,  
padding same**

0	0	0	0	0	0
0	3	2	1	0	0
0	0	3	2	1	0
0	1	2	1	3	0
0	0	1	3	1	0
0	0	0	0	0	0

**Filter Sobel  
Vertical**

-1	-2	-1
0	0	0
1	2	1

3	8	8	4
-4	-2	3	6
-2			

$$[3,1] = (0 \times -1) + (0 \times -2) + (3 \times -1) + (0 \times 0) + (1 \times 0) + (2 \times 0) + (0 \times 1) + (0 \times 2) + (1 \times 1) = -2$$

**Gambar 4x4x1, stride 1,  
padding same**

0	0	0	0	0	0
0	3	2	1	0	0
0	0	3	2	1	0
0	1	2	1	3	0
0	0	1	3	1	0
0	0	0	0	0	0

**Filter Sobel  
Vertical**

-1	-2	-1
0	0	0
1	2	1

3	8	8	4
-4	-2	3	6
-2	-3		

$$[3,2] = (0 \times -1) + (3 \times -2) + (2 \times -1) + (1 \times 0) + (2 \times 0) + (1 \times 0) + (0 \times 1) + (1 \times 2) + (3 \times 1) = -3$$

**Gambar 4x4x1, stride 1,  
padding same**

0	0	0	0	0	0
0	3	2	1	0	0
0	0	3	2	1	0
0	1	2	1	3	0
0	0	1	3	1	0
0	0	0	0	0	0

**Filter Sobel  
Vertical**

-1	-2	-1
0	0	0
1	2	1

3	8	8	4
-4	-2	3	6
-2	-3	0	

$$[3,3] = (3 \times -1) + (2 \times -2) + (1 \times -1) + (2 \times 0) + (1 \times 0) + (3 \times 0) + (1 \times 1) + (3 \times 2) + (1 \times 1) = 0$$

Gambar 4x4x1, stride 1,  
padding same

0	0	0	0	0	0
0	3	2	1	0	0
0	0	3	2	1	0
0	1	2	1	3	0
0	0	1	3	1	0
0	0	0	0	0	0

Filter Sobel  
Vertical

-1	-2	-1
0	0	0
1	2	1

3	8	8	4
-4	-2	3	6
-2	-3	0	1

$$[3,4] = (2 \times -1) + (1 \times -2) + (0 \times -1) + (1 \times 0) + (3 \times 0) + (0 \times 0) + (3 \times 1) + (1 \times 2) + (0 \times 1) = 1$$

Gambar 4x4x1, stride 1,  
padding same

0	0	0	0	0	0
0	3	2	1	0	0
0	0	3	2	1	0
0	1	2	1	3	0
0	0	1	3	1	0
0	0	0	0	0	0

Filter Sobel  
Vertical

-1	-2	-1
0	0	0
1	2	1

3	8	8	4
-4	-2	3	6
-2	-3	0	1
-4			

$$[4,1] = (0 \times -1) + (1 \times -2) + (2 \times -1) + (0 \times 0) + (0 \times 0) + (1 \times 0) + (0 \times 1) + (0 \times 2) + (0 \times 1) = -4$$

Gambar 4x4x1, stride 1,  
padding same

0	0	0	0	0	0
0	3	2	1	0	0
0	0	3	2	1	0
0	1	2	1	3	0
0	0	1	3	1	0
0	0	0	0	0	0

Filter Sobel  
Vertical

-1	-2	-1
0	0	0
1	2	1

3	8	8	4
-4	-2	3	6
-2	-3	0	1
-4	-6		

$$[4,2] = (1 \times -1) + (2 \times -2) + (1 \times -1) + (0 \times 0) + (1 \times 0) + (3 \times 0) + (0 \times 1) + (0 \times 2) + (0 \times 1) = -6$$

Gambar 4x4x1, stride 1, padding same						Filter Sobel Vertical			
0	0	0	0	0	0	-1	-2	-1	
0	3	2	1	0	0	0	0	0	
0	0	3	2	1	0	1	2	1	
0	1	2	1	3	0	3	8	8	4
0	0	1	3	1	0	-4	-2	3	6
0	0	0	0	0	0	-2	-3	0	1
						-4	-6	-7	

$$[4,3] = (2 \times -1) + (1 \times -2) + (3 \times -1) + (1 \times 0) + (3 \times 0) + (1 \times 0) + (0 \times 1) + (0 \times 2) + (0 \times 1) = -7$$

Gambar 4x4x1, stride 1, padding same						Filter Sobel Vertical			
0	0	0	0	0	0	-1	-2	-1	
0	3	2	1	0	0	0	0	0	
0	0	3	2	1	0	1	2	1	
0	1	2	1	3	0	3	8	8	4
0	0	1	3	1	0	-4	-2	3	6
0	0	0	0	0	0	-2	-3	0	1
						-4	-6	-7	-7

$$[4,4] = (1 \times -1) + (3 \times -2) + (0 \times -1) + (3 \times 0) + (1 \times 0) + (0 \times 0) + (0 \times 1) + (0 \times 2) + (0 \times 1) = -7$$

Proses diatas adalah proses konvolusi, dimana *filter sobel vertical* atau ekstraksi fitur tepi vertikal akan bergerak 1 langkah dan melakukan perkalian dot pada matriks gambar.

## 2.6.2 Activation Function

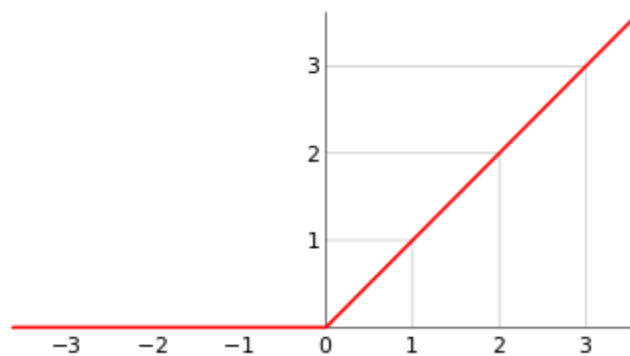
*Activation function* atau fungsi aktivasi adalah fungsi yang berperan dalam memperkenalkan non-linearitas ke dalam jaringan sehingga variasi data yang terbentuk menjadi lebih detail dan kompleks. Pemilihan fungsi aktivasi dalam arsitektur CNN sangat memengaruhi kinerja model secara signifikan (Wathani et al., 2023). Terdapat beberapa jenis aktivasi fungsi yang umum digunakan yaitu fungsi aktivasi *ReLU* dan *Softmax*.

a. Fungsi aktivasi *ReLU*

Fungsi aktivasi *ReLU* (*Rectified Linear Unit*) adalah fungsi aktivasi yang mengembalikan nilai input jika input positif (lebih besar dari 0) atau mengembalikan nilai 0 jika input negative (kurang dari atau sama dengan 0). Fungsi aktivasi *ReLU* memiliki keuntungan terhadap fungsi aktivasi *sigmoid* dan *tanh* karena tidak pernah menjadi jenuh terhadap nilai  $x$  yang besar (Ode & Sagala, 2022). Meskipun sederhana, fungsi aktivasi *ReLU* mampu menerapkan konsep non-linearitas yang dapat memungkinkan jaringan untuk mengabaikan nilai yang *outlier* atau tidak relevan. Persamaan 2 dari fungsi aktivasi *ReLU* adalah sebagai berikut.

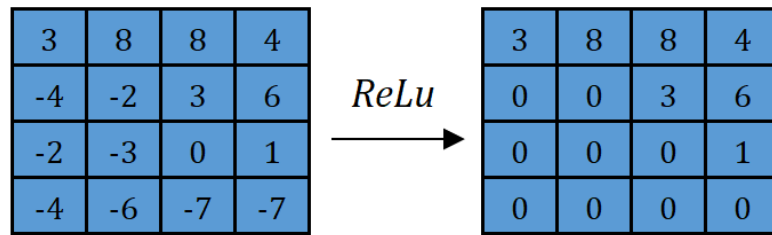
$$ReLU(x) = \max(0, x) \quad (2)$$

Grafik fungsi aktivasi *ReLU* dapat dilihat pada gambar 8.



**Gambar 8** Grafik fungsi aktivasi *ReLU*

Hasil dari proses konvolusi akan menghasilkan *feature map*. Selanjutnya *feature map* akan dikenakan fungsi aktivasi *ReLU* yang akan mengubah semua nilai yang lebih kecil atau sama dengan nol menjadi 0 dan mempertahankan nilai yang lebih besar dari 0. Proses dari fungsi aktivasi *ReLU* dapat dilihat pada gambar 9.



**Gambar 9** Proses fungsi aktivasi *ReLU*

b. Fungsi aktivasi *softmax*

Fungsi aktivasi *softmax* adalah fungsi yang berfokus pada konteks pengenalan pola klasifikasi. Fungsi aktivasi *softmax* adalah cara yang efektif untuk mengubah sejumlah bilangan riil dalam rentang nilai apa pun menjadi distribusi probabilitas yang berjumlah satu. Fungsi aktivasi *softmax* akan mengubah vektor yang berisi nilai-nilai numerik menjadi distribusi probabilitas dengan menjalankan setiap nilai melalui fungsi eksponensial dan akan dinormalisasi. Fungsi aktivasi *softmax* akan menghasilkan masing-masing peluang dari sebuah model multi-kelas (Ode & Sagala, 2022). Persamaan 3 dari fungsi aktivasi *softmax* adalah sebagai berikut.

$$\sigma(z_i) = \frac{e^{z_i}}{\sum_{j=1}^k e^{z_j}} \quad (3)$$

Keterangan:

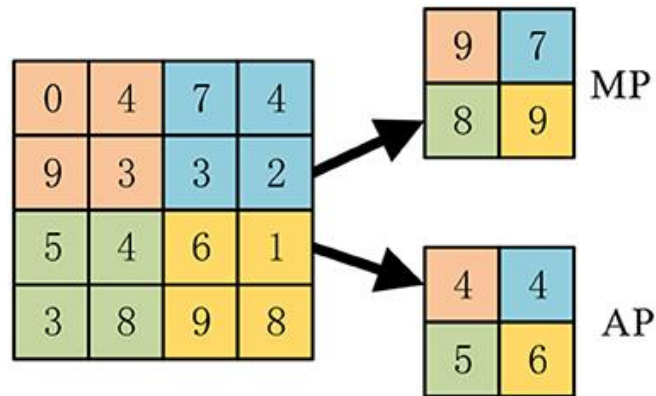
$\sigma(z_i)$  : hasil probabilitas / *confidence score* label ke-i

$e^{z_i}$  : eksponensial dari neuron label ke-i

$\sum_{j=1}^k e^{z_j}$  : jumlah dari  $e^{z_j}$  dari  $j = 1$  sampai  $k$

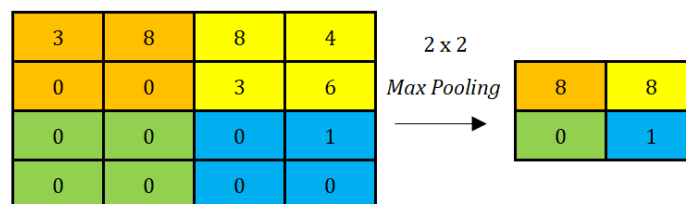
### 2.6.3 *Pooling Layer*

*Pooling Layer* adalah lapisan yang digunakan untuk mereduksi dimensi spasial dari gambar yang dihasilkan setelah proses konvolusi dan aktivasi. Lapisan *pooling* ini bertujuan untuk mengambil fitur utama atau penting pada gambar untuk mengurangi *overfitting* dan mempercepat waktu komputasi.



**Gambar 10** Proses *max-pooling* dan *average-pooling*

Terdapat dua metode yang dapat dilakukan dalam lapisan *pooling* yaitu *max-pooling* dan *average-pooling*. *Max-pooling* yaitu mengambil nilai intensitas terbesar dengan ukuran kernel tertentu. Sedangkan, *average-pooling* yaitu menghitung rata-rata dari nilai intensitas yang tersebar dengan ukuran kernel tertentu. Namun, metode *pooling* yang paling banyak digunakan yaitu metode *max-pooling* (Hidayat et al., 2019).



**Gambar 11** Proses *max pooling*

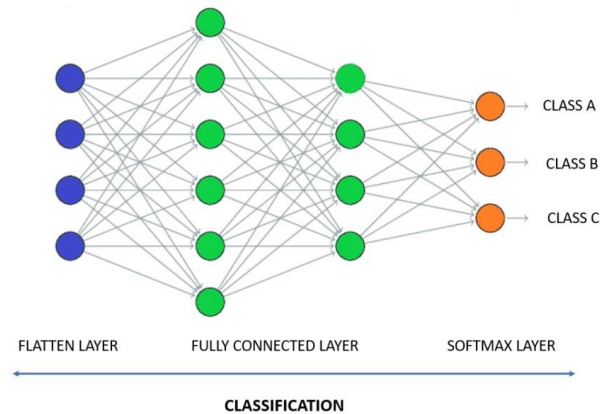
Pada gambar 11 diatas, *max pooling* bekerja dengan menerapkan kernel berukuran 2x2 dengan stride atau perpindahan 2 untuk mengambil nilai maksimum atau terbesar dari matriks input 4x4 sehingga menghasilkan hasil *feature map* yang jauh lebih kecil berukuran 2x2.

#### 2.6.4 Fully Connected Layer

*Fully Connected Layer* atau dikenal lapisan *dense* adalah lapisan yang berfungsi untuk menggabungkan informasi yang diperoleh dari lapisan-lapisan sebelumnya (*feature map*). Lapisan *Fully Connected* akan menghubungkan semua neuron satu sama lain (Liu et al., 2018). Pada CNN, lapisan ini akan digunakan pada bagian



akhir jaringan saraf, yaitu dalam proses klasifikasi dengan mengambil fitur-fitur yang telah diekstrak sebelumnya (*feature map*), kemudian menggabungkan semua fiturnya untuk menghasilkan prediksi sesuai dengan label yang sesuai.



**Gambar 12** *Fully Connected Layer*

## 2.7 Confusion Matrix

*Confusion Matrix* adalah teknik yang digunakan untuk mengevaluasi hasil klasifikasi untuk mengukur kinerja model klasifikasi. Ukuran baris dan kolom *confusion matrix* bergantung dari jumlah kelas atau label yang ada. Misalnya jika jumlah kelas ada 3 yaitu kecil, sedang, besar maka matriks akan berukuran menjadi 3x3 (Salmon et al., 2015). Dengan adanya *Confusion matrix* maka dapat diperoleh parameter evaluasi seperti presisi (*precision*), *recall*, akurasi (*accuracy*) dan *F1-score*. Jika label klasifikasi yang digunakan lebih dari dua maka *confusion matrix* akan menjadi ukuran sesuai dengan jumlah label yaitu 3x3 seperti pada gambar 13.

		True Class		
		A	B	C
Predicted Class	A	TP <sub>A</sub>	E <sub>BA</sub>	E <sub>CA</sub>
	B	E <sub>AB</sub>	TP <sub>B</sub>	E <sub>CB</sub>
	C	E <sub>AC</sub>	E <sub>BC</sub>	TP <sub>C</sub>

**Gambar 13** *Confusion Matrix Multiclass*

Untuk dapat menghitung nilai akurasi, presisi, *recall* dan *f1-score* maka *confusion matrix multiclass* untuk setiap kelasnya akan dibentuk *confusion matrix* 2x2 seperti pada gambar 14. Kemudian melakukan perhitungan pada parameter TP, FP, FN dan TN.

		Actual Values	
		Positive (1)	Negative (0)
Predicted Values	Positive (1)	TP	FP
	Negative (0)	FN	TN

**Gambar 14** *Confusion Matrix* dua kelas

Pada gambar 14, terdapat 4 parameter yaitu TP, FP, FN dan TN. Parameter ini adalah hubungan antara nilai prediksi dan nilai aktual.

- True Positives* (TP): jumlah kasus yang labelnya positif dan diprediksi sebagai positif.
- True Negatives* (TN): jumlah kasus yang labelnya negatif dan diprediksi sebagai negatif.
- False Positives* (FP): jumlah kasus yang label sebenarnya adalah negatif tetapi diprediksi sebagai positif.
- False Negatives* (FN): jumlah kasus yang label sebenarnya adalah positif tetapi diprediksi sebagai negatif.

## 2.8 Akurasi, Presisi, *Recall*, dan *F1-Score*

Akurasi, Presisi, *Recall*, dan *F1-Score* adalah parameter yang digunakan untuk melakukan evaluasi terhadap kinerja model (Rahayu et al., 2021).

- Akurasi (*accuracy*): mengukur sejauh mana model benar-benar melakukan prediksi yang sesuai yang ditandai dengan hasil klasifikasi data benar. Persamaan 4 untuk menghitung akurasi adalah sebagai berikut:

$$akurasi = \frac{(TP+TN)}{(TP+TN+FP+FN)} \quad (4)$$

- b. Presisi (*Precision*): mengukur sejauh mana kebenaran hasil label positif yang diprediksi oleh model. Persamaan 5 untuk menghitung presisi adalah sebagai berikut:

$$presisi = \frac{TP}{(TP+FP)} \quad (5)$$

- c. *Recall* (Sensitivitas): mengukur sejauh mana model dapat mengidentifikasi semua kasus positif yang sebenarnya terjadi. Persamaan 6 untuk menghitung sensitivitas adalah sebagai berikut:

$$sensitifitas = \frac{TP}{(TP+FN)} \quad (6)$$

- d. *F1-Score*: perpaduan antara presisi (*precision*) dan *recall* menjadi satu angka tunggal. Persamaan 7 untuk menghitung *F1-score* adalah sebagai berikut:

$$F1 - Score = \frac{2 \times Presisi \times Recall}{Presisi+Recall} \quad (7)$$

## 2.9 TensorFlow Lite

*TensorFlow Lite* adalah versi ringan (*lightweight*) dari kerangka kerja *TensorFlow* yaitu sebuah *framework deep learning* yang dikembangkan oleh *google*. *Tensorflow Lite* adalah *library* yang dirancang agar model yang telah dilatih dapat diterapkan di *smartphone*, perangkat IoT, dan perangkat yang tidak memiliki daya komputasi yang tinggi. *Tensorflow Lite* mampu dijalankan jauh lebih cepat pada perangkat seluler (Alsing, 2018).

*TensorFlow Lite* akan melakukan konversi model *TensorFlow* menjadi format yang lebih ringan dengan menghapus komponen-komponen yang tidak diperlukan dan mengoptimalkan model untuk berjalan lebih cepat.

## 2.10 Android

Android adalah sistem operasi yang dikembangkan oleh *google* pada tahun 2007. Android berjalan di kernel Linux yang menyediakan lingkungan perangkat lunak yang kuat. Android menjadi pilihan dalam membuat aplikasi berbasis *java* karena sifatnya yang *open source* (Goggin, 2012). Android menjadi sistem operasi yang paling banyak digunakan di dunia untuk perangkat seluler, seperti *smartphone* dan tablet.