

**KLASIFIKASI JENIS PENYAKIT LUKA BAKAR MENGGUNAKAN
METODE FASTER REGION CONVOLUTIONAL
NEURAL NETWORK (FASTER R-CNN)**

SKRIPSI

JHUAN AVRYGANDA SITORUS

191402128



**PROGRAM STUDI S1 TEKNOLOGI INFORMASI
FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI
UNIVERSITAS SUMATERA UTARA
MEDAN
2024**

**KLASIFIKASI JENIS PENYAKIT LUKA BAKAR MENGGUNAKAN
METODE FASTER REGION CONVOLUTIONAL
NEURAL NETWORK (FASTER R-CNN)**

**Diajukan untuk melengkapi tugas dan memenuhi syarat memperoleh ijazah
Sarjana Teknologi Informasi**

JHUAN AVRYGANDA SITORUS

191402128



**PROGRAM STUDI S1 TEKNOLOGI INFORMASI
FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI
UNIVERSITAS SUMATERA UTARA**

MEDAN

2024

Persetujuan

Judul : **Klasifikasi Jenis Penyakit Luka Bakar Menggunakan Metode Faster Region Convolutional Neural Network (Faster R-CNN)**

Kategori : **Skripsi**

Nama Mahasiswa : **Jhuan Avryganda Sitorus**

Nomor Induk Mahasiswa : **191402128**

Program Studi : **Sarjana (S-1) Teknologi Informasi**

Fakultas : **Ilmu Komputer dan Teknologi Informasi**
Universitas Sumatera Utara

Medan, 4 Juli 2024

Komisi Pembimbing:

Pembimbing 2,



Ulfi Andayani S.Kom., M.Kom
NIP. 198604192015042004

Pembimbing 1,



Dr. Erna Budhiarti Nababan M.IT
NIP. 196210262017042001

Diketahui/disetujui oleh
Program Studi S-1 Teknologi Informasi
Ketua,



Dedy Arisandi S.T., M.Kom.
NIP. 197908312009121002

PERNYATAAN

**KLASIFIKASI JENIS PENYAKIT LUKA BAKAR MENGGUNAKAN METODE
FASTER REGION CONVOLUTIONAL NEURAL NETWORK (FASTER R-CNN)**

SKRIPSI

Saya mengakui bahwa skripsi ini hasil karya saya sendiri, kecuali beberapa kutipan dan ringkasan yang masing-masing telah disebutkan sumbernya.

Medan, Juni 2024

Jhuan Avryganda Sitorus

191402128

UCAPAN TERIMA KASIH

Penulis mengucapkan puji syukur kepada Tuhan Yang Maha Kuasa atas rahmat dan karunia-Nya, penulis dapat menyelesaikan skripsi sebagai salah satu syarat kelulusan program Sarjana dan memperoleh gelar Sarjana Komputer pada program studi S1 Teknologi Informasi pada Fakultas Ilmu Komputer dan Teknologi Informasi Universitas Sumatera Utara

Penulis skripsi juga mengucapkan terima kasih kepada semua pihak atas dukungan, bantuan, serta doa yang diberikan kepada penulis dalam menyelesaikan skripsi ini. Sehubungan dengan itu, penulis ingin mengucapkan terima kasih kepada :

1. Tuhan Yesus Kristus yang selalu memberikan rahmat, kekuatan, penguatan, serta berkat yang melimpah sehingga penulis dapat menyelesaikan skripsi ini dengan baik
2. Keluarga penulis, Bapak Leonardo Sitorus dan Ibu Jenny Purba, yang selalu mendukung serta mendoakan, dan memberikan nasehat, dorongan, dan semangat dalam proses perkuliahan hingga penyelesaian skripsi, begitu juga dengan adik penulis yang terkasih dan tersayang, Aranya Sitorus, juga saudara-saudari penulis Tobias Purba, Steven Purba, Indri Sitorus dan Vera Marpaung yang senantiasa memberikan doa dan dukungan hingga penguatan serta tempat penulis untuk bercerita dan berkeluh kesah selama penulisan skripsi
3. Bapak Prof. Dr. Muryanto Amin, S.Sos., M.Si. selaku Rektor Universitas Sumatera Utara.
4. Ibu Dr. Maya Silvi Lydia, M.Sc selaku Dekan Fasilkom-TI Universitas Sumatera Utara.
5. Bapak Dedy Arisandi, S.Kom., M.Kom. selaku Ketua program studi Teknologi Informasi Universitas Sumatera Utara
6. Ibu Dr. Erna Budhiarti Nababan, M.IT selaku Dosen Pembimbing I dan Ibu Ulfi Andayani S.Kom., M.Kom. selaku Dosen Pembimbing II yang telah bersedia membimbing, memberikan saran dan kritik yang membangun, dan memotivasi penulis dalam penelitian serta penulisan skripsi ini.
7. Ibu Sarah Purnamawati S.T., M.Sc. selaku Dosen Pembimbing I saya dan Ibu Dr. Marischa Elveny S.TI., M.Kom. selaku Dosen Pembimbing II saya yang telah memberikan kritik dan saran dalam penyempurnaan tugas akhir saya.

8. Seluruh Dosen, Staff dan Pegawai Program Studi S1 Teknologi Informasi yang telah memberikan manfaat bagi penulis serta membantu proses perkuliahan penulis
9. Teman-teman seangkatan Penulis, Sheren Alvionita Siahaan, Daniel Situmeang, Josua Ronaldo Pandiangan, Monang Limbong, Vincent Sirait, Christopher Manurung, Anri Marpaung, Brian Tarihoran, Samuel Malau, Tomy Tambunan, Geylfedra Panggabean, Gemilang Sibarani, Mahdan Hasibuan, Anggi Yohanes Pardede, Sebastian Sitorus, Zikri Ihsan, Aulia Rahman Sihite, Denaya Pramudia, Farisal Lazuardy, dan Irsyad Dalimunte yang selalu memberi warna, pengalaman, dan Pelajaran mulai dari masa perkuliahan hingga sampai dengan selesainya skripsi ini
10. Teman-teman Angkatan 2019 selaku teman seperjuangan penulis dalam menyelesaikan perkuliahan
11. Dan kepada Kakak-Abang dan Adik-Adik lainnya yang tidak dapat saya sebutkan satu persatu yang telah memberikan dukungan serta saran dalam masa perkuliahan serta proses penyelesaian tugas akhir ini.

Penulis menyadari bahwa penulisan skripsi ini masih memiliki kekurangan. Oleh sebab itu, penulis mengharapkan kritik dan saran yang dapat membangun demi penyempurnaan skripsi ini

Medan, Juni

Penulis

ABSTRAK

Luka bakar merupakan salah satu jenis dari cedera pada kulit yang disebabkan oleh panas baik dari api, terkena bahan kimia, sengatan listrik hingga radiasi matahari. Luka bakar juga merupakan salah satu jenis cedera yang paling umum terjadi secara tidak sengaja. Luka Bakar merupakan salah satu jenis luka luar yang sering terjadi pada Manusia. Ada beberapa jenis luka bakar dengan penanganan yang berbeda dalam setiap jenis lukanya. Dikarenakan penanganan terkhususnya penangan pertama yang salah dalam mengobati salah satu jenis luka bakar, banyak efek samping yang terjadi setelah luka sembuh seperti kulit yang berpotensi terkena infeksi, bekas luka yang tidak wajar dan membuat luka sembuh lebih lama. Penelitian ini menghasilkan sebuah sistem yang dapat mengklasifikasikan luka bakar berdasarkan warna dan bentuk luka. Jenis dari luka bakar tersebut dibagi atas tiga tingkatan, yaitu Luka Bakar Derajat I, Derajat II dan Derajat III. Luka Bakar Derajat I pada penelitian ini ditandai dengan warna kulit belang hingga warna kulit kemerahan, Luka Bakar Derajat II ditandai dengan kulit berwarna kemerahan hingga berwarna putih, lebih berstektur, dan Luka Bakar Derajat III ditandai dengan warna kulit berwarna hitam dengan tekstur yang lebih kasar serta area kerusakan yang lebih besar. Adapun data yang digunakan pada penelitian ini berjumlah 1250 data yang kemudian dibagi menjadi 1000 data *training* 125, data *validasi* serta 125 data *testing*. Setelah dilakukan pengujian, penelitian ini menghasilkan akurasi sebesar 95,20% .

Kata Kunci : Luka Bakar, Derajat Luka Bakar, Citra Digital, *Faster RCNN*

CLASSIFICATION OF TYPES OF BURNS USING FASTER REGION CONVOLUTIONAL NEURAL NETWORK

ABSTRACT

Burns are one type of injury to the skin caused by heat either from fire, exposure to chemicals, electric shock to solar radiation. Burns are also one of the most common types of injuries that occur accidentally. Burns are one type of external injury that often occurs in humans. There are several types of burns with different treatments in each type of wound. Due to treatment, especially the wrong first handler in treating one type of burn, many side effects occur after the wound heals such as skin that has the potential for infection, unnatural scars and makes the wound heal longer. This research resulted in a system that can classify burns based on the color and shape of the wound. The types of burns are divided into three levels, namely Degree I, Degree II and Degree III Burns. Grade I Burns in this study were characterized by striped skin color to reddish skin color, Grade II Burns were characterized by reddish to white, more sculptural skin, and Grade III Burns were characterized by black skin color with a rougher texture and a larger area of damage. The data used in this study amounted to 1250 data which were then divided into 1000 training data, 125 validation data, and 125 testing data. After testing, this study resulted in an accuracy of 95.20%.

Keywords : Burns Wound, Burns Wound Degree, Digital Image, Faster R-CNN

DAFTAR ISI

PERSETUJUAN	iii
PERNYATAAN	iv
UCAPAN TERIMA KASIH.....	v
ABSTRAK.....	vii
ABSTRACT.....	viii
DAFTAR ISI.....	ix
DAFTAR TABEL.....	xi
DAFTAR GAMBAR.....	xii
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah.....	2
1.4 Tujuan Penelitian.....	3
1.5 Manfaat Penelitian.....	3
1.6 Metodologi Penelitian	3
1.7 Sistematika Penulisan.....	4
BAB II LANDASAN TEORI.....	5
2.1 Luka.....	5
2.2 Luka Bakar	6
2.3 Pengolahan Citra Digital (Digital Image Processing)	8
2.4 Citra Digital.....	8
2.5 <i>Convolutional Neural Network</i> (CNN)	10
2.6 <i>Region Convolutional Neural Network</i> (R-CNN)	12
2.7 <i>Fast Region Convolutional Neural Network</i> (Fast R-CNN)	12
2.8 <i>Faster Region Convolutional Neural Network</i> (Faster R-CNN).....	13
2.9 <i>Android Studio</i>	15
2.10 <i>Python</i>	16
2.11 <i>Tensorflow Lite</i>	16
2.12 <i>Keras</i>	17
2.13 <i>Confussion Matrix</i>	17
2.14 Penelitian Terdahulu	19
2.15 Perbedaan Penelitian	22

BAB III ANALISIS DAN PERANCANGAN SISTEM.....	23
3.1 Dataset	23
3.2 Analisis Sistem	24
3.3 Perancangan Antarmuka Sistem.....	41
3.3.5 Rancangan tampilan Halaman <i>Result</i>	44
BAB IV PENGUJIAN SISTEM	46
4.1 Implementasi Sistem	46
4.2 Prosedur Operasional	52
4.3 Pelatihan Sistem	52
4.4 Pengujian Sistem	56
BAB V KESIMPULAN DAN SARAN.....	64
5.1 Kesimpulan.....	64
5.2 Saran.....	64
DAFTAR PUSTAKA.....	65

DAFTAR TABEL

Tabel 2. 1 Confussion Matrix.....	17
Tabel 2. 2 Penelitian Terdahulu	21
Tabel 3. 1 Contoh Data Citra Luka Bakar.....	23
Tabel 3. 2 Jumlah Data Citra Derajat Luka Bakar	24
Tabel 4. 1 Hasil Pelatihan Sistem.....	54
Tabel 4. 2 Hasil Pengujian Sistem	56
Tabel 4. 3 Hasil Evaluasi Model Menggunakan Confussion Matrix	60
Tabel 4. 4 Nilai TP, FP dan FN Citra Luka Bakar	61
Tabel 4. 5 Nilai Precission, Recall dan F1-Score.....	63

DAFTAR GAMBAR

Gambar 2. 1 Luka Terbuka (Sumber : https://en.wagner-healthcare.com/)	5
Gambar 2. 2 Luka Tertutup (Sumber : https://www.klinikindonesia.com).....	5
Gambar 2. 3 Luka Bakar derajat I	7
Gambar 2. 4 Luka Bakar Derajat II	7
Gambar 2. 5 Luka Bakar Derajat III (Morandi, M.E, et al ., 2017)	7
Gambar 2. 6 Citra RGB (Andono et al., 2017).....	9
Gambar 2. 7 <i>Citra Grayscale</i> (Andono et al., 2017)	9
Gambar 2. 8 Citra Biner (Andono et al., 2017)	10
Gambar 2. 9 Arsitektur Umum Convolutional Neural Network	11
Gambar 2. 10 Arsitektur R-CNN	12
Gambar 2. 11 Arsitektur Fast R-CNN	13
Gambar 2. 12 Arsitektur Faster R-CNN (Ghoury, S et al., 2019).....	15
Gambar 2. 13 Android Studio (Sumber niagahoster.co.id)	16
Gambar 2. 14 Tensorflow Lite (Sumber researchgate.net)	17
Gambar 3. 1 Arsitektur Umum	25
Gambar 3. 2 Contoh data yang digunakan	26
Gambar 3. 3 Labelling	27
Gambar 3. 4 <i>Pseudocode Resizing</i>	27
Gambar 3. 5 Proses Perhitungan <i>Resizing</i>	28
Gambar 3. 6 Nilai Matrix dari proses <i>Resizing</i>	28
Gambar 3. 7 Citra Luka Bakar sebelum (kiri) dan sesudah di <i>resize</i> (kanan).....	29
Gambar 3. 8 <i>Pseudocode</i> Augmentasi	29
Gambar 3. 9 Citra luka sebelum (kiri) dan sesudah <i>flip horizontal</i> (kanan)	30
Gambar 3. 10 Citra luka sebelum (kiri) dan sesudah <i>flip vertical</i> (kanan).....	30
Gambar 3. 11 Citra luka sebelum (kiri) dan sesudah <i>rotate 90°</i> (kanan)	30
Gambar 3. 12 Matrix Input pada Gambar	31
Gambar 3. 13 <i>Pseudocode</i> proses Klasifikasi	34
Gambar 3. 14 <i>Feature Map</i>	35
Gambar 3. 15 <i>Region Proposal Network</i>	36
Gambar 3. 16 Target <i>Region of Interest (RoI) Pooling</i>	37
Gambar 3. 17 Pemetaan <i>Pooling</i> dan Lapisan <i>Pooling</i>	37

Gambar 3. 18 Pemetaan <i>Pooling</i> dan Lapisan <i>Pooling</i>	38
Gambar 3. 19 Pemetaan <i>Pooling</i>	38
Gambar 3. 20 Proses <i>Pooling Data</i>	38
Gambar 3. 21 <i>RoI Pooling 3x3 (Full Size)</i>	39
Gambar 3. 22 <i>Pseudocode Tensorflow Lite</i>	41
Gambar 3. 23 <i>Splash Screen</i>	42
Gambar 3. 24 <i>Main Page</i>	42
Gambar 3. 25 <i>Article Page</i>	43
Gambar 3. 26 <i>Testing Page</i>	43
Gambar 3. 27 <i>Result Page</i>	44
Gambar 4. 1 Data Gambar Bukan Luka Bakar	47
Gambar 4. 2 Data Gambar Luka Bakar Derajat I.....	47
Gambar 4. 3 Data Gambar Luka Bakar Derajat II.....	47
Gambar 4. 4 Data Gambar Luka Bakar Derajat III	48
Gambar 4. 5 <i>Splashscreen</i>	48
Gambar 4. 6 Tampilan Utama	49
Gambar 4. 7 Tampilan Halaman Artikel	50
Gambar 4. 8 Tampilan Halaman Klasifikasi	51
Gambar 4. 9 Tampilan Halaman Hasil	51

BAB I

PENDAHULUAN

1.1 Latar Belakang

Luka bakar merupakan salah satu jenis dari cedera pada kulit yang disebabkan oleh panas baik dari api, terkena bahan kimia, sengatan listrik hingga radiasi matahari. Luka bakar juga merupakan salah satu jenis cedera yang paling umum terjadi secara tidak sengaja.

Luka bakar sendiri dibedakan menurut derajat dari luka bakar tersebut. Pada derajat satu, gejala yang terjadi ialah kemerahan pada kulit hingga nyeri. Ditahap ini, ciri-ciri yang ditunjukkan ialah kulit berwarna merah, kering dan terasa sakit. Pada derajat dua, gejala yang terjadi ialah kulit lecet, melepuh, bengkak serta menimbulkan nyeri yang hebat. Kerusakan yang ditimbulkan pada jenis luka ini meliputi area epidermis dan lapisan atas dermis pada kulit. Pada derajat ketiga, gejala yang ditunjukkan ialah luka tampak putih, terdapat bagian hangus, kasar serta mati rasa. Pada derajat III, kerusakan yang ditimbulkan dari Luka Bakar tersebut bersifat Permanen (Anggowarsito, 2014). Penentuan derajat luka bakar dilihat juga dari warna, lebar dan bentuk dari luka tersebut. Semakin gelap, lebar dan semakin dalam lukanya maka derajat luka tersebut semakin naik.

Penelitian terdahulu yang pernah dilakukan terkait pada penyakit luka bakar adalah penelitian terdahulu tentang klasifikasi jenis luka bakar menggunakan CNN berdasarkan bentuk dan jenis luka bakar yang dialami. Data yang digunakan adalah data standar dari luka bakar yang diklasifikasikan kedalam 2 grup. Dengan menggunakan pendekatan CNN, didapat hasil akurasi klasifikasi sebesar 93% (Ashish, 2021).

Selanjutnya terdapat juga penelitian yang mengklasifikasikan Penyakit dengan menggunakan Metode Faster R-CNN seperti penelitian yang dilakukan oleh Hartanto & Wibowo (2020) tentang *Development Of Mobile Skin Cancer Detection Using Faster-R CNN and MobileNet v2 Model* dimana didapatkan akurasi sebesar 85% dengan nilai learning rate sebesar 0.005 untuk 30.000 step. Lalu ada juga penelitian dengan menggunakan metode yang sama yang dilakukan oleh R.Ezhilarasi & P.Varalakshmi tentang *Tumor Detection in the Brain using Faster R-CNN* dengan akurasi sebesar 99% dalam mendeteksi Tumor yang terdapat pada otak. Lalu ada juga penelitian yang

dilakukan oleh Yu Yang & Yang Qun tentang “*An Improved Faster R-CNN for Colorectal Cancer Cell Detection*” dengan menggunakan sekitar 1000 gambar sebagai *training* dan 1000 gambar sebagai *testing* dengan akurasi sebesar 96,4%.

Berdasarkan latar belakang serta hasil penelitian-penelitian terdahulu yang telah disebutkan, maka peneliti tertarik untuk melakukan penelitian Tugas Akhir dengan judul “Klasifikasi Jenis Penyakit Luka Bakar Berdasarkan Warna dan Bentuk Luka pada Manusia dengan menggunakan Algoritma Faster R-CNN” sebagai solusi untuk mengklasifikasi Jenis luka bakar dengan metode Faster R-CNN untuk mendeteksi objek dan memprediksi jenis luka dari gambar yang ditampilkan

1.2 Rumusan Masalah

Luka bakar merupakan salah satu jenis penyakit luka pada tubuh manusia yang sering terjadi secara tidak sengaja dan disebabkan oleh terkena panas, terpapar bahan kimia, dsb. Ada beberapa jenis luka bakar dengan penanganan yang berbeda dalam setiap jenis lukanya. Dikarenakan penanganan terkhususnya penangan pertama yang salah dalam mengobati salah satu jenis luka bakar, banyak efek samping yang terjadi setelah luka sembuh seperti kulit yang berpotensi terkena infeksi, bekas luka yang tidak wajar dan membuat luka sembuh lebih lama. Terdapat juga beberapa kendala baik dari segi orang awam ketika menilai sebuah jenis luka bakar dikarenakan memiliki ciri khas yang mirip terkhususnya luka bakar derajat pertama dan kedua . maupun luka derajat kedua dengan luka derajat ketiga. Oleh karena itu, dibutuhkan sebuah sistem untuk mengklasifikasikan jenis luka bakar berdasarkan derajat luka bakar tersebut.

1.3 Batasan Masalah

Untuk mendapatkan maksud serta tujuan yang jelas dari penelitian, maka dibutuhkan batasan-batasan dalam penelitian, diantaranya :

1. *Dataset* ataupun jenis Luka bakar yang mampu diklasifikasikan menggunakan model Faster R-CNN ada 3 tingkat, diantaranya Luka Bakar Tingkat Pertama, Luka Bakar Tingkat Kedua, dan Luka Bakar Tingkat Ketiga
2. Ekstensi gambar dari citra yang akan digunakan ialah .jpg dan .jpeg
3. *Dataset* yang akan digunakan adalah data yang berasal dari Internet yang diambil menggunakan metode *Scraping Data/Extraction Data* serta data-data yang didapat dari klinik kesehatan yang ada di Kota Medan, Sumatera Utara.

1.4 Tujuan Penelitian

Adapun tujuan dari penelitian ini adalah mengimplementasikan Faster R-CNN untuk mengklasifikasikan Jenis Luka Bakar berdasarkan Warna dan Bentuk Luka Bakar

1.5 Manfaat Penelitian

Adapun manfaat-manfaat yang nantinya diharapkan dari penelitian ini adalah :

1. Membantu Pengguna untuk mengenali derajat luka bakar pada kelas tertentu
2. Sebagai media pembelajaran bagi orang awam untuk mengenali ciri pada luka bakar dengan derajat tertentu
3. Menguji kemampuan komputer dalam mengklasifikasikan Citra Luka Bakar.
4. Mencari Model terbaik dalam mengklasifikasikan Citra Luka Bakar.
5. Menjadi referensi untuk pengembangan penelitian-penelitian selanjutnya terkhususnya bidang Kesehatan mengenai citra digital dengan menggunakan metode Faster R-CNN.

1.6 Metodologi Penelitian

Tahapan-tahapan yang akan dilakukan peneliti dalam penelitian ini adalah sebagai berikut :

1. Studi Literatur

Pada Tahap ini dilakukan studi literatur dengan mengumpulkan beberapa bahan referensi dari Jurnal, Artikel, Website, serta sumber lainnya mengenai , *Image Processing, Faster-R CNN, Skin Burn Type*.

2. Analisis Permasalahan

Pada tahap ini dilakukan Analisis terhadap tahapan yang telah dilakukan sebelumnya pada Studi Literatur untuk mendapatkan pemahaman mengenai *Skin Burn Classification Type* yang akan diimplementasikan dalam penelitian untuk membangun sistem Klasifikasi Luka Bakar

3. Perancangan Sistem

Pada tahap ini dilakukan proses perancangan sistem berdasarkan analisis permasalahan yang telah dilakukan sebelumnya. Perancangan sistem ini meliputi perancangan dan pembuatan arsitektur, pengumpulan data, serta pembagian data *training* dan data *testing*.

4. Implementasi

Pada tahap ini dilakukan implementasi dari sistem yang telah dirancang kedalam bentuk source code untuk menghasilkan sistem mesin Klasifikasi Jenis Luka Bakar

5. Pengujian Sistem

Pada tahap ini dilakukan Pengujian sistem yang telah dibangun untuk mengetahui tingkat akurasi yang didapat dari implementasi Klasifikasi Jenis Luka bakar dengan menggunakan Faster R-CNN dalam pengembangan mesin pengklasifikasi Jenis Luka Bakar pada Manusia.

6. Dokumentasi dan Penyusunan Laporan

Pada tahap ini dilakukan dokumentasi dan penyusunan laporan yang menjabarkan hasil dari penelitian yang sudah dilakukan.

1.7 Sistematika Penulisan

Sistem Penulisan Skripsi ini terdiri dari lima bagian, diantaranya :

BAB I : PENDAHULUAN

Bab ini berisikan tentang Latar Belakang, Rumusan Masalah, Batasan Masalah, Tujuan Penelitian, Manfaat Penelitian, Metodologi Penelitian, dan Sistematika Penelitian.

BAB II : LANDASAN TEORI

Bab ini berisikan tentang penjelasan dari teori-teori yang relevan dengan sistem dan permasalahan yang diangkat pada skripsi ini.

BAB III : ANALISIS DAN PERANCANGAN SISTEM

Bab ini berisi tentang arsitektur umum serta metode yang nantinya akan diterapkan dalam pembuatan sistem

BAB IV : IMPLEMENTASI DAN PENGUJIAN

Bab Ini berisi tentang implementasi atau penerapan sistem berdasarkan analisis dan rancangan sistem yang telah dibuat. Bab ini juga berisikan tentang tampilan dari sistem yang telah dibangun beserta pengujian untuk mengetahui akurasi sistem dalam menentukan jenis luka bakar beserta cara menangannya.

BAB V : KESIMPULAN DAN SARAN

Bab ini berisikan tentang kesimpulan dari keseluruhan isi penelitian yang telah dilakukan serta saran yang diajukan untuk pengembangan penelitian selanjutnya.

BAB II

LANDASAN TEORI

2.1 Luka

Luka adalah kerusakan pada fungsi perlindungan kulit yang disertai hilangnya kontinuitas jaringan epitel dengan atau tanpa adanya kerusakan pada jaringan lainnya seperti pada bagian otot, tulang dan nervus (saraf) yang disebabkan oleh beberapa faktor seperti tekanan, sayatan, dan luka karena operasi (Ryan, 2014). Luka terklasifikasi atas 2 jenis diantaranya Luka Terbuka dan Luka Tertutup. Luka terbuka merupakan luka yang terpapar oleh udara yang dikarenakan adanya kerusakan pada kulit baik tanpa maupun disertai kerusakan jaringan di bawahnya. Adapun jenis luka terbuka diantaranya Luka Lecet, Luka Iris, Luka Robek, Luka Tusuk, Luka dikarenakan Gigitan, serta Luka Bakar. Sedangkan Luka tertutup merupakan cedera pada jaringan dimana kulit masih utuh/tidak nampak luka dipermukaan kulit seperti Luka Memar dan Hematoma (Penggumpalan darah)



Gambar 2. 1 Luka Terbuka (Sumber : <https://en.wagner-healthcare.com/>)



Gambar 2. 2 Luka Tertutup (Sumber : <https://www.klinikindonesia.com>)

2.2 Luka Bakar

Luka bakar merupakan salah satu jenis luka pada tubuh yang disebabkan oleh hantaran benda panas kepada kulit tubuh. Luka bakar biasa terjadi dikarenakan banyak penyebab seperti paparan sinar matahari, cairan zat kimia, listrik maupun api. Adapun efek yang ditimbulkan dari luka bakar ialah kulit terasa sangat nyeri, kulit berwarna merah dan mengelupas hingga menyebabkan terjadinya luka, pembengkakan serta perubahan warna yang terdapat pada kulit. Nyeri yang disebabkan juga tidak selamanya berbanding lurus dengan tingkat keparahan luka bakar. Semakin tinggi derajat keparahan luka bakar sering sekali tidak terasa nyeri pada bagian luka tersebut.

Dilihat dari lebar, bentuk dan warna luka, Luka bakar terbagi atas 3 derajat diantaranya :

1. Luka Bakar Derajat 1
2. Luka Bakar Derajat 2
3. Luka Bakar Derajat 3

Perbedaan antara ketiga jenis derajat luka bakar ialah dari seberapa luas lukanya, seberapa dalam lukanya hingga warna luka. Pada derajat luka bakar pertama, bentuk luka yang ditimbulkan berukuran kecil dan luka hanya menyerang bagian lapisan luar kulit (epidermis) serta warna luka yang cenderung kemerahan hingga putih penyebab dari luka bakar ini ialah paparan sinar matahari ataupun cedera ringan seperti percikan api (Saputra, D. 2023). Pada derajat luka bakar tingkat kedua, bentuk luka yang ditimbulkan berukuran sedang dengan ciri luka berwarna merah hingga berwarna putih. Jika dibiarkan, luka akan mengalami pembengkakan serta luka lepuh kulit yang melebar sehingga menimbulkan rasa sakit yang parah. Butuh waktu yang cukup lama dalam penyembuhan luka ini (Evers, LH *et*, 2013). Sedangkan pada luka bakar tingkat 3, luka cenderung berwarna putih-hitam. Lebar luka diatas 10 cm. Pada fase ini kulit akan kelihatan lebih kasar. Efek samping dari luka bakar tingkat ketiga ini ialah menyebabkan mati rasa pada bagian yang luka dan akan meninggalkan bekas pada luka ketika luka sudah kering (Gupta, N 2018). Adapun ketiga luka bakar ini memiliki penanganan pertama yang berbeda-beda sesuai dengan derajat tingkat luka bakar tersebut. Semakin tinggi derajat luka, semakin khusus juga pertolongan pertama dalam penanganan luka. Perbedaan dari ketiga derajat luka dapat terlihat lebih jelas pada gambar 2.1, 2.2, dan 2.3.



Gambar 2. 3 Luka Bakar derajat I

(Sumber : <https://www.jems.com/patient-care/basics-burn-management/>)



Gambar 2. 4 Luka Bakar Derajat II

(Sumber : <https://www.jems.com/patient-care/basics-burn-management/>)



Gambar 2. 5 Luka Bakar Derajat III (Morandi, M.E, et al ., 2017)

2.3 Pengolahan Citra Digital (Digital Image Processing)

Pengolahan Citra digital adalah salah satu tahap bentuk pemrosesan informasi dengan input berupa citra (image) dan output berupa citra atau juga bagian dari citra tersebut. Tujuan dalam proses ini adalah memperbaiki kualitas citra agar dipermudah dalam interpretasi oleh manusia maupun mesin komputer. Pengolahan citra digital merupakan istilah yang umum untuk berbagai teknik yang keberadaannya adalah untuk memanipulasi dan memodifikasi citra dengan menggunakan berbagai cara. (Efford, 2003).

2.4 Citra Digital

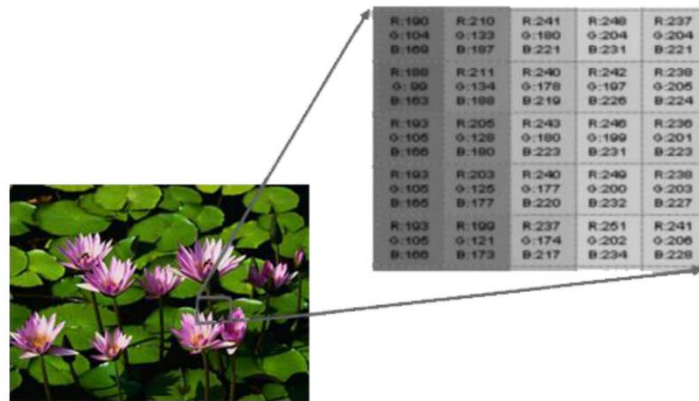
Citra atau image adalah representasi spasial dari suatu objek yang sebenarnya dalam bidang dua dimensi yang biasanya ditulis dalam koordinat cartesian x-y, dan setiap koordinat merepresentasikan satu sinyal terkecil dari objek (Kulkarni, 2001).

Citra digital dapat diartikan sebagai fungsi dua dimensi, $F(x,y)$. Dimana fungsi x dan y adalah koordinat spasial, dan Amplitudo F pada setiap pasangan koordinat (x,y) disebut intensitas dari gambar. Pada titik tersebut. Ketika x , y dan nilai amplitudo F terbatas, maka kita dapat menyimpulkan sebagai citra digital.

Berdasarkan kombinasi warna yang terdapat pada pixel, citra digital terbagi atas 3 jenis yaitu citra RGB, Citra Grayscale dan Citra Biner.

2.4.1 Citra RGB

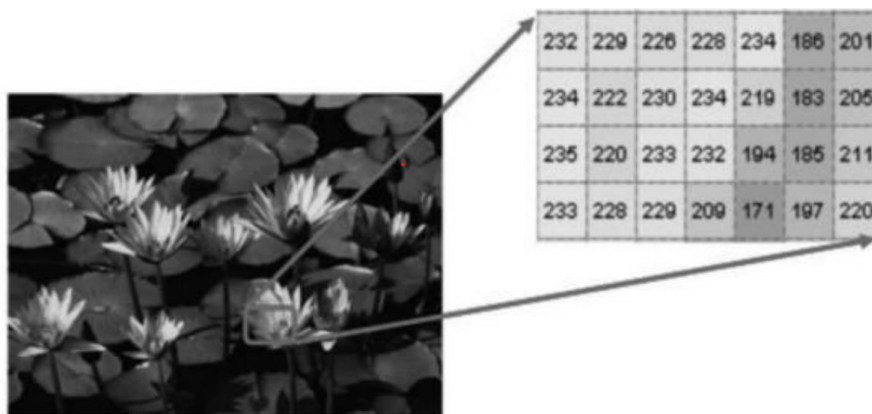
Citra RGB (Bewarna) merupakan citra yang menyajikan warna dalam bentuk komponen R (Merah), G (Hijau), dan B (Biru). Setiap komponen dari ketiga warna tersebut menggunakan 8 bit dimana setiap komponen dari 8 bit tersebut memiliki nilai berkisar antara 0 sampai 255. Dengan begitu, kemungkinan nilai warna yang dapat disajikan citra RGB dapat mencapai $255 \times 255 \times 255$ kombinasi warna atau 16.581.375 kombinasi warna yang berbeda. jika sebuah piksel memiliki nilai RGB (255, 0, 0), itu berarti piksel tersebut adalah warna merah murni, sedangkan nilai (0, 255, 0) akan merepresentasikan warna hijau murni. RGB sendiri didasarkan pada teori bahwa mata manusia peka terhadap panjang gelombang 630nm (merah), 530 nm (hijau), dan 450 nm (biru).



Gambar 2. 6 Citra RGB (Andono et al., 2017)

2.4.2 Citra *Grayscale*

Citra *Grayscale* merupakan citra digital yang hanya memiliki satu nilai warna pada setiap pixelnya, yang berarti nilai warna dari *Red* = *Green* = *Blue*. Nilai-nilai dari warna tersebut digunakan untuk menunjukkan intensitas warna. Citra yang ditampilkan dari jenis ini terdiri atas warna abu-abu, yang divariasikan dengan warna hitam sebagai intensitas terlemah dan warna putih sebagai intensitas terkuat. Citra *Grayscale* sendiri memiliki perbedaan terhadap citra hitam-putih, dimana citra hitam putih hanya memiliki variasi warna hitam dan putih saja sedangkan pada citra *grayscale* variasi warna yang digunakan berupa hitam-putih dengan variasi warna yang sangat banyak. Citra *grayscale* sendiri disimpan dalam format 8 bit untuk setiap sampel pixel, dengan kemungkinan sebanyak 256 intensitas. Untuk mengubah citra berwarna yang dari berbentuk R, G, B menjadi citra *Grayscale* dengan nilai X, konversi dilakukan dengan mengambil nilai rata-rata dari ketiga citra warna sehingga dapat dituliskan menjadi : $X = (R+G+B)/3$, dengan Warna = RGB(X,X,X)

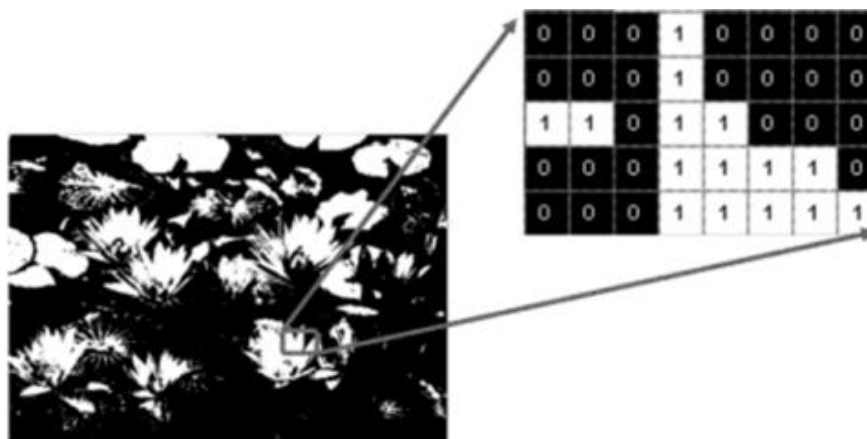


Gambar 2. 7 Citra *Grayscale* (Andono et al., 2017)

2.4.3 Citra Biner

Sesuai dengan namanya, citra biner sendiri hanya memiliki 2 kemungkinan spektrum warna yaitu hanya hitam dan putih. Citra biner sendiri disebut juga dengan W&B (*White & Black*) atau citra monokrom. Pada Citra Biner sendiri hanya dibutuhkan 1 bit untuk mewakili nilai setiap piksel citra biner.

Pembuatan Citra Biner sendiri juga memerlukan batas nilai keabuan yang akan dijadikan sebagai patokan. Piksel dengan derajat keabuan lebih besar akan diberi nilai 1 dan derajat keabuan kecil akan diberi nilai 0. Citra biner sendiri biasanya muncul dari proses pengolahan gambar seperti segmentasi, pengembangan, morfologi ataupun *dithering*. Citra Biner sendiri memiliki fungsi untuk mempermudah proses pengenalan pola warna, karena pola akan lebih mudah terdeteksi pada citra yang mengandung sedikit warna.



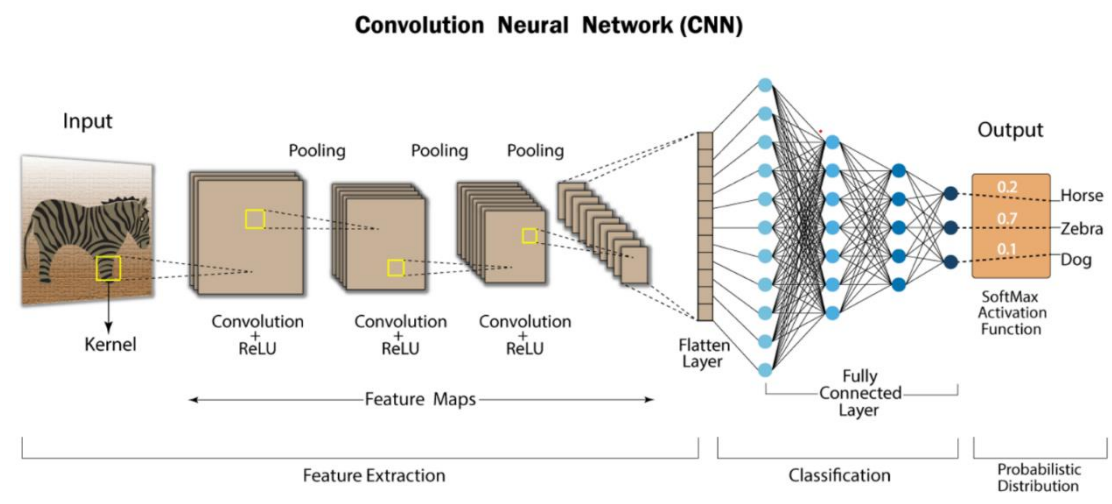
Gambar 2. 8 Citra Biner (Andono et al., 2017)

2.5 Convolutional Neural Network (CNN)

Convolutional Neural Network (CNN) merupakan salah satu kelas yang terdapat pada deep learning dimana mampu melakukan identifikasi serta klasifikasi dari sebuah gambar. Metode CNN sendiri sering juga digunakan dalam mengidentifikasi wajah, menganalisis isi dokumen, mengklasifikasikan wajah, mengklasifikasikan gambar, dsb.

Metode CNN sendiri mengklasifikasikan jenis gambar dengan memproses gambar yang diinput kedalam, lalu diklasifikasikan ke kategori tertentu seperti pada citra hewan peliharaan, dimana terdapat bagian bulu, warna, bentuk wajah, mata, kaki, dsb. Semua gambar akan dibuat menjadi bentuk array yang berisikan nilai pada setiap pixel dengan resolusi tinggi*panjang*dimensi yang disebut dengan channel. Channel

disini terdiri atas 3 jenis warna yang mewakili warna RGB dengan masing-masing lapisan mengimplementasikan *Red-Green-Blue* ataupun 1 lapisan jika gambar berbentuk grayscale. Namun, jumlah lapisan juga bisa melebihi 3 lapisan, bahkan hingga ratusan yang merepresentasikan berbagai jenis warna lainnya dengan Arsitektur RGR



Gambar 2. 9 Arsitektur Umum *Convolutional Neural Network*

(Sumber : www.developersbreach.com/convolution-neural-network-deep-learning)

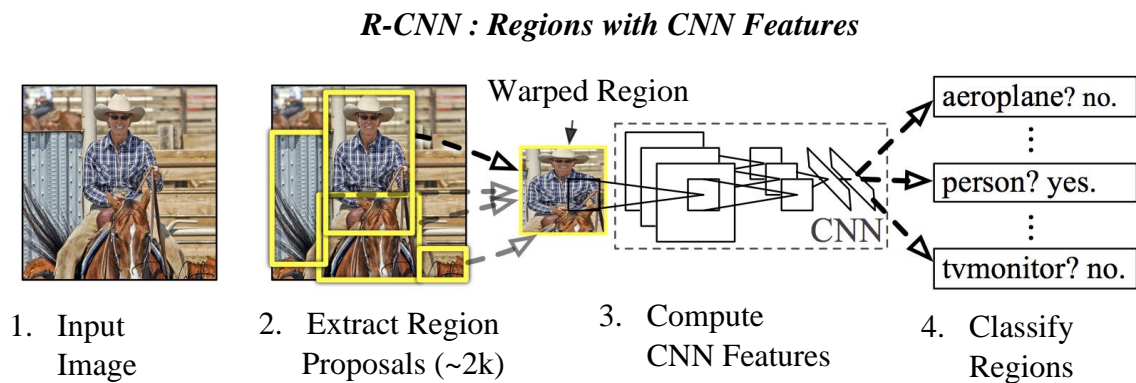
Seperti pada gambar 2.9, arsitektur CNN terbagi atas 2 bagian yaitu *Feature Extraction Layer* dan *Fully Connected Layer*. Masing-masing layer memiliki fungsi dan kegunaannya masing-masing. Pada *Feature Extraction Layer* algoritma akan menjalankan rangkaian operasi convolution dan pooling yang akan digunakan untuk membuat Feature Map yang berisikan angka numerik sebagai representasi dari citra tersebut. *Convolutional Layer* adalah lapisan pertama yang mengekstraksi ciri dari citra yang masuk. Pada proses Konvolusi, *Convolution Layer* berguna untuk menjaga hubungan antara *pixel* dengan mempelajari ciri citra menggunakan operasi matematis antara matriks citra dengan filter atau *kernel*. *Kernel* adalah sebuah operator yang diterapkan ke seluruh citra guna mendapat nilai dari *array* pada sebuah citra. Kernel biasanya memiliki ukuran matriks 3*3 hingga 5*5 yang berisikan nilai acak antara -1 dan 1.

Singkatnya, cara kerja dari Metode CNN ialah CNN akan melatih serta menguji setiap gambar melalui rangkaian proses yang dimulai dari pemecahan gambar menjadi gambar yang lebih kecil, lalu memasukkan potongan-potongan gambar kecil ke *Neural*

Network yang lebih kecil, menyimpan semua hasil dari masing-masing potongan gambar kedalam *array* baru, mengurangi ukuran spasial untuk mengurangi parameter dan perhitungan ketika citra memiliki ukuran yang besar, lalu membuat prediksi.

2.6 Region Convolutional Neural Network (R-CNN)

Region Convolutional Neural Network (R-CNN) merupakan salah satu Algoritma pendeteksi objek yang dibuat oleh Girshick et al., di UC Berkeley pada tahun 2014. R-CNN sendiri merupakan salah metode untuk mendeteksi objek pada *image processing* yang berbasis *deep learning*. Adapun pada R-CNN pemrosesan citra dengan skala region yang besar jauh lebih cepat dan efisien dibandingkan metode CNN. R-CNN juga memiliki algoritma yang menemukan dua buah wilayah termirip dan digabungkan bersama yang disebut juga *selective search*.



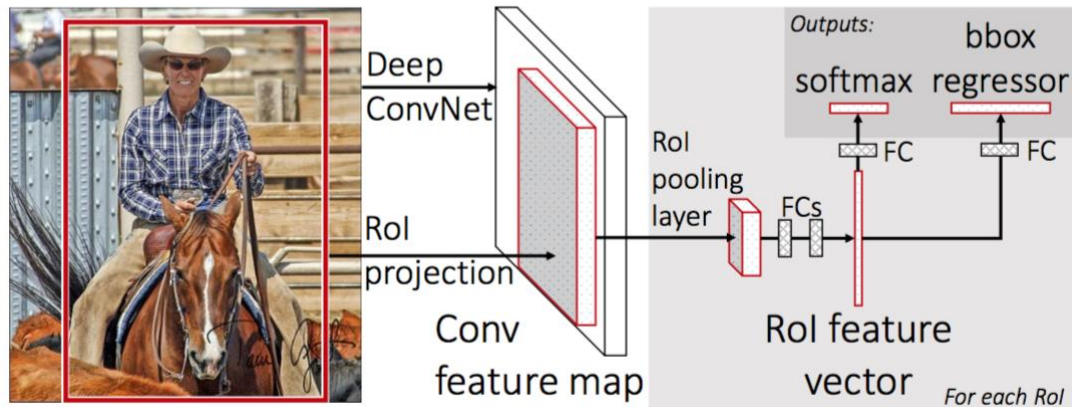
Gambar 2. 10 Arsitektur R-CNN

(Sumber : geekforgeeks.com)

2.7 Fast Region Convolutional Neural Network (Fast R-CNN)

Fast R-CNN merupakan salah satu algoritma dari bentuk penyempurnaan algoritma R-CNN. Metode ini dikembangkan oleh Ross Gershick pada tahun 2015. Implementasi dari Fast R-CNN sama dengan R-CNN, namun pada Fast R-CNN gambar tidak diinput kedalam proposal region CNN, melainkan langsung diinput kedalam CNN untuk menghasilkan *Convolutional Feature Map*. Dari *Convolutional Feature Map*, lalu diidentifikasi Region Proposal dan dikelompokkan menggunakan *RoI Pooling Layer* dan diubah bentukannya menjadi bentuk tetap yang nantinya bisa diinput kedalam *Fully Connected Layer*.

Singkatnya, Fast R-CNN jauh lebih cepat dibandingkan R-CNN ialah dikarenakan kita tidak perlu memasukkan 2000 Region proposal kedalam CNN setiap waktu. Melainkan, *Convolutional Operation* dilakukan pada setiap gambar dan *Feature Map* dihasilkan dari operasi tersebut.



Gambar 2. 11 Arsitektur Fast R-CNN

(Sumber : geeksforgeeks.com)

2.8 *Faster Region Convolutional Neural Network (Faster R-CNN)*

Faster Region Convolutional Neural Network (Faster R-CNN) merupakan salah satu algoritma dari bentuk penyempurnaan Algoritma Fast R-CNN. Faster R-CNN dibuat berdasarkan algoritma yang lama dengan tujuan klasifikasi *Deep Learning* yang jauh lebih efisien. Dibandingkan dengan Fast R-CNN, Faster R-CNN menggunakan *Region Proposal Network* (RPN) dan tidak memerlukan metode external untuk kandidat *Region Proposal*. Baik Algoritma R-CNN dan Fast R-CNN membutuhkan waktu untuk mengetahui region proposal dengan penggunaan selective search, sedangkan pada Faster R-CNN tidak menggunakan Algoritma *Selective Search* dan jaringan dapat mempelajari proposal wilayahnya sendiri.

Metode Faster R-CNN terdiri atas 2 tahap yaitu tahap ekstraksi dan melatih dengan menggunakan *Region Proposal Network* (RPN) dan mengklasifikasikan seluruh fitur yang telah diperoleh.. Pada proses training data, terdapat 4 tahapan untuk melatih data diantaranya : a) Melatih jaringan R-CNN dengan RPN dan hasil akan diklasifikasikan sesuai dengan kategori, b) Jaringan RPN yang telah dilatih akan diinput kedalam RoI Pooling Layer lalu diinput kedalam *Fully Connected Layer*, c)Setelah itu data yang telah

diklasifikasi akan diekstraksi dengan bantuan *pre-trained* model seperti VGG16, VGG50, maupun Resnet50 yang berfungsi sebagai *backbone* untuk menghasilkan RPN yang lebih sempurna, d) Lalu RPN yang telah disempurnakan digunakan kembali untuk menyempurnakan FC Layer dari *Detection Network* (Dedhia, P.R, 2020).

Pada metode Faster R-CNN, citra akan diinput lalu diproses pada *layer* CNN. Proses konvolusi biasanya menggunakan *pre-trained* model *Backbone Network* seperti Resnet dan VGG. Output dari proses tersebut adalah *Feature Map* yang menyajikan fitur peta konvolusional. Hasil dari Proposal wilayah akan diproses dengan *Region Proposal Network* (RPN) menggunakan anchor point dimana anchor point sendiri memiliki fungsi untuk mengklasifikasikan lebih awal apakah gambar dapat dikatakan objek maupun bukan objek (Gambar Latar). Setelah diproses pada RPN, Proposal Wilayah akan diprediksi dan dibentuk ulang dengan menggunakan *RoI Pooling Layer* dimana RoI nantinya akan mengklasifikasikan gambar dalam *Region Proposal* serta memprediksi nilai offset pada *Bounding Box*.

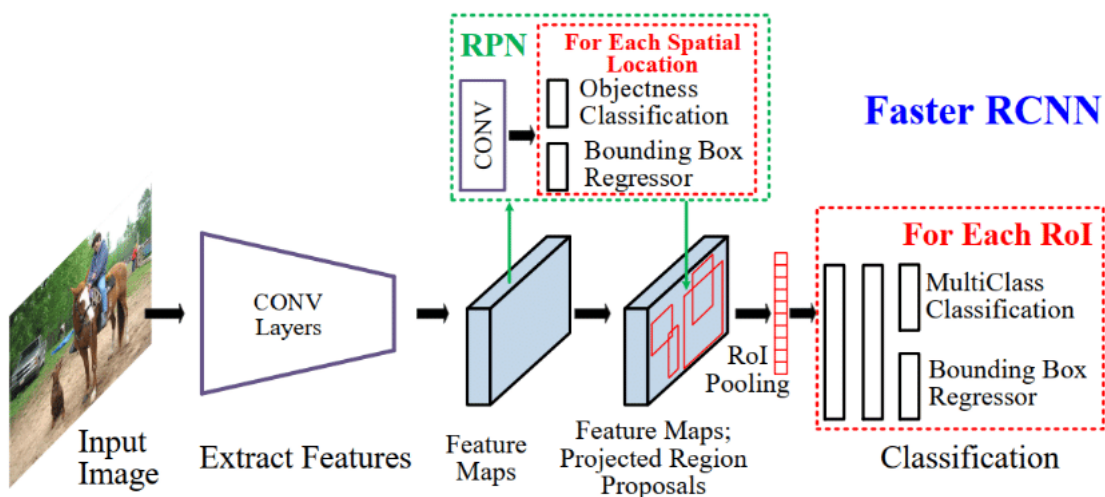
Faster R-CNN sendiri mengubah *Selective Search Method* yang terdapat pada Fast R-CNN dengan *Region Proposal Network* (RPN) yang menyebabkan algoritma bekerja lebih cepat, dengan kecepatan pengolahan gambar sebesar 0,2 detik per gambarnya. Pendeteksian dilakukan dengan mencari ciri dari objek pada citra melalui proses konvolusi atau CNN. Faster R-CNN sendiri memiliki 3 fungsi utama diantaranya mengklasifikasikan dan menghasilkan *feature map* dengan menggunakan *Deep Neural Network*, menghasilkan proposal wilayah dengan menggunakan *Region Proposal Network*, dan menemukan regresi dan menambah lapisan convolutional *Regression* (Nguyen et al., 2018). Adapun *layer* (bagian) yang terdapat pada Faster R-CNN adalah :

- *Convolutional Layer*, merupakan bagian utama dari CNN terdiri dari lapisan yang membentuk filter dua dimensi dengan panjang dan tinggi. Ukuran dari filter tersebut jauh lebih kecil dari ukuran gambar sebenarnya. Bagian ini berfungsi untuk mempelajari bagian dari ciri objek serta membentuk *feature map* dari objek yang telah diambil.
- *Feature Map*, merupakan salah satu output pada proses konvolusi. Feature map memiliki fungsi untuk mempermudah dalam memahami apa saja fitur dari input yang terdeteksi dan dipertahankan oleh model. Semakin dalam sebuah layer,

maka semakin umum juga fitur yang akan diambil. Output dari *Feature Map* adalah Vektor dari citra yang diambil

- *Region Proposal Network*, berfungsi untuk mengolah *Feature Map* yang telah dibuat serta memprediksi *Bounding Box* objek dan bagian yang dianggap sebagai objek tersebut
- RoI Pooling, memiliki fungsi untuk ekstraksi *Feature Map* yang sebelumnya diproses pada RPN untuk diklasifikasikan nantinya pada *Classification Layer*.
- *Classification Layer*, merupakan tahap terakhir yang dilakukan pada klasifikasi objek yang telah dideteksi oleh RPN dan membuat label objek yang telah diklasifikasikan.

Adapun arsitektur umum dari Faster R-CNN dapat dilihat pada Gambar 2.12 di bawah ini :



Gambar 2. 12 Arsitektur Faster R-CNN (Ghoury, S *et al.*, 2019)

2.9 Android Studio

Android Studio merupakan sebuah tools *Integrated Development Environment* (IDE) resmi yang dibangun secara khusus untuk mengembangkan aplikasi berbasis Android dengan berdasarkan *IntelliJ IDEA*. *Android Studio* diumumkan pada 16 Mei 2013 pada konferensi Google I/O dan diluncurkan versi pertamanya (1.0) pada Desember 2014. Terdapat beberapa fitur yang disajikan oleh *Android Studio*, diantaranya:

- Memiliki *framework* serta alat *testing* yang memadai
- Memberikan dukungan untuk C++ dan NDK

- Terintegrasi juga dengan GitHub serta dapat menggunakan template kode
- Kaya akan fitur-fitur dan memiliki berbagai jenis *emulator*
- Memberikan dukungan penuh terhadap pengguna *Google Cloud Platform*
- Memiliki sistem *build* berbasis *Gradle*



Gambar 2. 13 Android Studio (Sumber niagahoster.co.id)

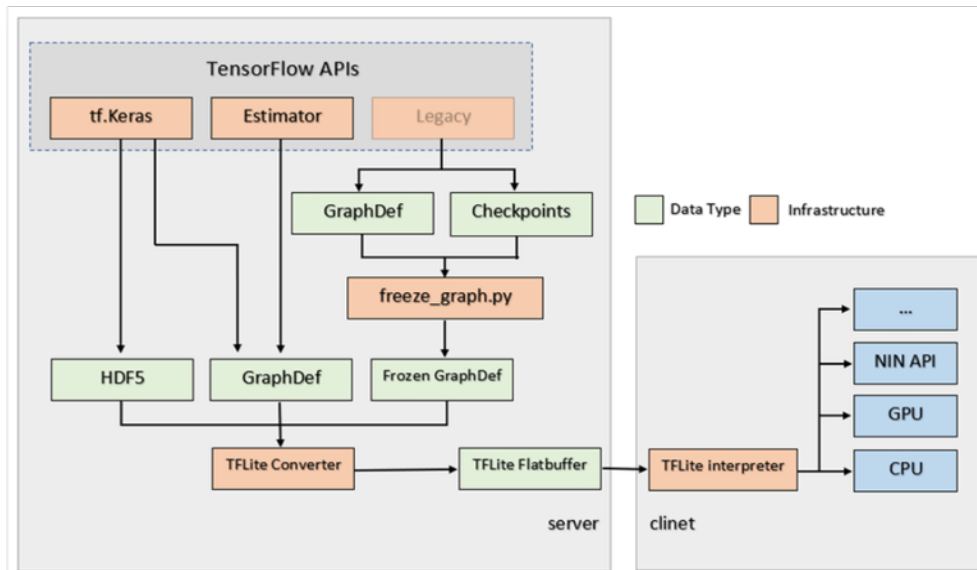
2.10 Python

Python adalah bahasa pemrograman interpretatif multiguna dengan filosofi perancangan yang berfokus pada tingkat keterbacaan kode. *Python* diklaim sebagai bahasa yang menggabungkan kapabilitas, kemampuan, dengan sintaksis kode yang sangat jelas, dan dilengkapi dengan fungsionalitas pustaka standar yang besar serta komprehensif. *Python* bisa dibilang bahasa pemrograman dengan tujuan umum yang dikembangkan secara khusus untuk membuat source code mudah dibaca. *Python* juga memiliki library yang lengkap sehingga memungkinkan programmer untuk membuat aplikasi yang mutakhir dengan menggunakan source code yang tampak sederhana (Ljubomir Perkovic, 2012).

2.11 Tensorflow Lite

Tensorflow Lite merupakan salah satu *library machine learning* yang dapat membantu pengembang dalam menjalankan model di perangkat *edge, mobile* yang bersifat *open source*. Sedangkan *Tensorflow* merupakan sistem *machine learning* yang bekerja dalam lingkungan heterogen dan pada skala yang besar. Untuk menggambarkan komputasi, *Tensorflow* menerapkan grafik aliran data (Abadi, et al, 2016).

Model pembelajaran yang dibuat oleh *Tensorflow* akan dikonversikan kedalam format yang dapat digunakan oleh *Tensorflow Lite* dengan menggunakan *converter*. *Tensorflow Lite interpreter* menggunakan model transformasi untuk inferensi.



Gambar 2. 14 Tensorflow Lite (Sumber researchgate.net)

2.12 Keras

Keras merupakan salah satu *framework* atau *library* yang digunakan dalam pengembangan *deep learning*. *Keras* dirancang untuk memudahkan penggunaan dan mengurangi kompleksitas dalam pengembangan model *deep learning*. *Keras* diciptakan dengan tujuan untuk mendukung algoritma *RNN* dan *CNN*, atau bahkan penggabungan dari keduanya, yang beroperasi baik pada perangkat *GPU* maupun *CPU* sejak *Tensorflow* versi 2.0, *Keras* telah menjadi API Tingkat tinggi resmi untuk *Tensorflow*, yang berarti *Keras* dapat digunakan sebagai bagian dari integral dari lingkungan pengembangan *Tensorflow*.

2.13 Confussion Matrix

Confussion Matrix merupakan salah satu metode yang digunakan untuk mengukur *performance* dalam permasalahan terkait klasifikasi yang menghasilkan output berupa dua kelas ataupun lebih. Perbedaan dari nilai prediksi serta nilai actual pada *Confussion Matrix* menghasilkan empat kombinasi. Adapun tabel dari *Confussion Matrix* terdapat pada Tabel 2.1.

Tabel 2. 1 *Confussion Matrix*

		Prediksi	
		<i>Positive</i>	<i>Negative</i>
Aktual	<i>Positive</i>	TP	FN
	<i>Negative</i>	FP	TN

Berdasarkan tabel *confussion matrix* tersebut, hasil dari proses klasifikasi digambarkan menjadi empat istilah, yaitu :

- a. TP (*True Positive*) adalah data yang bernilai positif serta menghasilkan prediksi yang benar
- b. FN (*False Negative*) adalah data yang bernilai positif, namun menghasilkan prediksi yang salah
- c. FP (*False Positive*) adalah data yang bernilai negatif, namun menghasilkan prediksi yang benar
- d. TN (*True Negative*) adalah data yang bernilai negatif dan menghasilkan prediksi yang salah.

Confussion Matrix juga dapat menghitung berbagai *performance matrix*, seperti :

2.13.1 Accuracy

Accuracy merupakan sebuah metode yang digunakan untuk menghitung berapa akurasi model dapat mengklasifikasikan secara benar. *Accuracy* juga dapat melihat kedekatan dari nilai hasil prediksi dengan nilai aktual. Semakin banyak data yang diprediksi benar, maka semakin tinggi juga akurasi dari sebuah penelitian. Adapun persamaan dari *accuracy* terdapat pada Persamaan 2.1.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \times 100\% \dots\dots\dots (2.1)$$

2.13.2 Precision

Precision merupakan metode yang digunakan untuk menghitung perbandingan jumlah data yang sesuai berdasarkan dari hasil yang didapat oleh sistem dengan jumlah data yang telah terbaca oleh sistem dengan sesuai ataupun tidak sesuai. Adapun persamaan dari *precision* terdapat pada persamaan 2.2.

$$Precision = \frac{TP}{TP+FP} \times 100\% \dots\dots\dots (2.2)$$

2.13.3 Recall

Recall merupakan sebuah metode yang digunakan untuk menghitung kumpulan data, baik yang terbaca maupun tidak terbaca oleh sistem. Adapun persamaan dari *recall* terdapat pada Persamaan 2.3.

$$Recall = \frac{TP}{TP+FP} \dots\dots\dots (2.3)$$

2.13.4 F1-Score

F1-Score merupakan metode yang digunakan untuk menghitung perbandingan dari rata-rata *precision* dan *recall* yang dibobotkan. Adapun persamaan dari *F-1 Score* terdapat pada Persamaan 2.4.

$$F1 - Score = 2 \times \frac{TP+TN}{TP+TN+FP+FN} \dots\dots\dots (2.4)$$

2.14 Penelitian Terdahulu

Terdapat beberapa penelitian mengenai klasifikasi luka bakar pada manusia yang sebelumnya sudah pernah dilakukan, diantaranya adalah penelitian yang dilakukan oleh B. Vanathi et al., (2020) yang berjudul “Densemask RCNN: A Hybrid Model for Skin Burn Image Classification and Severity Grading” dengan tujuan untuk mengembangkan dan mencari sistem terbaik dalam mengklasifikasi luka bakar beserta dengan jenisnya dikarenakan banyaknya kesalahan dalam pengklasifikasian luka bakar secara umum. Peneliti menggunakan 1200 dan 500 gambar luka bakar dengan 100 gambar digunakan untuk validasi yang diolah dengan menggunakan metode transfer learning dengan memakan waktu sekitar 23 detik. Adapun tingkat akurasi dari transfer learning sendiri dalam mengklasifikasikan luka bakar adalah sebesar 86,63%

Pada penelitian yang dilakukan oleh Sayma, Alam et al., (2022) yang berjudul “A deep convolutional neural network-based approach for detecting burn severity from skin burn images” yang bertujuan untuk mendeteksi derajat luka bakar dari gambar luka bakar. Peneliti menggunakan sekitar 1530 gambar dimana 1440 gambar diambil dari situs Kaggle (2022) dan sisanya diambil dari divisi luka bakar dari 3 rumah sakit di Bangladesh dengan menjaga privasi dari pasien rumah sakit. Data dikategorikan menjadi 3 kategori, dengan 670 gambar dari luka derajat 1, 347 gambar luka derajat 2 dan 513 gambar luka derajat 3. Semua gambar diubah bentuknya menjadi ukuran 244*244*3 pixel yang berisikan data tentang baris, kolom, dan *channel* gambar lalu disimpan didalam *3D Multichannel Array*. Dalam melakukan penelitian, peneliti mencari model terbaik juga dalam mengklasifikasikan luka bakar dengan menggunakan metode transfer learning. Didapat hasil akurasi sebesar 77,15% dengan hanya menggunakan Arsitektur CNN, akurasi sebesar 95% dengan menggunakan VGG16

sebagai Transfer learning dengan Arsitektur CNN, akurasi sebesar 89% dengan menggunakan MobileNet sebagai Transfer Learning dengan Arsitektur CNN, dan akurasi sebesar 93% dengan menggunakan ResNet50 sebagai Transfer Learning dengan Arsitektur CNN.

Pada penelitian yang dilakukan oleh Ashish, S (2021) yang berjudul “Skin Burn Detection Using Machine Learning” dengan tujuan untuk mengidentifikasi lokasi luka bakar dan efek yang ditimbulkan sehingga luka dapat dikalkulasikan dengan tujuan perawatan luka yang lebih efektif. Peneliti menggunakan arsitektur CNN dalam mengidentifikasi luka. Data yang digunakan dalam penelitian ini adalah dataset luka bakar dari database BIP_US. Training dilakukan dengan mengklasifikasikan gambar kedalam 2 kelompok. Akurasi yang didapat dari penelitian ini adalah sebesar 93% hanya dengan menggunakan model dasar Arsitektur CNN.

Adapun penelitian lainnya tentang Luka Bakar ialah penelitian yang dilakukan oleh Abubakar, A. et al., (2020) dengan judul “Assessment of Human Skin Burns: A Deep Transfer Learning Approach”. Tujuan dari penelitian ini ialah membuat mesin dengan akurasi tinggi untuk mendiagnosa sebuah jenis luka bakar dibandingkan hanya dengan penglihatan manusia. Adapun kendala yang dihadapi dengan penglihatan manusia ialah penentuan jenis luka bakar yang hanya dapat dinilai secara subjektif oleh dokter spesialis kulit untuk beberapa bentukan luka serta tidak semua daerah di dunia ini memiliki dokter spesialis kulit. Adapun penelitian dilakukan dengan menggunakan model *pre-trained* Resnet50 dengan menggunakan sekitar 676 sampel luka bakar. Didapat akurasi terbaik dalam klasifikasi luka menggunakan metode transfer learning dengan menggunakan pre-trained model ResNet101 dan didapat hasil sebesar 81,66%

Adapun acuan peneliti terhadap metode yang peneliti gunakan ialah penelitian yang dilakukan oleh Ezhilarasi, R. et al (2018) tentang “Tumor Detection in the Brain using Faster R-CNN”. Adapun dataset yang digunakan pada penelitian ini berjumlah 50 dataset gambar hasil MRI otak yang diubah ukurannya dan disamakan menjadi 320*400 pixel setiap gambarnya. Adapun akurasi Faster R-CNN didalam penelitian ini sebesar 99% dalam mengklasifikasi hasil MRI tumor otak.

Selain itu terdapat juga penelitian dengan mengimplementasikan metode yang sama yaitu Faster R-CNN oleh Yang, Y. et al (2020) dengan judul “An Improved Faster

R-CNN for Colorectal Cancer Cell Detection”. Adapun jumlah data yang digunakan dalam penelitian ini sebesar 10000 data training dan 1000 data testing. Adapun akurasi yang didapat dalam penelitian ini sebesar 98,8%

Tabel 2. 2 Penelitian Terdahulu

No	Peneliti	Tahun	Metode	Keterangan
1	Vanathi, B. et al.,	2020	<i>Densemark</i> <i>RCNN</i> <i>Transfer</i> <i>Learning</i>	Penelitian ini membahas tentang Model dalam klasifikasi citra luka bakar. Rata-rata akurasi yang dihasilkan dari penelitian ini sekitar 86,63% dengan menggunakan 1200 dan 500 data gambar.
2	Suha, S. et al.	2022	<i>Transfer</i> <i>Learning, Deep</i> <i>Convolutional</i> <i>Neural Network</i>	Penelitian ini membahas tentang model terbaik untuk mendeteksi derajat luka dari citra luka bakar. Adapun tingkat akurasi dari penelitian ini sekitar 77%, 89%, 93% dan 95% berdasarkan metode yang digunakan dengan jumlah citra sebanyak 1530 citra.
3	Ashish, S	2021	<i>Convolutional</i> <i>Neural Network</i> (CNN)	Penelitian ditujukan pada lokasi luka bakar beserta efek dari luka tersebut (luka kering, luka basah, bernanah. Adapun akurasi dari penelitian ini sekitar 93,3%
4	Abubakar, A. et al.	2020	<i>Transfer</i> <i>Learning</i> dengan <i>Backbone pre-</i> <i>trained model</i> ResNet	Penelitian ini dilakukan guna membuat mesin pendeteksi luka dengan akurasi tinggi guna tepat diagnosa luka bakar. Adapun akurasi dari penelitian ini sekitar 676 sampel data dengan akurasi 81,66%

No	Peneliti	Tahun	Metode	Keterangan
5	Ezhilarasi, R. et al	2018	<i>Faster R-CNN</i>	Penelitian ini dilakukan untuk mendeteksi tumor otak dari hasil MRI otak. Adapun akurasi yang didapat dari penelitian ini bernilai 99% dengan jumlah data awal senilai 50 data
6	Yang, Y. et al.	2020	<i>Faster R-CNN</i>	Penelitian ini bertujuan untuk mengklasifikasikan scan <i>Colorectal Cancer Cell</i> . Adapun akurasi yang didapat berkisar 98% dengan total 10000 data train dan 1000 data test

2.15 Perbedaan Penelitian

Adapun perbedaan penelitian ini dengan penelitian yang sudah dilakukan sebelumnya, diantaranya adalah penelitian terdahulu yang dilakukan oleh (Vanathi *et al*, 2020), (Abubakar *et al*, 2020), (Ashish, 2021), dan (Sayma *et al*, 2022) terdapat pada metode yang dilakukan guna meneliti mengenai klasifikasi serta identifikasi Jenis Luka Bakar pada Kulit Manusia. Penelitian yang dilakukan oleh (Vanathi *et al*, 2020) menggunakan metode *Densemark Regional Convolutional Neural Network* (R-CNN), penelitian oleh (Abubakar *et al*, 2020) menggunakan metode *Transfer Learning* dengan *Pre Trained Model ResNet*, lalu penelitian yang dilakukan oleh (Ashish, 2021) menggunakan metode *Convolutional Neural Network* (CNN) dan penelitian yang dilakukan oleh (Sayma *et al*, 2022) menggunakan metode *Transfer Learning* dengan *Deep Convolutional Neural Network*. Sedangkan pada penelitian ini, peneliti menggunakan metode *Faster Regional Convolutional Network* (Faster R-CNN).

Adapun sistem yang dibangun pada penelitian ialah berupa aplikasi berbasis Android, setiap pengambilan objek dilakukan menggunakan kamera smartphone dalam pengujian aplikasi, tidak seperti penelitian yang lain yang hanya mengukur tingkat akurasi dari model yang telah dilatih dengan menguji sampel yang ada seperti penelitian yang dilakukan oleh (Abubakar *et al*, 2020).




BAB III

ANALISIS DAN PERANCANGAN SISTEM

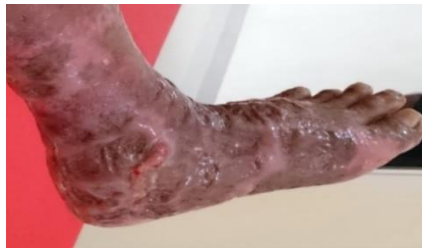
3.1 Dataset

Data input yang digunakan ialah citra luka bakar yang didapat dengan mengambil dari klinik perawatan luka, rumah sakit, dan mengambil data luka dari internet. Untuk data dari klinik perawatan luka serta rumah sakit akan diambil langsung menggunakan kamera smartphone. Seluruh data berekstensi JPG dengan total data sebanyak 1250 citra. Contoh data citra luka bakar yang nantinya akan digunakan pada penelitian ini dapat dilihat pada Tabel 3.1

Tabel 3. 1 Contoh Data Citra Luka Bakar

No	Gambar	Derajat Luka
1		Bukan Luka Bakar
2		Luka Bakar Derajat I
3		Luka Bakar Derajat II

4



Luka Bakar Derajat III

Pengklasifikasian luka berdasarkan jenis lukanya dibagi menjadi 4 kelas diantaranya bukan luka bakar, luka bakar derajat 1, luka bakar derajat 2 dan luka bakar derajat 3. Total data kemudian dibagi menjadi 3 proses, diantaranya : *Data Training*, *Data Validation*, dan *Data Testing*. *Training Data* merupakan proses awal dalam mengolah data dengan melatih data serta mengubah parameter agar dapat disesuaikan dengan data yang diberikan oleh sistem sehingga sistem dapat mengenali ciri dari data tersebut. *Testing Data* merupakan proses pengujian algoritma pada model yang telah dilatih, dan *Validation Data* merupakan tahap memvalidasi seluruh isi model dalam mencegah terjadinya overfitting. Adapun jumlah keseluruhan data dapat dilihat pada tabel berikut:

Tabel 3. 2 Jumlah Data Citra Derajat Luka Bakar

No	Data	Jumlah Data
1	Bukan Luka Bakar	200
1	Luka Bakar Derajat I	350
2	Luka Bakar Derajat II	350
3	Luka Bakar Derajat III	350
	Total Data	1250

Pembagian dilakukan dengan 80% data digunakan sebagai *Data Training*, 10% data digunakan sebagai *Data Testing*, dan 10% data digunakan sebagai *Data Validation*. Maka pada keseluruhan data didapat 1000 data digunakan sebagai *Data Training*, 125 data digunakan sebagai *Data Testing*, dan 125 data digunakan sebagai *Data Validation*

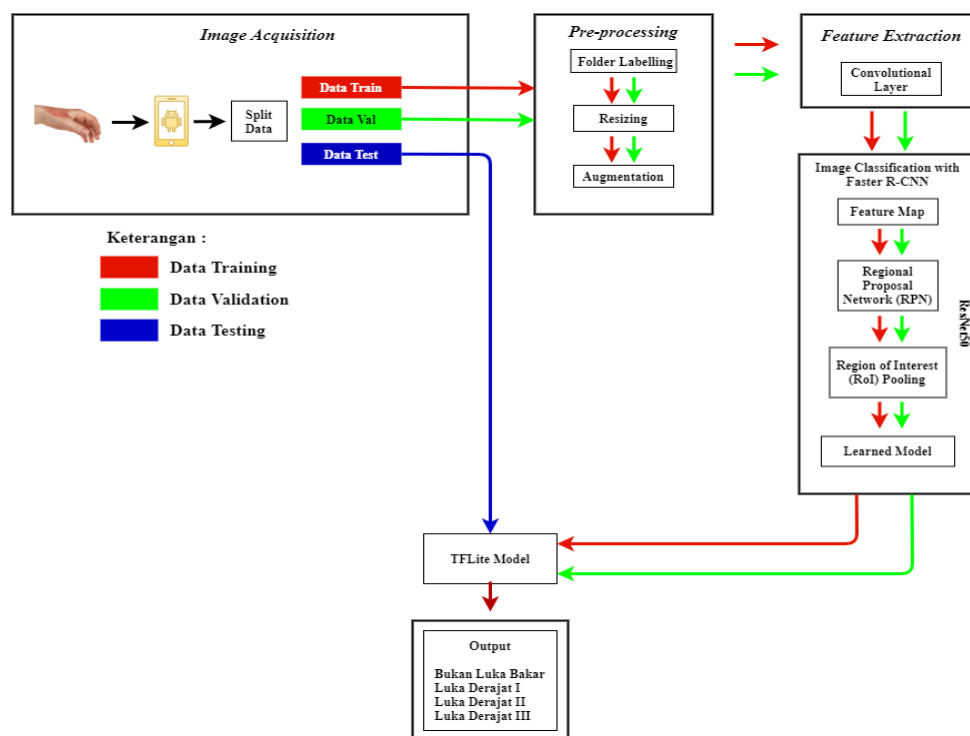
3.2 Analisis Sistem

Adapun metode yang nantinya digunakan pada penelitian ini terdiri dari beberapa tahapan. Tahap Pertama merupakan proses pengumpulan data citra luka bakar yang terbagi atas luka bakar berwarna kemerahan yaitu luka bakar derajat pertama, luka bakar

berwarna putih yaitu luka bakar derajat kedua dan luka bakar berwarna gelap/gosong yaitu luka bakar dengan derajat ketiga.

Tahap berikutnya ialah *pre-processing* data yang terbagi atas 3 tahap. Pertama labelling, memberikan penamaan pada setiap citra luka bakar. Resizing, mengubah seluruh ukuran citra luka bakar menjadi ukuran 300x300 pixel. Lalu pada proses augmentasi data citra akan diubah posisi gambarnya dengan melakukan rotasi gambar sebesar 90 derajat serta mencerminkan arah foto secara vertikal dan horizontal. Citra yang sudah melewati tahap *pre-processing* akan dijadikan data latih dan data uji.

Tahap Terakhir sendiri merupakan tahap pengklasifikasian data citra, dimana penelitian ini menggunakan fitur ekstraksi yang terdapat pada algoritma klasifikasi *Faster Region Convolutional Neural Network* dengan bantuan *pre-trained* model ResNet50 sebagai *backbone*. ResNet50 sendiri merupakan salah satu jenis *backbone* dengan parameter yang jauh lebih kecil dari VGG sehingga waktu yang dibutuhkan untuk mengolah data citra jauh lebih cepat. Adapun output yang nantinya akan dihasilkan setelah tahap klasifikasi berupa 3 jenis luka bakar berdasarkan warna dan bentuk lukanya diantaranya luka bakar derajat 1, luka bakar derajat 2 dan luka bakar derajat 3. Keseluruhan tahap penelitian yang digunakan dapat dilihat dalam arsitektur umum pada gambar 3.1

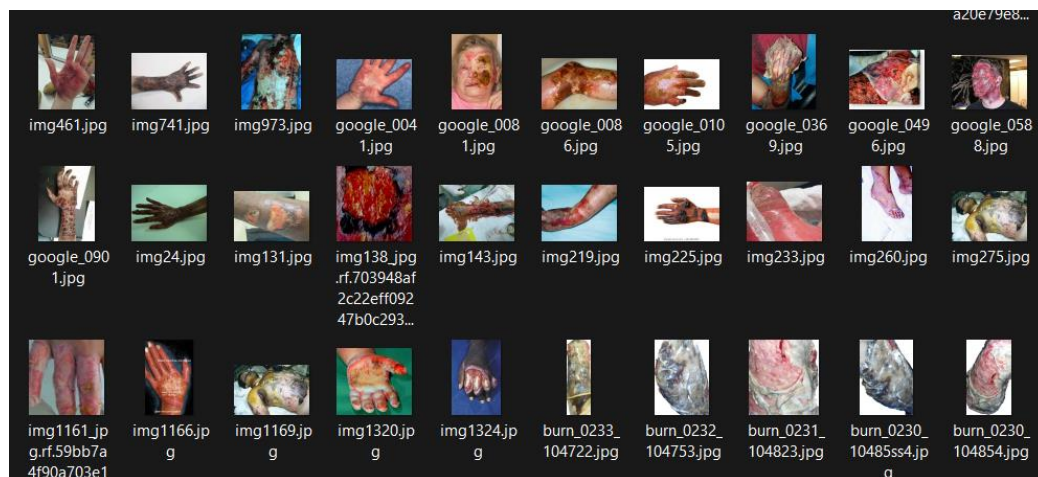


Gambar 3. 1 Arsitektur Umum

Berikut penjelasan terkait proses arsitektur yang terdapat pada gambar 3.1

3.2.1 *Image Acquisition*

Image Acquisition merupakan tahap pertama system berupa pengumpulan dari data citra luka sebagai awal input pada aplikasi. Data citra luka bakar yang digunakan pada penelitian ini terdiri atas 3 jenis luka diantaranya luka derajat 1, luka derajat 2 dan luka derajat 3. Data yang digunakan diambil dari beberapa sumber diantaranya platform *open dataset* Kaggle, *Google Images*, dan beberapa klinik perawatan Luka serta Perawat Luka yang ada di Kota Medan. Data yang bersumber dari Kaggle maupun Google Images akan digunakan sebagai data *training*, dan data yang diambil dari Klinik perawatan Luka maupun Perawat Luka akan digunakan sebagai data *testing*. Seluruh citra berekstensi *.jpg/.jpeg*. Dan data citra akan dibagi menjadi 3, diantaranya : 80% *training data*, 10% *testing data*, 10% *validation data*.



Gambar 3. 2 Contoh data yang digunakan

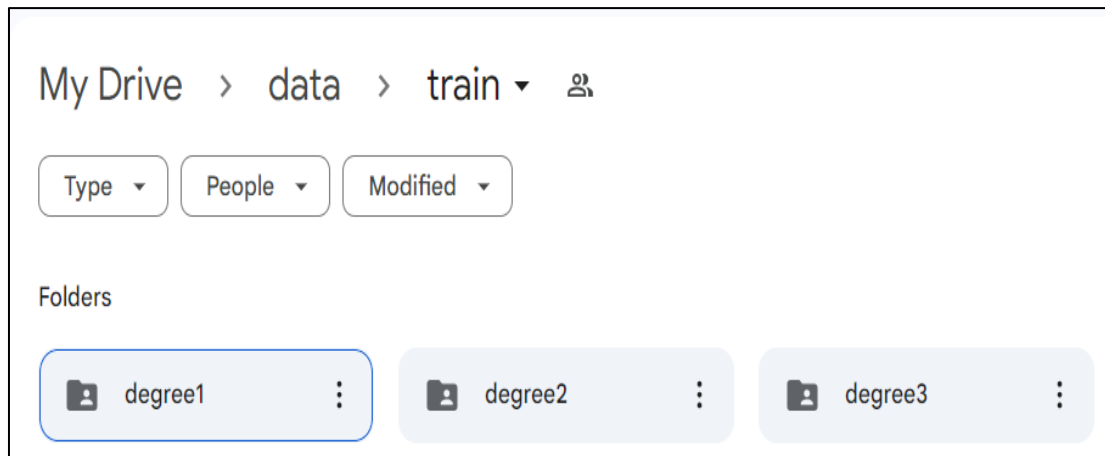
3.2.2 *Image Preprocessing*

Pada proses ini, citra luka bakar akan diproses untuk mendapatkan hasil lebih baik untuk diproses di tahap selanjutnya. Adapun tahapan *preprocessing* yang akan dilakukan pada penelitian ini yaitu *Labelling*, *Resizing*, dan *Data Augmentation*

3.2.2.1 *Labelling*

Proses *Labelling* dilakukan guna memberi nama atau membuat label pada data yang akan digunakan untuk *training* sistem. Proses pelabelan data citra pada penelitian ini dilakukan dengan mengelompokkan gambar kedalam folder *Google Drive*, dimana setiap folder memiliki nama yang disesuaikan dengan Tingkat klasifikasi, diantaranya

Degree1, Degree2, dan Degree3. Lalu sistem akan *menginput* gambar dari folder-folder tersebut sesuai dengan label yang tertera. Adapun proses dapat dilihat pada gambar 3.3 di bawah ini :



Gambar 3. 3 Labelling

3.2.2.2 *Resizing*

Proses *Resizing* merupakan proses mengubah ukuran gambar yang akan diinput kedalam sistem menjadi bentuk yang sama. Semakin banyak jumlah *pixel* maka pemrosesan sistem akan semakin lama, maka dari itu *resizing* dilakukan guna membuat citra memiliki pixel yang lebih sedikit dan ukuran pixel yang sama. Pada penelitian ini, citra diubah ukurannya menjadi 300x300 *pixel* guna menciptakan keseragaman pada *dataset* serta memudahkan untuk pelatihan pada sistem, sekaligus meminimalisir penggunaan sumber daya komputasi. Berikut *pseudocode* untuk melakukan proses *resizing* :


```
START PROCEDURE thumbnail(data, size)
    CALL data.thumbnail(size)
END PROCEDURE
```

Gambar 3. 4 *Pseudocode Resizing*

Jika dilihat pada Gambar 3.4, “*procedure*” digunakan untuk mendefinisikan sebuah fungsi yang digunakan untuk mengubah ukuran dan “*call*” digunakan untuk menjalankan fungsi ‘*size*’ yang berfungsi mengubah ukuran gambar pada “*thumbnail*”

Adapun contoh proses *resizing* pada adalah sebagai berikut :

214	103	255	231
108	205	235	215
148	85	267	189
214	147	70	67
78	203	54	134
231	254	164	194
108	233	103	145
117	45	233	143
65	195	244	154
78	144	201	127



P1	P2
P3	P4

Gambar 3. 5 Proses Perhitungan *Resizing*

Pada Gambar 3.5, terdapat proses pengubahan ukuran citra 4 x 10 *pixel* menjadi 2 x 2 *pixel*. Dalam proses ini, nilai untuk setiap *pixel* yang baru didapatkan dengan cara mengambil setiap nilai dari rata-rata 4 *pixel terdekat* untuk diubah kedalam bentuk citra yang lebih kecil yaitu 2 x 2. Berikut detail perhitungan proses *resizing* :

$$P1 = (214+103+108+205+148+85+214+147+78+203) : 10 = 151$$

$$P2 = (255+231+235+215+267+189+70+67+54+134) : 10 = 172$$

$$P3 = (231+254+108+233+117+45+65+195+78+144) : 10 = 147$$

$$P4 = (164+194+103+145+233+143+244+154+201+127) : 10 = 171$$

Setiap nilai *pixel* yang baru dihasilkan dari perhitungan proses *resizing* dengan matriks diatas dapat dilihat pada gambar 3.6 di bawah ini :

151	172
147	171

Gambar 3. 6 Nilai Matrix dari proses *Resizing*

Adapun contoh citra sebelum dan sesudah proses *resizing* dapat dilihat pada gambar 3.7 di bawah ini :



Gambar 3. 7 Citra Luka Bakar sebelum (kiri) dan sesudah di *resize* (kanan)

3.2.2.3 *Augmentasi*

Augmentasi (*Data Augmentation*) merupakan proses memanipulasi atau memodifikasi bentuk citra sehingga citra yang asli akan berubah bentuk dan posisinya. Augmentasi juga berguna untuk meningkatkan jumlah serta keberagaman data *training* guna menghasilkan data yang lebih bervariasi. Proses augmentasi juga membuat transformasi pada citra data yang asli dengan memutar data citra pada interval tertentu, yaitu *flip* foto baik secara *vertical* maupun *horizontal* serta merotasi gambar sebesar 90 derajat yang berguna untuk membantu peningkatan kuantitas jumlah *dataset*. Tanpa Augmentasi, sangat mudah terjadi *overfitting* pada model. Dengan kata lain, Augmentasi memiliki peran penting dalam mencegah terjadinya *overfitting* pada model dengan memperkaya jumlah data (Sanjaya, *et al* 2020). Adapun *pseudocode* proses augmentasi dapat dilihat pada gambar 3.8 di bawah ini :

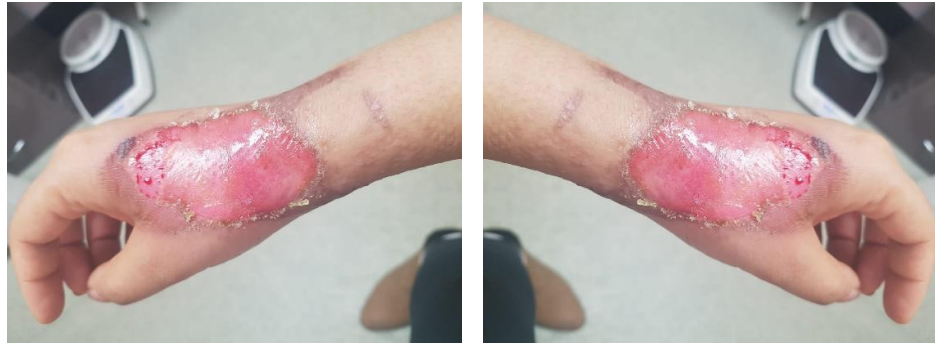
```

START PROCEDURE save_transform_images(data, folder, target-
name, index)
    CALL("flip_left_right")
    index += 1
    CALL("rotate_90")
    index += 1
    CALL("flip_top_bottom")
    index += 1
END PROCEDURE

```

Gambar 3. 8 *Pseudocode* Augmentasi

Berbagai citra Luka Bakar baik sebelum dan sesudah proses augmentasi dapat dilihat pada gambar 3.9 , 3.10 dan 3.11 di bawah ini



Gambar 3. 9 Citra luka sebelum (kiri) dan sesudah *flip horizontal* (kanan)



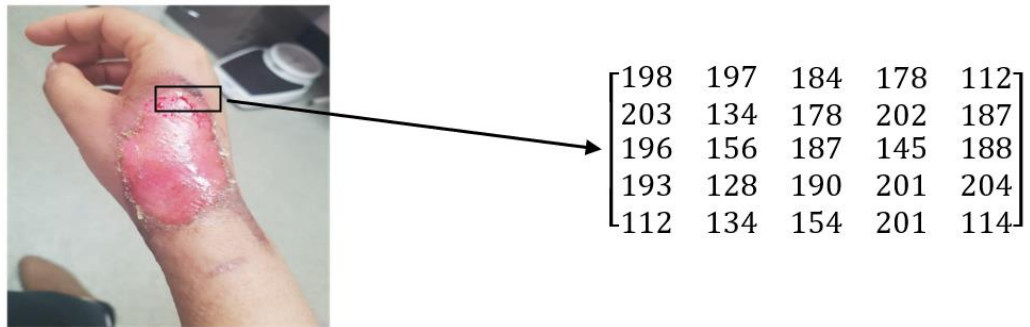
Gambar 3. 10 Citra luka sebelum (kiri) dan sesudah *flip vertical* (kanan)



Gambar 3. 11 Citra luka sebelum (kiri) dan sesudah *rotate 90°* (kanan)

3.2.3 Feature Extraction

Feature Extraction merupakan salah satu proses mengambil sebuah data maupun informasi yang berhubungan dari data mentah maupun dataset yang memiliki sifat kompleks untuk digunakan dalam melakukan analisis serta pemodelan data (visualisasi data). *Convolutional layer* yang digunakan sebagai *feature extraction* pada *Faster R-CNN* memiliki fungsi untuk mengekstraksi fitur-fitur penting dari data *input* serta memeriksa bagian-bagian penting yang menjadi fitur dari citra. *Convolutional Layer* menggunakan filter dalam mengekstrak segala fitur yang terkain guna mengenali objek. Pada setiap lokasi gambar, akan dihasilkan sebuah angka berupa *dot product* antara bagian gambar dengan filter yang digunakan. Dengan menggeser (*convolve*) filter pada sebuah gambar maka akan dihasilkan sebuah *activation map*. Layer ini yang memiliki fungsi untuk membuat *feature map* dari objek yang telah diambil. Adapun contoh matriks gambar *input* yang terdapat pada penelitian ini sebagai berikut :



Gambar 3. 12 *Matrix Input* pada Gambar

Kemudian matriks dari input gambar diatas dihitung dengan menggunakan filter yang terdapat pada *convolutional layer*. *Activation map* akan dihasilkan dari setiap pengulangan proses perhitungan ini untuk setiap elemen dari gambar *input*. Berikut merupakan contoh salah satu perhitungan *filter* pada matriks gambar penelitian :

$$\begin{bmatrix} 198 & 197 & 184 & 178 & 112 \\ 203 & 134 & 178 & 202 & 187 \\ 196 & 156 & 187 & 145 & 188 \\ 193 & 128 & 190 & 201 & 204 \\ 112 & 134 & 154 & 201 & 114 \end{bmatrix} \times \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} = \begin{matrix} A1 & A2 & A3 \\ A4 & A5 & A6 \\ A7 & A8 & A9 \end{matrix}$$

Pada *Convolutional layer* nilai matriks gambar pada penelitian ini sebesar 3 x 5 *pixel*. Lalu nilai tersebut akan dikalikan dengan nilai filter sebesar 3 x 3. Matriks gambar

input tersebut akan bergeser sebesar 3 x 3 dari kiri atas ke kanan dengan pergeseran masing-masing 1 pixel sampai melewati semua pixel dari barisan pertama. Pergeseran matriks tersebut adalah sebagai berikut :

$$\begin{bmatrix} 198 & 197 & 184 & 178 & 112 \\ 203 & 134 & 178 & 202 & 187 \\ 196 & 156 & 187 & 145 & 188 \\ 193 & 128 & 190 & 201 & 204 \\ 112 & 134 & 154 & 201 & 114 \end{bmatrix} \begin{bmatrix} 198 & 197 & 184 & 178 & 112 \\ 203 & 134 & 178 & 202 & 187 \\ 196 & 156 & 187 & 145 & 188 \\ 193 & 128 & 190 & 201 & 204 \\ 112 & 134 & 154 & 201 & 114 \end{bmatrix} \begin{bmatrix} 198 & 197 & 184 & 178 & 112 \\ 203 & 134 & 178 & 202 & 187 \\ 196 & 156 & 187 & 145 & 188 \\ 193 & 128 & 190 & 201 & 204 \\ 112 & 134 & 154 & 201 & 114 \end{bmatrix}$$

$$\begin{bmatrix} 198 & 197 & 184 & 178 & 112 \\ 203 & 134 & 178 & 202 & 187 \\ 196 & 156 & 187 & 145 & 188 \\ 193 & 128 & 190 & 201 & 204 \\ 112 & 134 & 154 & 201 & 114 \end{bmatrix} \begin{bmatrix} 198 & 197 & 184 & 178 & 112 \\ 203 & 134 & 178 & 202 & 187 \\ 196 & 156 & 187 & 145 & 188 \\ 193 & 128 & 190 & 201 & 204 \\ 112 & 134 & 154 & 201 & 114 \end{bmatrix} \begin{bmatrix} 198 & 197 & 184 & 178 & 112 \\ 203 & 134 & 178 & 202 & 187 \\ 196 & 156 & 187 & 145 & 188 \\ 193 & 128 & 190 & 201 & 204 \\ 112 & 134 & 154 & 201 & 114 \end{bmatrix}$$

$$\begin{bmatrix} 198 & 197 & 184 & 178 & 112 \\ 203 & 134 & 178 & 202 & 187 \\ 196 & 156 & 187 & 145 & 188 \\ 193 & 128 & 190 & 201 & 204 \\ 112 & 134 & 154 & 201 & 114 \end{bmatrix} \begin{bmatrix} 198 & 197 & 184 & 178 & 112 \\ 203 & 134 & 178 & 202 & 187 \\ 196 & 156 & 187 & 145 & 188 \\ 193 & 128 & 190 & 201 & 204 \\ 112 & 134 & 154 & 201 & 114 \end{bmatrix} \begin{bmatrix} 198 & 197 & 184 & 178 & 112 \\ 203 & 134 & 178 & 202 & 187 \\ 196 & 156 & 187 & 145 & 188 \\ 193 & 128 & 190 & 201 & 204 \\ 112 & 134 & 154 & 201 & 114 \end{bmatrix}$$

Kemudian hasil perkalian antar matriks filter dan matriks pada input menjadi nilai dari *feature map*. Berikut merupakan perhitungan *convolutional layer* :

$$\begin{aligned} A1 &= (198 \times 0) + (197 \times 1) + (184 \times 0) + (203 \times 1) + (134 \times 0) + (178 \times 1) \\ &+ (196 \times 0) + (156 \times 1) + (187 \times 0) \end{aligned}$$

$$= 197 + 203 + 178 + 156 = 734$$

$$\begin{aligned} A2 &= (197 \times 0) + (184 \times 1) + (178 \times 0) + (134 \times 1) + (178 \times 0) + (202 \times 1) \\ &+ (156 \times 0) + (187 \times 1) + (145 \times 0) \end{aligned}$$

$$= 184 + 134 + 202 + 187 = 707$$

$$\begin{aligned} A3 &= (184 \times 0) + (178 \times 1) + (112 \times 0) + (178 \times 1) + (202 \times 0) + (187 \times 1) \\ &+ (187 \times 0) + (145 \times 1) + (188 \times 0) \end{aligned}$$

$$= 178 + 178 + 187 + 145 = 688$$

$$\begin{aligned} A4 &= (203 \times 0) + (134 \times 1) + (178 \times 0) + (196 \times 1) + (156 \times 0) + (187 \times 1) \\ &+ (193 \times 0) + (128 \times 1) + (190 \times 0) \end{aligned}$$

$$= 134 + 196 + 187 + 128 = 645$$

$$A5 = (134 \times 0) + (178 \times 1) + (202 \times 0) + (156 \times 1) + (187 \times 0) + (145 \times 1) \\ + (128 \times 0) + (190 \times 1) + (201 \times 0)$$

$$= 178 + 156 + 145 + 190 = 669$$

$$A6 = (178 \times 0) + (202 \times 1) + (187 \times 0) + (187 \times 1) + (145 \times 0) + (188 \times 1) \\ + (190 \times 0) + (201 \times 1) + (204 \times 0)$$

$$= 202 + 187 + 188 + 201 = 778$$

$$A7 = (196 \times 0) + (156 \times 1) + (187 \times 0) + (193 \times 1) + (128 \times 0) + (190 \times 1) \\ + (112 \times 0) + (134 \times 1) + (154 \times 0)$$

$$= 156 + 193 + 190 + 134 = 673$$

$$A8 = (156 \times 0) + (187 \times 1) + (145 \times 0) + (128 \times 1) + (190 \times 0) + (201 \times 1) \\ + (134 \times 0) + (154 \times 1) + (201 \times 0)$$

$$= 187 + 128 + 201 + 154 = 670$$

$$A9 = (187 \times 0) + (145 \times 1) + (188 \times 0) + (190 \times 1) + (201 \times 0) + (204 \times 1) \\ + (154 \times 0) + (201 \times 1) + (114 \times 0)$$

$$= 145 + 190 + 204 + 201 = 740$$

Dengan perhitungan diatas, maka didapat contoh hasil matriks dari penelitian ini yang sudah diproses oleh *convolutional layer* yaitu :

$$\begin{bmatrix} 734 & 707 & 688 \\ 645 & 669 & 778 \\ 673 & 670 & 740 \end{bmatrix}$$

3.2.4 Image Classifier

Proses klasifikasi yang dilakukan pada penelitian ini menggunakan algoritma *Faster Region Convolutional Neural Network* dengan bantuan *tools open source* milik Google bernama *google colab* dengan menggunakan *framework TensorFlow*. Klasifikasi sendiri dibagi kedalam 2 buah proses diantaranya proses *training* dan proses *testing*. Berikut pseudocode untuk melakukan proses klasifikasi *Faster R-CNN* dengan menggunakan *google colab* :

```

START PROCEDURE train_summarize_model(train_data,
validation_data, batch_size, epochs)

    Model = image_classifier.create(train, data
validation_data = validation_data,
Batch_size = batch_size,
Epochs=epochs)
    CALL model.summary()
END PROCEDURE

```

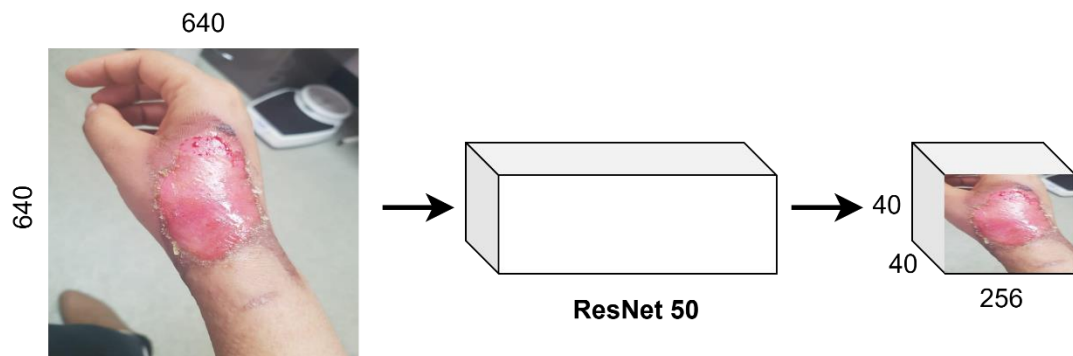
Gambar 3. 13 Pseudocode proses Klasifikasi

Pseudocode diatas merupakan model untuk *image classification*, dimana terdapat dua argumen yaitu *train_data* dan *validation_data* yang berguna untuk melatih serta mengevaluasi model. *Batch size* merupakan sebuah argumen yang menentukan ukuran *batch* (Jumlah keseluruhan data) yang digunakan dalam pelatihan model. *Epochs* merupakan salah satu argumen yang menentukan jumlah iterasi yang dilakukan selama pelatihan model. *Model summary* merupakan metode untuk menampilkan arsitektur dari model yang telah dibuat dengan menggunakan data dari *train_data* dan *validation_data* guna menghasilkan model yang dapat mempelajari korelasi antara gambar beserta label yang sesuai, sehingga ketika model diberikan sebuah gambar yang belum pernah model lihat, model dapat memprediksi label yang sesuai dengan gambar.

Beberapa tahapan Proses dari Faster R- CNN, yaitu :

3.2.4.1 Feature Map

Feature Map merupakan map yang dibuat oleh *convolutional layer* yang berisi tentang representasi vektor dari *image*. *Convolutional layer* menghasilkan dua *feature map* yang sama. Dimana *feature map* pertama diolah RPN untuk menghasilkan *region proposal*, *feature map* lainnya langsung dikirim ke *pooling layer*. Gambar yang ditampilkan sebagai *height*, *width*, dan *depth* diteruskan ke lapisan tengah menggunakan CNN yang sebelumnya sudah dilatih dan berakhir sebagai *convolutional feature map*. Ukuran dimensi gambar pada *feature map* awalnya yaitu 640 x 640 x 3, setelah dilakukan pelatihan menggunakan Resnet50, gambar berubah ukurannya menjadi 40 x 40 x 256. Adapun representasi dari feature map dapat dilihat pada gambar di bawah ini :

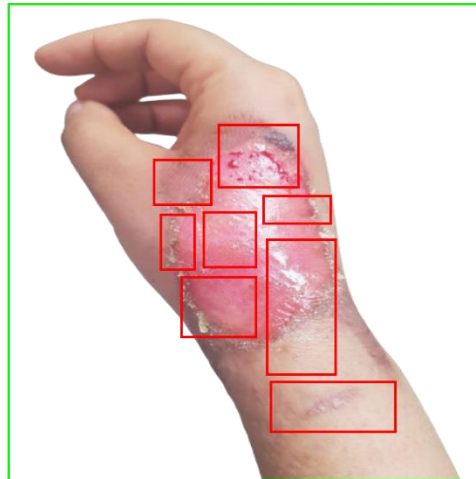


Gambar 3. 14 *Feature Map*

3.2.4.2 *Region Proposal Network*

Setelah melalui *feature map*, citra diproses oleh *RPN* dan dibuat kedalam sebuah convolutional *layer* untuk memprediksi benda yang dihitung sebagai objek serta membuat prediksi *bounding box* pada objek. *RPN* dibagi kedalam 2 convolutional *layer* dimana masing-masing *layer* memiliki tanggung jawab untuk mendeteksi tata letak objek dan satunya lagi memprediksi *bounding box*. *RPN* nantinya akan menilai objek pada sebuah citra dan membuat nilai objek menjadi kotak pembatas *output*. *RPN* memiliki beberapa *anchors* sebagai kotak pembatas tetap yang disusun dengan ukuran dan rasio yang berbeda-beda sebagai acuan untuk memprediksi posisi objek luka bakar pada citra.

RPN akan menilai objek pada suatu citra dan membuat nilai objek menjadi kotak pembatas *output*. Titik 40 x 40 yang terdapat pada gambar 3.14 di *feature map* dinilai sebagai *anchors*. Penentuan rasio dan ukuran secara spesifik diperlukan untuk setiap *anchors*, yaitu 1:1, 1:2, 2:1 untuk tiga rasio dan 128% , 256% , 512% untuk tiga ukuran pada citra. Dengan *anchors* berukuran 3 x 3 = 9 kotak, maka ukuran *anchors* yang terdapat pada citra adalah 40 x 40 x 9 = 14.400 kotak. Karena ukuran *anchors* yang terdapat pada citra terlalu besar, maka akan dipilih sebanyak 256 kotak sebagai *mini batch*. Representasi dari *RPN* dapat dilihat pada gambar di bawah ini :



Gambar 3. 15 *Region Proposal Network*

Pada gambar 3.15, kotak yang berwarna hijau merupakan *anchors* yang memiliki fungsi untuk menilai sebagai objek luka bakar yang akan dideteksi, dan kotak berwarna merah merupakan kumpulan *anchors* yang menilai objek bukan merupakan objek yang akan dideteksi. Hasil dari tahapan ini akan mendeteksi apakah citra yang dideteksi merupakan objek luka bakar atau bukan.

3.2.4.3 *Region of Interest Pooling*

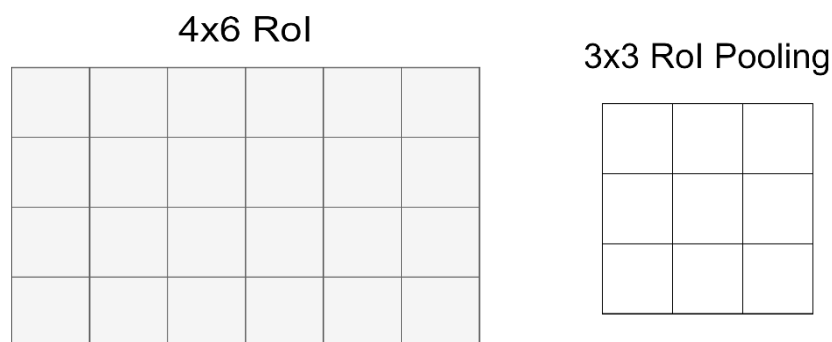
Region of Interest Pooling (RoI) merupakan wilayah yang direkomendasikan dari gambar asli. Sebelum dilakukan proses klasifikasi, objek harus melewati proses *RoI pooling*. *RoI pooling* menyamakan ukuran dari *feature map* dan *region proposal* yang telah diolah di RPN. *RoI Pooling* kemudian mengirim informasi yang terdapat pada *feature map* dan *proposal* untuk diklasifikasikan pada *classification layer*. Sebelum melakukan proses klasifikasi, objek harus melalui proses *RoI Pooling*. *RoI Pooling* memiliki fungsi untuk menyamakan setiap ukuran dari *feature map* dan *region proposal* yang telah diolah oleh RPN. *RoI Pooling* kemudian mengirim informasi *feature map* dan *proposal* untuk diklasifikasikan pada *classification layer*. Berikut merupakan contoh dari target ROI :



Gambar 3. 16 Target *Region of Interest (RoI) Pooling*

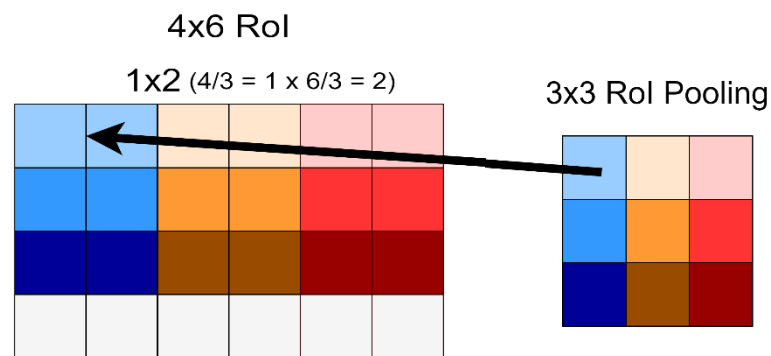
Gambar 3.15 menunjukkan objek yang akan dijadikan target oleh *RoI* dengan ukuran kotak 200x140 yang akan dibagi dengan 32 (faktor skala). Berikut merupakan perhitungannya :

Lebar : $200/32 = 6.25$, dibulatkan menjadi 6
 Tinggi : $140/32 = 4.37$, dibulatkan menjadi 4



Gambar 3. 17 Pemetaan *Pooling* dan Lapisan *Pooling*

Pada Gambar 3.16, ukuran *RoI* sebelum dilakukan pemetaan adalah 4 x 6 x 512. *RoI* yang berukuran 4 x 6 tidak dapat digunakan kedalam *fully connected layer* karena ukuran pemetaan yang terdapat pada *pooling* terlalu besar. Oleh karena itu dilakukan tahap *pooling* untuk membuat ukuran *RoI* menjadi 3 x 3 agar bisa masuk ke *fully connected layer*. Berikut Proses *RoI Pooling* :



Gambar 3. 18 Pemetaan *Pooling* dan Lapisan *Pooling*

Pada Gambar 3.17, Ukuran *RoI* yang sebelumnya 4 x 6 akan di *pooling* hingga menjadi ukuran 3 x 3 agar bisa masuk ke *fully connected layer*. Untuk mendapatkan nilai ukuran *RoI* 3 x 3, ukuran *RoI* 4 x 6 akan dibagi 3.

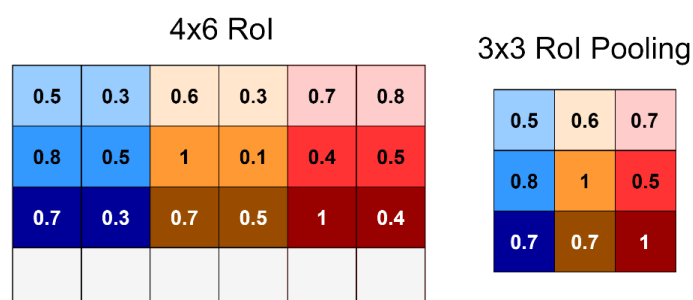
Panjang : $6 / 3 = 2$
 Lebar : $4 / 3 = 1.3$, dibulatkan menjadi 1

4x6 RoI (Lost Data)

0.5	0.3	0.6	0.3	0.7	0.8
0.8	0.5	1	0.1	0.4	0.5
0.7	0.3	0.7	0.5	1	0.4
0.7	0.2	0.9	0.6	0.9	0.4

Gambar 3. 19 Pemetaan *Pooling*

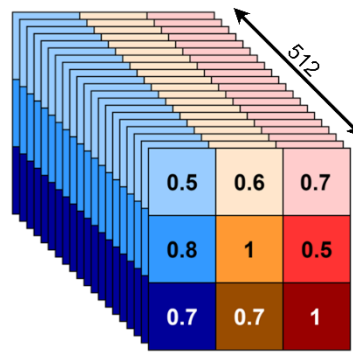
Nilai yang diambil pada proses data *pooling* merupakan nilai tertinggi yang terdapat pada suatu matriks.



Gambar 3. 20 Proses *Pooling Data*

Ukuran yang sebelumnya beragam akan diubah menjadi satu ukuran yaitu 3 x 3 untuk keseluruhan data. Proses ini dilakukan pada seluruh matriks *RoI*. Tidak hanya pada lapisan paling atas, melainkan pada seluruh lapisan. Berikut merupakan hasil akhir *RoI Pooling* :

3x3 RoI Pooling (Full Size)



Gambar 3. 21 *RoI Pooling 3x3 (Full Size)*

Gambar diatas merupakan bentuk representasi ratusan matriks 3 x 3 x 512 dari hasil *RoI Pooling*. Masing-masing matriks ini nantinya akan dikirim melalui seluruh lapisan jaringan untuk masing-masing model, dimana setiap model akan menghasilkan klasifikasi objek berupa derajat luka bakar secara terpisah

3.2.5 *Learned Model*

Proses Klasifikasi citra yang dilakukan pada saat data training adalah dengan menghasilkan sebuah learned model yang digunakan pada aplikasi. *Learned model (Saved Model)* merupakan tahap modelling yang berisikan tentang grafik serta bobot dari model hasil *training* data yang terdapat 4 jenis klasifikasi pada luka bakar diantaranya Bukan Luka Bakar, Luka Bakar derajat I, Luka Bakar derajat II, dan Luka Bakar derajat III. Model nantinya akan bekerja dengan menghitung kemungkinan keberadaan dari sebuah objek pada setiap kandidat wilayah yang dihasilkan oleh *Region Proposal Network (RPN)*, kemudian mengklasifikasikan kandidat dari wilayah tersebut menjadi kelas objek yang sesuai.

3.2.5.1 *Proses Training*

Proses *Training* merupakan sebuah proses dalam pembuatan model dengan memanfaatkan data-data yang telah disediakan. Pada proses ini, model akan mempelajari setiap pola dan fitur dari data serta menyesuaikan bobot data sesuai dengan hasil pelatihan. Hasil dari *model* ini akan digunakan pada proses *testing*. Pada proses

ini data akan diinput kedalam 3 bagian, diantaranya : data *training*, data *validation*, dan data *testing*.

- *Data Training*

Data training merupakan data yang digunakan untuk melatih model. Model akan mempelajari pola dari fitur data *training* dan menyesuaikan bobotnya dengan hasil pelatihan. Data ini berjumlah 80% dari total jumlah data penelitian yaitu 1000 data

- *Data Validation*

Merupakan data yang digunakan untuk mengevaluasi model Ketika proses pelatihan berlangsung. Data *Validation* digunakan untuk memastikan tidak terjadinya *overfitting* pada data *training*. Data validation berjumlah 10% dari total jumlah data penelitian yaitu 125 data.

- *Data Testing*

Merupakan data yang akan digunakan untuk menguji seberapa baik model tersebut dalam memprediksi serta mengelola data yang belum pernah dilihat sebelumnya. Data *testing* merupakan data paling objektif untuk mengevaluasi kinerja model dikarenakan data ini tidak termasuk dalam proses pelatihan model. Data testing sendiri berjumlah berjumlah 10% dari total jumlah data penelitian yaitu 125s data.

3.2.5.2 Proses Testing

Proses *Testing* merupakan proses untuk mengevaluasi kinerja model yang telah dibuat dengan menggunakan data baru, data yang belum pernah dilihat sebelumnya oleh model. Pada proses ini, model akan diberikan *input* berupa data *testing* dan diuji untuk melihat seberapa baik model tersebut dalam memprediksi serta mengolah data tersebut. Hasil dari proses *testing* akan menunjukkan seberapa baik model tersebut untuk bekerja pada data baru yang belum pernah dilihat sebelumnya. Fase ini menguji metode *Faster R-CNN* dalam mengklasifikasi derajat luka bakar. Model yang telah ditentukan jumlah klasifikasinya akan melewati beberapa tahapan *layers* yang terdapat pada algoritma *Faster R-CNN*

3.2.6 TFLite Model

Learned model yang telah dibuat diubah menjadi format Tensorflow Lite untuk dijalankan pada aplikasi web nantinya. Tensorflow Lite sendiri menyediakan Framework model Tensorflow yang telah dilatih. *Learned Model* yang berukuran besar diubah ke *TFLite* model sehingga ukuran model menjadi ukuran biner yang lebih kecil

dan lebih ringan untuk mengejar efisiensi dari program web. Berikut merupakan pseudocode dari proses *Tensorflow Lite* :

```

START PROCEDURE save.model('model.h5')
    model = load_model('model.h5')
    converter = TFLiteConverter_from_keras_model(model),
    tflite_model = converter.convert()
    CALL tflite_model()
END PROCEDURE

```

Gambar 3. 22 *Pseudocode Tensorflow Lite*

Adapun isi dari *TF Lite model* merupakan representasi biner atas arsitektur serta parameter-parameter model yang sudah dilatih sebelumnya. *File* ini berisikan struktural yang telah terorganisir untuk menyimpan informasi tentang *layer* yang terdapat dalam model, bobot/ukuran (*weights*), serta konfigurasi lainnya yang diperlukan untuk menjalankan model pada perangkat *android*.

3.2.7 Output

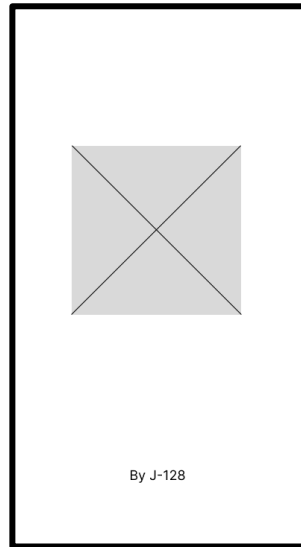
Hasil output dari semua proses diatas berupa klasifikasi jenis luka bakar pada manusia yang dibagi menjadi 3 bagian diantaranya Luka Bakar derajat I, Luka Bakar derajat II, dan Luka Bakar derajat III

3.3 Perancangan Antarmuka Sistem

Tahap ini merupakan salah satu tahapan yang menunjukkan serta menjelaskan mengenai perancangan tampilan antar muka sistem Klasifikasi Derajat Luka Bakar dengan tujuan memberikan gambaran kepada pengguna bagaimana mengoperasikan aplikasi sebagaimana mestinya. Terdapat 4 menu didalam aplikasi ini, diantaranya : *Splash Screen*, *Main Page*, *Article Page*, *Testing* dan *Result*.

3.3.1 Rancangan Tampilan *Splash Screen*

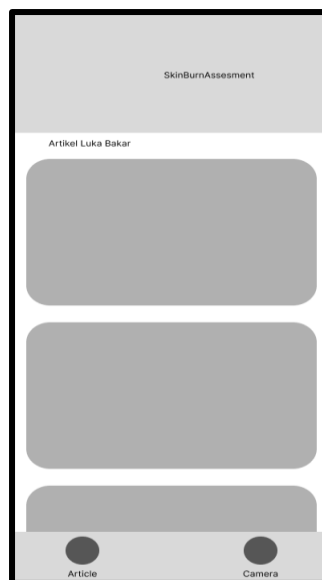
Splash Screen merupakan tampilan awal saat pengguna membuka aplikasi. Pada tahap ini, pengguna akan dihadapkan oleh tampilan logo aplikasi sebelum masuk ke halaman utama.



Gambar 3. 23 *Splash Screen*

3.3.2 Rancangan tampilan *Main Page*

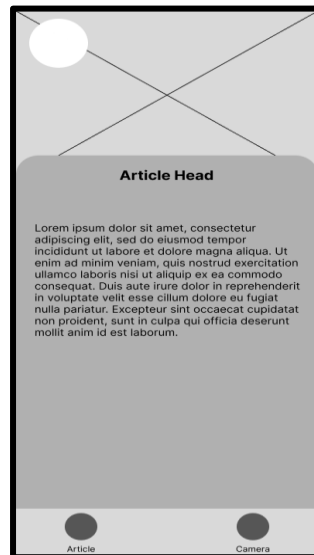
Main Page merupakan tampilan kedua yang muncul setelah tampilan *Splash Screen* dan merupakan tampilan utama yang terdapat pada aplikasi. Terdapat beberapa tampilan didalam seperti tampilan *Article* yang Ketika di klik akan menuju ke laman artikel serta tombol di *Navigation Bar* bawah untuk mengarah ke fitur klasifikasi pada aplikasi



Gambar 3. 24 *Main Page*

3.3.3 Rancangan tampilan *Article Page*

Article Page merupakan salah satu laman dan fitur yang terdapat pada aplikasi ini dimana memuat artikel-artikel yang up to date yang berisikan tentang berita luka bakar, penanganan luka bakar, dan info-info lainnya seputar luka bakar



Gambar 3. 25 *Article Page*

3.3.4 Rancangan tampilan *Testing*

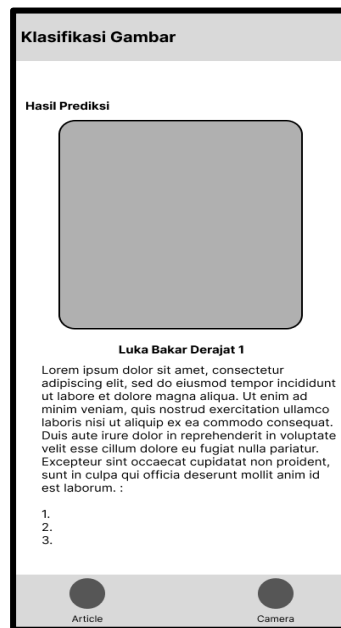
Testing Page merupakan tampilan untuk mengidentifikasi objek luka bakar yang nantinya pengguna akan mengambil foto objek luka bakar yang ingin diketahui derajat lukanya didalam bingkai yang sudah disediakan dalam aplikasi.



Gambar 3. 26 *Testing Page*

3.3.5 Rancangan tampilan Halaman *Result*

Tampilan result merupakan tampilan yang muncul setelah pengguna memproses gambar yang ingin diklasifikasikan, tampilan ini akan menampilkan gambar yang telah diambil beserta juga penjelasan dari foto yang telah diambil serta informasi-informasi singkat seputar luka bakar sesuai derajat luka.



Gambar 3. 27 *Result Page*

BAB IV

PENGUJIAN SISTEM

4.1 Implementasi Sistem

Klasifikasi derajat luka bakar menggunakan metode Faster Regional Convolutional Neural Network dibangun dan diimplementasikan kedalam sistem serta dibutuhkan perangkat keras, perangkat lunak, dan data-data pendukung, yaitu :

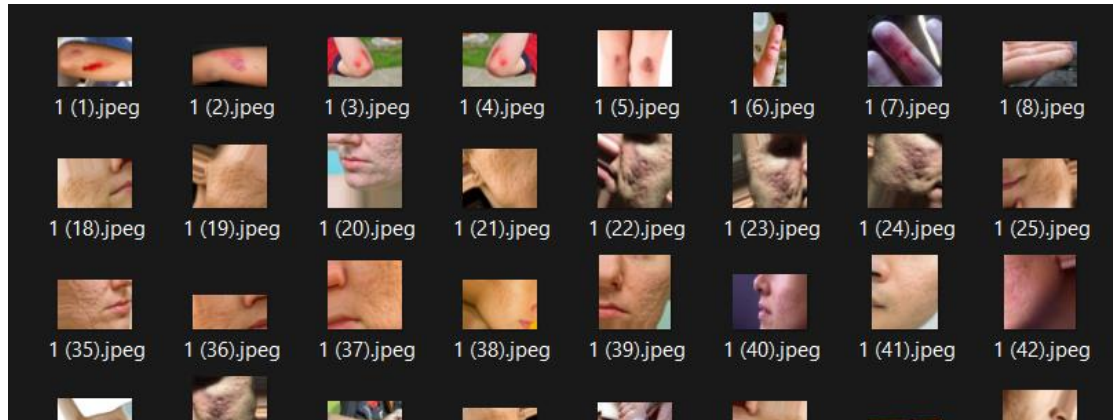
4.1.1 Perangkat Keras dan Perangkat Lunak

Spesifikasi perangkat keras dan perangkat lunak yang digunakan untuk membangun aplikasi ini adalah :

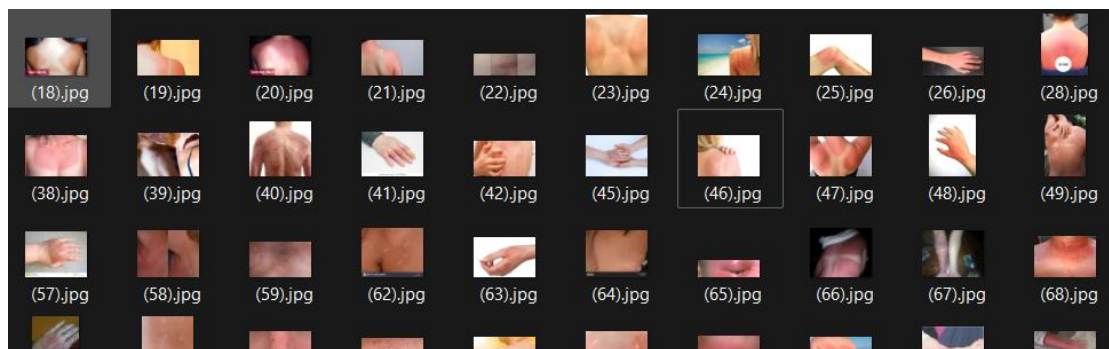
1. *Komputer Windows Based*
2. *Processor AMD® Ryzen™ 5 3600*
3. *Graphic Card NVIDIA Geforce RTX 2060 6GB*
4. *Storage SSD NVMe 512GB*
5. *RAM 16GB*
6. *Sistem Operasi Windows 11 Pro 64-bit*
7. *Android Studio*
8. *Google Colab Notebooks*
9. *Tensorflow Lite*

4.1.2 Implementasi Data

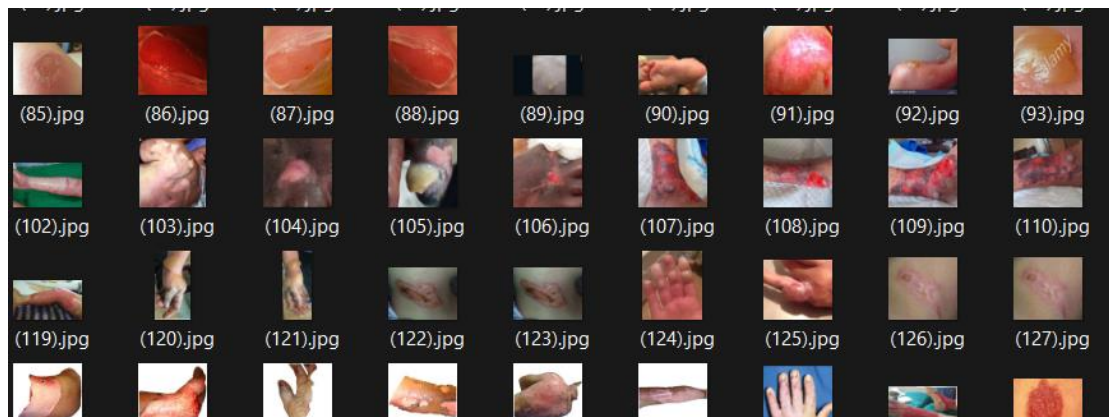
Data yang digunakan pada penelitian ini merupakan data yang diambil dari Klinik Perawatan Luka yang berada di Kota Medan dan *Platform Open Dataset* seperti Kaggle dan Roboflow dengan pengelompokkan data berdasarkan derajat dari luka bakar tersebut diantaranya Bukan Luka Bakar, Luka Bakar Derajat 1, Luka Bakar Derajat II dan Luka Bakar Derajat III. Adapun jumlah data yang diambil sekitar 1250 data dengan pembagian 200 data Citra bukan Luka Bakar, 350 data Citra Luka Bakar Derajat I, 350 data Citra Luka Bakar Derajat II, dan 350 data Citra Luka Bakar Derajat III



Gambar 4. 1 Data Gambar Bukan Luka Bakar



Gambar 4. 2 Data Gambar Luka Bakar Derajat I



Gambar 4. 3 Data Gambar Luka Bakar Derajat II



Gambar 4. 4 Data Gambar Luka Bakar Derajat III

4.1.3 Implementasi Aplikasi

Sistem diimplementasikan berbasis *mobile* menggunakan Android Studio. Setelah model dilatih, model akan disimpan kedalam format *TensorFlow Lite*. Berikut merupakan hasil implementasi desain antarmuka yang dirancang pada bab 3, yaitu :

1. Tampilan *Splashscreen*

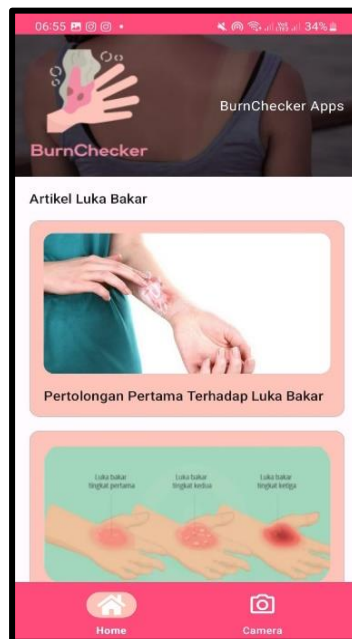
Tampilan *Splashscreen* merupakan tampilan awal dari aplikasi ini. Tampilan ini juga berfungsi sebagai tampilan “*loading*” sebelum masuk kedalam aplikasi.



Gambar 4. 5 *Splashscreen*

2. Tampilan Halaman Awal

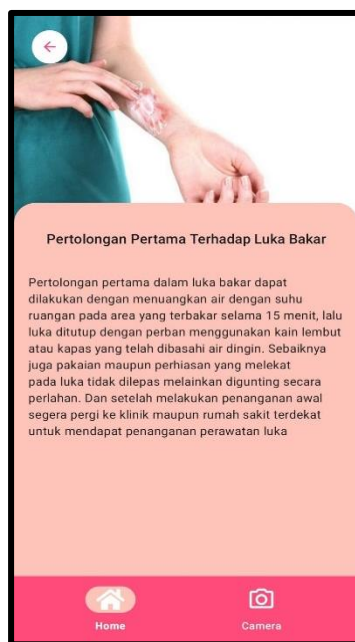
Tampilan Halaman Awal pada aplikasi ini berisikan tentang beberapa artikel seputar Luka Bakar yang diupdate secara berkala. Pada tampilan ini terdapat 3 buah *button*, diantaranya *button* “home” sebagai penanda halaman utama pada aplikasi, *button* “kamera” yang berfungsi untuk memulai proses klasifikasi, dan *button* article yang berfungsi untuk membuka artikel yang terdapat pada aplikasi. Adapun gambar tampilan Halaman awal dan Splashscreen dapat dilihat pada gambar di bawah ini :



Gambar 4. 6 Tampilan Utama

3. Tampilan Halaman Artikel

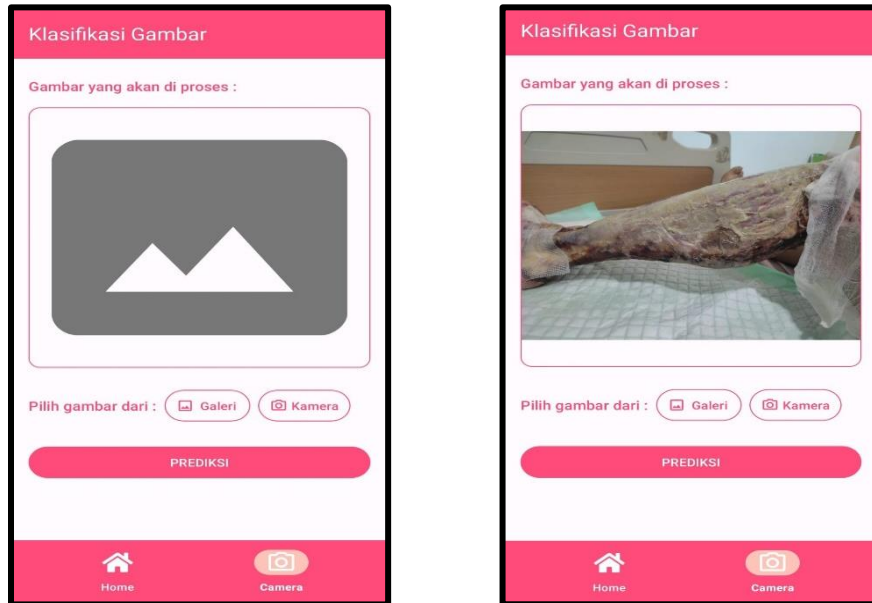
Halaman Artikel merupakan salah satu fitur yang terdapat pada aplikasi. Halaman artikel membahas segala hal yang berhubungan dengan luka bakar, baik dari menghindari terjadinya cedera pada luka hingga bagaimana cara merawat luka. Halaman artikel juga diupdate secara berkala guna memberikan informasi-informasi terbaru yang terdapat pada aplikasi, serta informasi yang tepat dan akurat juga mengenai informasi tentang luka bakar. Adapun gambar tampilan halaman artikel dapat dilihat pada gambar di bawah ini :



Gambar 4. 7 Tampilan Halaman Artikel

4. Tampilan Halaman Klasifikasi

Tampilan halaman klasifikasi dapat diakses dengan menekan button "Kamera" pada aplikasi. Pada halaman ini juga pengguna akan melakukan proses klasifikasi data. Pengguna dibebaskan untuk memilih metode pengambilan citra luka bakar, baik dengan memilih dan memasukkan gambar dari galeri dengan memilih tombol galeri maupun mengambil langsung menggunakan kamera handphone dengan memilih tombol kamera. Untuk pemilihan dari Galeri, pengguna akan langsung diarahkan ke Galeri pengguna serta memilih gambar yang akan diklasifikasikan. Untuk menggunakan Kamera, pengguna hanya perlu mengambil gambar dan gambar akan terinput. setelah pengguna menetapkan gambar yang akan diklasifikasi, gambar yang akan diklasifikasikan akan ditampilkan pada laman. Untuk melakukan proses klasifikasi, pengguna tinggal menekan tombol Prediksi dan aplikasi akan memproses gambar yang ingin diklasifikasi pengguna. Tampilan halaman klasifikasi dapat dilihat pada gambar di bawah ini :



Gambar 4. 8 Tampilan Halaman Klasifikasi

5. Tampilan Halaman Hasil

Halaman Hasil merupakan halaman yang keluar setelah dilakukannya proses klasifikasi pada Halaman Klasifikasi. Halaman ini berisikan keterangan dari gambar yang diinput, diantaranya gambar luka yang dideteksi, jenis luka bakar, diantaranya Luka Bakar Derajat 1, Luka Bakar derajat 2, atau Luka Bakar Derajat 3, serta berisikan juga keterangan dan penanganan pertama terhadap jenis luka tertentu. Adapun tampilan halaman hasil dapat dilihat pada gambar di bawah ini :



Gambar 4. 9 Tampilan Halaman Hasil

4.2 Prosedur Operasional

Prosedur Operasional merupakan salah satu tahapan dimana berfungsi untuk menjelaskan bagaimana menjalankan sistem yang telah dibuat. Sistem yang telah dibuat tadi terdiri atas tampilan *splashscreen*, tampilan *home*, tampilan halaman artikel, dan tampilan halaman hasil aplikasi.

Langkah awal, pengguna membuka aplikasi terlebih dahulu dengan menyentuh ikon aplikasi. Setelah itu, pengguna akan dihadapkan pada tampilan *Splashscreen* dimana tampilan ini merupakan tampilan awal dan menjadi *loading screen* aplikasi. Lalu pengguna akan dihadapkan pada laman utama tampilan aplikasi dimana terdiri dari Laman Artikel dan Laman pengujian. Ketika pengguna menekan salah satu artikel, maka laman artikel akan terbuka beserta isi artikel dari laman artikel tersebut. Artikel-artikel yang terdapat pada aplikasi terdiri atas kasus-kasus pada luka bakar, perawatan pada luka bakar baru, perawatan untuk menghilangkan bekas luka bakar, dan mengantisipasi agar tidak terjadinya kecelakaan dikarenakan luka bakar. Lalu ketika pengguna menekan tombol kamera, Halaman *testing* akan muncul. Terdapat dua buah button pada laman tersebut, sehingga pengguna dapat memilih apakah ingin mengambil gambar dari kamera ataupun mengambil gambar dari galeri pengguna. Setelah memilih gambar yang akan diklasifikasikan, pengguna hanya perlu menekan tombol prediksi dan aplikasi langsung diarahkan ke laman *result*, dimana aplikasi akan menampilkan derajat dari luka bakar tersebut serta bagaimana mengatasi luka bakar tersebut sesuai derajatnya.

4.3 Pelatihan Sistem

Pelatihan Sistem dilakukan dengan menerapkan Algoritma Faster R-CNN pada 3000 citra derajat luka bakar setelah melewati tahap augmentasi. Jumlah data tersebut berasal dari 1000 data citra derajat luka bakar yang merupakan 80% dari total data penelitian yang nantinya akan dikalikan 3 kali proses augmentasi. Pembagian data untuk pelatihan sistem dapat dilihat pada tabel 4.1

Pelatihan sistem menggunakan beberapa *layer*. *Layer* yang digunakan diantaranya *conv2d layer*, *maxpooling2d layer*, *batch normalization*, *dropout layer*, *flatten layer*, dan *dense layer*. *conv2d layer* merupakan *convolutional layer* yang memiliki fungsi untuk mengekstraksi fitur yang terdapat pada citra, *MaxPooling2D Layer* yang berfungsi untuk mengurangi dimensi data dan mempertahankan fitur-fitur

penting yang terdapat pada citra. *BatchNormalization* yang memiliki fungsi untuk melakukan proses normalisasi *batch* pada data guna membantu dalam pelatihan model yang lebih stabil dan konvergensi yang lebih cepat. Lalu, *flatten layer* digunakan sebelum *layer Dense* guna meratakan *tensor* multidimensi menjadi *tensor* satu dimensi. Dan *Dense Layer* yang berguna untuk menghubungkan setiap *neuron* yang ada di setiap lapisan.

Terdapat beberapa percobaan yang dilakukan pada penelitian ini dimana dilakukan percobaan dengan menggunakan *batch size* 16 dengan *epoch* 24 dengan memperoleh akurasi sebesar 91% (0.91250). Lalu dilakukan juga percobaan dengan mengubah *batch size* menjadi 32 dengan *epoch* 24 dan terdapat kenaikan akurasi yang tidak terlalu signifikan sebesar 92% (0.91875). Lalu dilakukan Kembali percobaan dengan menaikkan jumlah *batch size* menjadi 64 dengan jumlah *epoch* 32 dan memperoleh akurasi maksimal sebesar 95% (0.9469) Hal ini juga didukung dengan penelitian yang telah dilakukan oleh (Wasil et al, 2022) yang menyatakan bahwa nilai *epoch* yang besar dapat meningkatkan nilai akurasi. Selain itu (Alvera Kusuma, 2022) menyatakan bahwa nilai *epoch* yang besar menghasilkan tingkat akurasi yang besar juga.

Adapun hasil pengujian yang diperoleh dengan menggunakan *batch size* 64 dan *epoch* 32 dapat dilihat pada tabel 4.1

Tabel 4. 1 Hasil Pelatihan Sistem

epoch	loss	accuracy	val_loss	val_acc
1	1.3522	0.3321	1.2937	0.4313
2	1.2196	0.4471	1.1047	0.5031
3	1.0461	0.5386	0.9795	0.5500
4	0.9459	0.6051	0.8860	0.6156
5	0.8501	0.6562	0.7640	0.7406
6	0.7666	0.6868	0.7675	0.6938
7	0.6971	0.7214	0.6279	0.7500
8	0.6176	0.7610	0.6562	0.7563
9	0.5596	0.7829	0.5680	0.7844
10	0.4813	0.8248	0.5679	0.7969
11	0.4225	0.8321	0.5404	0.7937
12	0.3767	0.8580	0.5950	0.8188
13	0.3125	0.8860	0.3959	0.8531
14	0.2850	0.8956	0.4410	0.8281
15	0.2418	0.9106	0.4388	0.8719
16	0.2237	0.9186	0.4056	0.8750
17	0.1950	0.9362	0.3448	0.8906
18	0.1647	0.9435	0.2939	0.9031
19	0.1428	0.9488	0.3090	0.9156
20	0.1279	0.9525	0.4309	0.8562
21	0.1118	0.9634	0.4036	0.8969
22	0.1002	0.9674	0.3176	0.9094
23	0.0945	0.9674	0.3955	0.8969
24	0.0867	0.9734	0.3461	0.8906
25	0.0747	0.9727	0.4047	0.8938
26	0.0760	0.9761	0.2882	0.9156
27	0.0725	0.9761	0.2880	0.9219
28	0.0678	0.9791	0.3038	0.9156
29	0.0700	0.9747	0.4349	0.9062
30	0.0587	0.9814	0.1982	0.9469

Tabel 4.1 Hasil Pelatihan Sistem (Lanjutan)

epoch	loss	accuracy	val_loss	val_acc
31	0.0595	0.9797	0.2826	0.9156
32	0.0574	0.9781	0.3031	0.9281

Berdasarkan Tabel 4.1, dapat dilihat bahwa sistem mengalami peningkatan performa untuk mengenali setiap citra derajat luka bakar pada manusia. Hal ini dapat kita lihat dari angka *Loss* yang semakin berkurang disetiap prosesnya. *Loss* merupakan nilai yang mengukur seberapa jauh sebuah model mendapati kesalahan dalam memprediksi label yang terdapat pada data *training* dan memiliki tujuan juga untuk menekan angka *loss* selama proses pelatihan model. Dengan angka *Loss* yang kecil, performa dari sebuah model yang akan mengklasifikasikan sebuah citra akan jauh lebih baik. Selain *loss*, peningkatan performa juga dapat dilihat dari *accuracy*. *Accuracy* memiliki peran penting untuk mengukur seberapa jauh mana model dapat memprediksi data uji yang disajikan serta memprediksi benar sebuah data *training*. Nilai yang disajikan diatas dihitung sebagai persentase prediksi benar terhadap total jumlah sampel pelatihan serta menggambarkan seberapa baik model dalam mengklasifikasikan data *training*. Ada juga *Val_Loss* yang berfungsi untuk menilai kinerja model dengan data validasi, serta membantu memantau kinerja model terhadap data yang belum pernah dilihat pada proses *training*. Adapun tujuan dari *val_loss* sendiri ialah mencegah terjadinya *overfit* serta melakukan generalisasi dengan baik terhadap data yang baru. *Val_Loss* juga berfungsi untuk mencegah terjadinya *overfitting* pada proses pengujian data. Dan *Val_Accuracy* yang tidak jauh berbeda dengan *accucary* pada proses training, yaitu untuk memberikan gambaran sejauh mana model dapat memprediksi dengan benar terhadap data yang belum pernah dilihat sebelumnya

Pelatihan sistem ini menggunakan *optimizer Stochastic Gradient Descent (SGD)*, dengan menggunakan *initializer uniform* beserta *activation relu* untuk mendapatkan nilai terbaik dalam menentukan *Loss*, *Accuracy*, *Val_Loss*, *Val_Accuracy*. Pada tabel yang telah disajikan diatas dapat dilihat bahwa hasil validasi akurasi terbesar yang diperoleh dari pengujian ini terdapat pada *epochs* 30 dimana hasil dari pengujian tersebut sebesar 0.9469 (95%).

4.4 Pengujian Sistem

Tahapan Pengujian Sistem dilakukan dengan cara melakukan pengujian sistem yang telah selesai dirancang untuk melihat hasil kerja sistem yang dimulai dari proses *pre-processing* dan juga identifikasi menggunakan *Faster R-CNN* dalam mengklasifikasikan citra derajat luka bakar yang terdiri atas 4 tingkatan diantaranya, Bukan Luka Bakar, Luka Bakar Derajat 1, Luka Bakar Derajat II dan Luka Bakar Derajat III. Pengujian dari sistem ini menggunakan data *testing* sebanyak 140 data citra derajat luka bakar yang merupakan 10% dari jumlah total data yang digunakan. Adapun beberapa contoh hasil pengujian sistem terdapat pada tabel 4.2

Tabel 4. 2 Hasil Pengujian Sistem

No	Citra	Manual	Sistem	Status
1.		Luka Bakar Derajat I	Luka Bakar Derajat I	Benar
2.		Luka Bakar Derajat II	Luka Bakar Derajat II	Benar
3.		Luka Bakar Derajat II	Bukan Luka Bakar	Salah

Tabel 4.2 Hasil Pengujian Sistem

No	Citra	Manual	Sistem	Status
4.		Luka Bakar Derajat II	Luka Bakar Derajat II	Benar
5.		Bukan Luka Bakar	Bukan Luka Bakar	Benar
6.		Bukan Luka Bakar	Bukan Luka Bakar	Benar
7.		Bukan Luka Bakar	Bukan Luka Bakar	Benar
8.		Luka Bakar Derajat III	Luka Bakar Derajat III	Benar

Tabel 4.2 Hasil Pengujian Sistem

No	Citra	Manual	Sistem	Status
9.		Luka Bakar Derajat III	Luka Bakar Derajat III	Benar
10.		Luka Bakar Derajat I	Luka Bakar Derajat I	Benar
11.		Luka Bakar Derajat III	Luka Bakar Derajat III	Benar
12.		Luka Bakar Derajat II	Luka Bakar Derajat II	Benar
13.		Luka Bakar Derajat I	Luka Bakar Derajat II	Salah

Tabel 4.2 Hasil Pengujian Sistem

No	Citra	Manual	Sistem	Status
14.		Luka Bakar Derajat I	Luka Bakar Derajat I	Benar
15.		Luka Bakar Derajat III	Luka Bakar Derajat III	Benar
16.		Bukan Luka Bakar	Bukan Luka Bakar	Benar
17.		Luka Bakar Derajat III	Luka Bakar Derajat III	Benar
18.		Luka Bakar Derajat I	Luka Bakar Derajat I	Benar

Pada tabel 4.2 di atas, terdapat 18 contoh citra pengujian pada sistem dari keseluruhan hasil uji. Pengujian citra Luka Bakar dengan 125 citra luka bakar dapat dilihat pada lampiran 1 penelitian. Setelah dilakukan proses pengujian, dilakukan proses evaluasi terhadap model dengan menggunakan *confussion matrix*. Proses ini dilakukan guna mengukur evaluasi model terhadap kinerja dari sistem dalam melakukan proses klasifikasi. Hasil dari perhitungan *confussion matrix* dapat dilihat pada tabel dibawah ini.

Tabel 4. 3 Hasil Evaluasi Model Menggunakan Confussion Matrix

		Prediksi				
		Bukan Luka Bakar	Derajat I	Derajat II	Derajat III	Total
Aktual	Bukan Luka Bakar	19	0	1	0	20
	Derajat I	0	33	2	0	35
	Derajat II	1	1	33	0	35
	Derajat III	0	0	1	34	35
	Total	31	29	30	90	90

Hasil Evaluasi model dengan menggunakan *confussion matrix* pada tabel 4.3 diatas menampilkan data uji sebanyak 125 citra dengan 1 kelas sebanyak 20 citra Luka Bakar dan 3 kelas sebanyak 35 citra Luka Bakar. Pada data Bukan Luka Bakar terdapat data yang terbaca Luka Bakar Derajat II dikarenakan bentuk luka terlihat seperti ciri Luka Bakar Derajat II dimana citra tersebut merupakan Citra dari Luka Gesekan. Selanjutnya pada pengujian sistem untuk Luka Bakar Derajat I terdapat dua kesalahan prediksi dikarenakan citra yang ditangkap kurang tajam dan warna dari citra yang tertangkap terlihat semu serta pencahayaan yang kurang baik dalam pengambilan citra. Pada pengujian sistem terhadap Luka Bakar Derajat II terdapat dua kali kesalahan dalam mendeteksi citra dimana citra dari luka bakar tersebut dibaca bukan luka bakar dan citra luka bakar derajat I dikarenakan posisi pengambilan citra yang kurang pas dengan pencahayaan yang kurang baik dalam pengambilan gambar citra. Pada pengujian sistem terhadap Luka Bakar Derajat III terdapat juga satu kali kesalahan

dalam deteksi citra, dimana citra terbaca sebagai citra Luka Bakar Derajat II. Hal ini disebabkan oleh tampak citra yang ditangkap dari kamera sepintas terlihat seperti efek dari luka lepuh yang hanya mengikis bagian *dermis* pada kulit, sedangkan luka sebenarnya ialah luka yang cukup dalam.

Dari pengujian citra yang telah dilakukan diatas, dihasilkan nilai *true positive*(TP), *false positive*(FP), dan *false negative*(FN). Tabel nilai ini dapat dilihat pada tabel 4.4 berikut ini

Tabel 4. 4 Nilai TP, FP dan FN Mutu Citra Luka Bakar

	True Positive (TP)	False Positive (FP)	False Negative (FN)
Bukan Luka Bakar	19	1	1
Derajat I	33	2	1
Derajat II	33	2	4
Derajat III	34	1	0
Total	119	6	6

Pada tabel 4.4 menunjukkan terdapat 119 data yang memiliki nilai *True Positive* (TP), yang berarti bahwa ini merupakan data yang memiliki nilai positif dan diprediksi benar oleh sistem. Kemudian terdapat juga 6 data yang bernilai *False Positive* (FP) yang berarti bahwa data yang terbaca merupakan data bernilai negatif namun sistem memprediksi data tersebut sebagai data positif. Selanjutnya terdapat 6 data yang bernilai *False Negative* (FN) yang berarti bahwa sistem memprediksi data sebagai nilai yang lain. Dengan nilai-nilai tersebut, dapat dihitung nilai dai *Precision*, *Recall*, dan *F1-Score* untuk setiap datanya.

a. *Precision*

Rumus perhitungan *Precision* dapat dilihat pada Persamaan 2.1

$$Precision \text{ Bukan Luka Bakar} = \frac{19}{20} \times 100\% = 95\%$$

$$Precision \text{ Derajat I} = \frac{33}{35} \times 100\% = 94,28\%$$

$$Precision \text{ Derajat II} = \frac{33}{35} \times 100\% = 94,28\%$$

$$\text{Precision Derajat III} = \frac{34}{35} \times 100\% = 97,14\%$$

b. *Recall*

Rumus perhitungan *Recall* dapat dilihat pada Persamaan 2.2

$$\text{Recall Bukan Luka Bakar} = \frac{19}{20} \times 100\% = 95\%$$

$$\text{Recall Derajat I} = \frac{33}{34} \times 100\% = 97,05\%$$

$$\text{Recall Derajat II} = \frac{33}{37} \times 100\% = 89,18\%$$

$$\text{Recall Derajat III} = \frac{34}{34} \times 100\% = 100\%$$

c. *F1 – Score*

Rumus perhitungan *F1-Score* dapat dilihat pada Persamaan 2.3

$$\begin{aligned} \text{F1-Score Bukan Luka Bakar} &= 2 \times \frac{(0.95 \times 0.95)}{(0.95 + 0.95)} \times 100\% \\ &= 2 \times \frac{(0.9025)}{(1.9)} \times 100\% \\ &= 95\% \end{aligned}$$

$$\begin{aligned} \text{F1-Score Derajat I} &= 2 \times \frac{(0.94 \times 0.97)}{(0.94 + 0.97)} \times 100\% \\ &= 2 \times \frac{(0.9118)}{(1.91)} \times 100\% \\ &= 95\% \end{aligned}$$

$$\begin{aligned} \text{F1-Score Derajat II} &= 2 \times \frac{(0.94 \times 0.89)}{(0.94 + 0.89)} \times 100\% \\ &= 2 \times \frac{(0.83)}{(1.83)} \times 100\% \\ &= 90\% \end{aligned}$$

$$\begin{aligned} \text{F1-Score Derajat III} &= 2 \times \frac{(0.97 \times 1)}{(0.97 + 1)} \times 100\% \\ &= 2 \times \frac{(97)}{(1.97)} \times 100\% \\ &= 98\% \end{aligned}$$

Berdasarkan perhitungan yang telah dilakukan sebelumnya , maka dapat dibuat sebuah table nilai yang terdapat pada dibawah ini

Tabel 4. 5 Nilai Precission, Recall dan F1-Score

	Precision	Recall	F1-Score
Bukan Luka Bakar	95%	95%	95%
Derajat I	94,28%	97,05%	95%
Derajat II	94,28%	89,18%	90%
Derajat III	97,14%	100%	98%

Berdasarkan hasil uji sistem yang telah dilakukan menggunakan Aplikasi Klasifikasi Derajat Luka Bakar menggunakan metode *Faster Region Convolutional Neural Network*, maka dihasilkan nilai akurasi sistem melalui perhitungan di bawah ini :

$$\begin{aligned}
 Accuracy &= \frac{N_{\text{Benar}}}{N} \times 100\% \\
 Accuracy &= \frac{119}{125} \times 100\% \\
 &= 95,20\%
 \end{aligned}$$

Berdasarkan hasil perhitungan yang telah dilakukan dengan menggunakan rumus diatas, maka diperoleh akurasi dari penelitian ini sebesar 95% dengan menggunakan metode *Faster R-CNN* untuk mengklasifikasikan Tingkat Derajat Luka Bakar berdasarkan bentuk dan warna luka. Adapun factor yang menyebabkan terjadinya kesalahan terhadap sistem dalam mengidentifikasi yaitu dikarenakan adanya kemiripan antara ciri masing-masing luka bakar maupun adanya kemiripan dari Luka Bakar dengan jenis luka yang lainnya

Pengujian terhadap sistem sangat dibutuhkan guna mengetahui seberapa baik dan sesuai sistem yang telah dibuat untuk mereka yang bekerja dibidang Kesehatan serta menggeluti bidang Kesehatan sebagai pengguna utama dari aplikasi Klasifikasi Luka Bakar, serta kepada orang awam melalui penggunaan aplikasi serta pengisian kuisioner penilaian. Pengujian sistem yang dilakukan meliputi pengujian fitur yang

terdapat pada aplikasi. Adapun fitur yang terdapat pada aplikasi klasifikasi derajat luka bakar terdiri atas :

1. Fitur Klasifikasi Derajat Luka Bakar
2. Informasi Penyakit Luka Bakar
3. Artikel tersaji pada Aplikasi

Teknik pengumpulan data yang dilakukan pada penelitian ini adalah survei. Peneliti menyebarkan survei berupa kuisisioner yang disebarkan kepada pihak umum serta pihak terkait yang berhubungan langsung dengan luka bakar. Kuesioner ini berisi pernyataan- pernyataan yang telah dibuat dengan menggunakan *google form*.

Penyebaran link google form ini dapat dilakukan melalui whatsapp, instagram maupun media sosial lain yang mendukung penyebaran angket. Jawaban pihak yang telah menjawab/mengisi kuesioner dan menjadi responden akan dimasukkan ke dalam *google drive*.

Populasi serta sampel merupakan hal terpenting dalam melakukan penelitian, Populasi dan sampel merupakan satuan atau kelompok yang mempunyai bentuk atau ciri tertentu dan sengalah dipilih untuk memperoleh informasi yang dapat digunakan dalam suatu penelitian yang telah direncanakan (Arfatin dkk, 2021). Populasi dan sampel dalam penelitian ini ditentukan dengan berbagai pertimbangan, adapun rinciannya sebagai berikut :

a. Populasi

Populasi diartikan sebagai wilayah generalisasi yang terdiri atas yaitu obyek/subyek yang mempunyai kualitas serta karakteristik tertentu yang ditetapkan oleh peneliti untuk dipelajari serta ditarik kesimpulannya (Sugiyono, 2013). Dalam penelitian ini yang dijadikan populasi diantaranya pihak umum serta pihak tenaga kesehatan

b. Sampel

Teknik pengambilan sampel yang digunakan adalah teknik purposive sampling. Menurut (Fauzy, 2014), purposive sampling adalah suatu metode penentuan sampel yang mana sampel dipilih berdasarkan pengetahuan peneliti terhadap tujuan atau pertanyaan penelitian. Sampel yang dipilih dianggap cocok untuk diteliti dan dapat memberikan informasi yang relevan karena mempunyai ciri, ciri, kriteria atau atribut tertentu. Oleh karena itu, pengambilan sampel tidak

dilakukan secara acak. Jenis pengambilan sampel purposif yang digunakan adalah pengambilan sampel variasi maksimum, karena teknik ini mengambil sampel peserta dari sudut pandang yang berbeda, yaitu mereka yang tidak tahu tentang luka bakar dan mereka yang tahu. Adapun pihak umum yang terlibat dalam pengujian ini diantaranya 10 orang tenaga Kesehatan dan 5 orang Awam dengan total 15 responden. Para pengisi kuisisioner diminta untuk mencoba aplikasi klasifikasi derajat luka bakar serta diminta untuk memberikan penilaian berdasarkan skala *likert* Dimana skala *likert* melakukan penilaian dengan skala 1 (sangat negatif) sampai 5 (skala positif) dimana skala *likert* berfungsi untuk mengukur sikap, pendapat, hingga persepsi seorang atau kelompok orang terhadap suatu hal. Penilaian ini dilakukan terhadap dua aspek yakni *understandability* dan *functionality*. Penjelasan aspek *understandability* dan *functionality* adalah sebagai berikut :

a. *Understandabilty*

Understandabilty merupakan aspek yang menunjukkan Tingkat pemahaman pengguna atas *output* yang dikembalikan oleh aplikasi. Adapun lima skala penilaian yang digunakan pada aspek *understandabilty* diantaranya :

- Skala 1 : Menyatakan *output* dari aplikasi sangat sulit dipahami
- Skala 2 : Menyatakan *output* dari aplikasi sulit dipahami
- Skala 3 : Menyatakan *output* dari aplikasi cukup dipahami
- Skala 4 : Menyatakan *output* dari aplikasi mudah dipahami
- Skala 5 : Menyatakan *output* dari aplikasi sangat mudah dipahami

b. *Functionality*

Functionality merupakan aspek yang menunjukkan Tingkat kegunaan *output* yang dikembalikan oleh fitur aplikasi kepada pengguna. Adapun terdapat lima skala penilaian yang digunakan pada aspek *functionality* pada penilaian diantaranya :

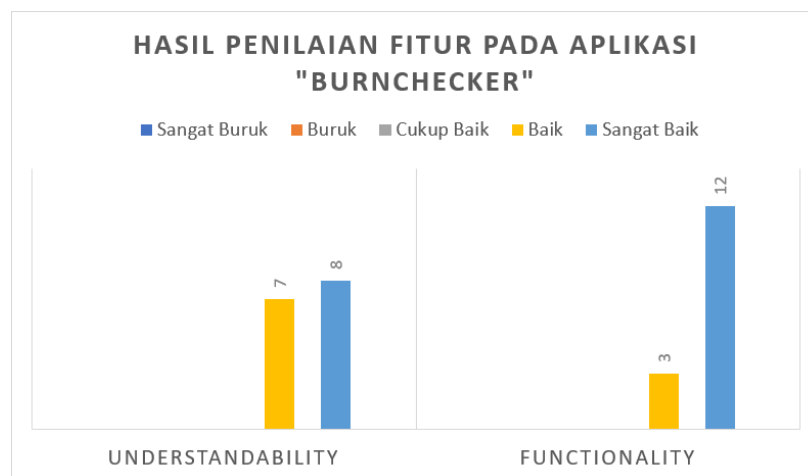
- Skala 1 : Menyatakan *output* dari aplikasi sangat tidak berguna
- Skala 2 : Menyatakan *output* dari aplikasi tidak berguna

- Skala 3 : Menyatakan *output* dari aplikasi cukup berguna
- Skala 4 : Menyatakan *output* dari aplikasi berguna
- Skala 5 : Menyatakan *output* dari aplikasi sangat berguna

Adapun pertanyaan yang diberikan kepada pengguna untuk melakukan penilaian terhadap fitur aplikasi yang dinilai dari aspek *understandability* dan *functionality* adalah sebagai berikut :

1. Pertanyaan terhadap penilaian fitur berdasarkan aspek *understandability*, diantaranya :
 - a. Skala 1 : Menyatakan *output* dari aplikasi sangat tidak berguna
 - b. Skala 2 : Menyatakan *output* dari aplikasi tidak berguna
 - c. Skala 3 : Menyatakan *output* dari aplikasi cukup berguna
 - d. Skala 4 : Menyatakan *output* dari aplikasi berguna
 - e. Skala 5 : Menyatakan *output* dari aplikasi sangat berguna
2. Pertanyaan terhadap penilaian fitur berdasarkan aspek *functionality*, diantaranya :
 - a) Skala 1 : Menyatakan *output* dari aplikasi sangat tidak berguna
 - b) Skala 2 : Menyatakan *output* dari aplikasi tidak berguna
 - c) Skala 3 : Menyatakan *output* dari aplikasi cukup berguna
 - d) Skala 4 : Menyatakan *output* dari aplikasi berguna
 - e) Skala 5 : Menyatakan *output* dari aplikasi sangat berguna

Gambar 4.10 menunjukkan hasil penilaian fitur terhadap aplikasi yang telah dinilai oleh 15 responden yang berkecimpung dibidang medis maupun orang awam.



Gambar 4. 10 Hasil Penilaian fitur Aplikasi “SkinBurn”

1. Hasil penilaian terhadap aspek *understandability* fitur aplikasi oleh pihak tenaga Kesehatan serta pihak umum didapat 7 orang menyatakan bahwa aplikasi sudah baik memberikan informasi kepada pengguna, dan 8 orang menyatakan bahwa aplikasi sudah baik dalam memberikan informasi kepada pengguna
2. Hasil penilaian terhadap aspek *functionality* fitur aplikasi baik dari pihak tenaga Kesehatan serta pihak umum didapat 12 orang sangat puas dalam mengklasifikasikan penyakit luka bakar dan 3 orang menyatakan bahwa aplikasi sudah baik dalam memberikan hasil terhadap derajat luka bakar

Berdasarkan hasil penilaian terhadap fitur aplikasi oleh 15 orang baik dari pihak tenaga Kesehatan serta pihak umum menggunakan skala likert dan didasarkan terhadap 2 buah aspek yakni aspek *understandability* dan aspek *functionality* terhadap fitur yang terdapat pada aplikasi, dapat disimpulkan bahwa aplikasi yang dihasilkan pada penelitian ini memiliki kemudahan pemahaman serta fungsionalitas yang baik.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan penelitian yang sudah dilakukan didapatkan kesimpulan sebagai berikut:

1. Algoritma yang digunakan pada penelitian ini berupa Algoritma *Faster Regional Convolutional Neural Network* dapat digunakan serta mampu untuk mengklasifikasikan jenis luka terkhususnya jenis luka bakar.
2. Algoritma *Faster R-CNN* yang diterapkan untuk mengklasifikasikan Derajat Luka Bakar menghasilkan sistem dengan performa yang cukup memuaskan dan mendapatkan tingkat akurasi sebesar 95,20%.
3. Ukuran dari *batch size* dan *epochs* yang tepat dapat mengurangi nilai *loss* dan menaikkan nilai akurasi sehingga berpengaruh terhadap keberhasilan sistem dalam mengidentifikasi derajat luka bakar.
4. Kesalahan sistem dalam klasifikasi disebabkan oleh gambar yang kurang tajam, pengambilan yang kurang pas serta pencahayaan yang kurang memadai yang sangat mempengaruhi warna dari citra luka bakar.

5.2 Saran

Penulis menyadari bahwa penelitian ini masih perlu dikembangkan. Untuk itu adapun beberapa saran guna meningkatkan penelitian selanjutnya yaitu:

1. Diharapkan untuk membuat sebuah sistem yang mampu mendeteksi lebih baik dengan akurasi yang lebih bagus terhadap citra derajat luka bakar
2. Diharapkan juga pada penelitian selanjutnya dibangun sebuah sistem klasifikasi derajat luka bakar secara *real time*.
3. Diharapkan untuk dapat menambahkan variasi dan jumlah data uji yang lebih banyak guna menambahkan ciri dan perbedaan yang lebih bervariasi terhadap jenis luka bakar seperti persentase kerusakan dan wilayah lukanya, dan
4. Diharapkan pada penelitian berikutnya menggunakan algoritma selain *Faster R-CNN* dalam proses klasifikasi derajat luka bakar, sehingga dapat dibandingkan serta disimpulkan baik nilai akurasi klasifikasi penelitian ini dengan algoritma baru lainnya.

DAFTAR PUSTAKA

- Ashish, Sharma. (2021). Skin Burn Detection Using Machine Learning, Advances in Intelligent Systems and Computing book series (AISC, volume 1388) Springer, Singapore. ISBN : 978-981-16-2597-8
- Vanathi, B. *et al.* (2020). Densemask RCNN: A Hybrid Model for Skin Burn Image Classification and Severity Grading. *Neural Process Lett* 53, 319–337 (2021). doi :10.1007/s11063-020-10387-5
- R. Ezhilarasi and P. Varalakshmi, "Tumor Detection in the Brain using Faster R-CNN," 2018 2nd International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC)I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC), 2018 2nd International Conference on, Palladam, India, 2018, pp. 388-392, doi: 10.1109/I-SMAC.2018.8653705.
- Hartanto, C.A., Wibowo, A.(2020). Development Of Mobile Skin Cancer Detection using Faster R-CNN and MobileNet v2 Model. 2020 7th International Conference on Information Technology, Computer, and Electrical Engineering (ICITACEE), ISBN :978-1-7281-7227-9
- Abubakar, A., Ugail, H., & Bukar, A.M. (2022). Assessment of Human Skin Burns: A Deep Transfer Learning Approach. *Journal of Medical and Biological Engineering vol.:* 40, ISSN :321–333
- Y. Yu, Q. Yang and L. Shaohan. (2020) "An Improved Faster R-CNN for Colorectal Cancer Cell Detection," 2020 IEEE 3rd International Conference on Electronics and Communication Engineering (ICECE), Xi'An, China, 2020, pp. 186-190, doi: 10.1109/ICECE51594.2020.9353044.
- Suha, S. A., Sanam, T F. (2022). A deep convolutional neural network-based approach for detecting burn severity from skin burn images. *Machine Learning With Applications Vol.9*. ISSN : 2666-8270
- Davies, M., Maguire, S., Okolie, C., Watkins, W., & Kemp, A. M. (2013). How much do parents know about first aid for burns?. *Burns*, 39(6), 1083-1090.
- Anggowarsito, Jose L. (2014). Luka Bakar Sudut Pandang Dermatologi. *Jurnal Widya Medika*, 2 (2). pp. 115-120, ISSN Online ISSN 2623-2723, Print ISSN: 2338-0373

- Morandi, *et al.* (2017). A small 'lick' will sink a great ship: Fulminant septicaemia after dog saliva wound treatment in an asplenic patient. *International wound journal*. 14. 10.1111/iwj.12752.
- Prihartini, T. S., & Andono, P. N. (2017). Deteksi Tepi Dengan Metode Laplacian of Gaussian Pada Citra Daun Tanaman Kopi.
- P. R. Dedhia, *et al* (2020). "Study on Abstractive Text Summarization Techniques," 2020 International Conference on Emerging Trends in Information Technology and Engineering (ic-ETITE), Vellore, India, 2020, pp. 1-8, doi: 10.1109/ic-ETITE47903.2020.087
- Wasil, M, Harianto, H, and Fathurrahman, F. 2 22. "Pengaruh Epoch Pada Akurasi Menggunakan Convolutional Neural Network Untuk Klasifikasi Fashion Dan Furniture." *Infotek : Jurnal Informatika dan Teknologi* 5(1) : 53-61.
- Girshick, Ross. 2 15. "Fast R-CNN." *Proceedings of the IEEE International Conference on Computer Vision 2015 Inter*: 1440–48.
- Ren, S., He, K., Girshick, R., & Sun, J. (2017). Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39, 1137-1149.
- Saputra, D. (2023). Tinjauan Komprehensif tentang Luka Bakar: Klasifikasi, Komplikasi dan Penanganan. *Scientific Journal*, 2(5), 197–208. <https://doi.org/10.56260/sciena.v2i5.113>
- Sanjaya, J., & Ayub, M. (2020). Augmentasi Data Pengenalan Citra Mobil Menggunakan Pendekatan Random Crop, Rotate, dan Mixup. *Jurnal Teknik Informatika Dan Sistem Informasi*, 6(2). <https://doi.org/10.28932/jutisi.v6i2.2688>
- Evers, L. H., Bhavsar, D., & Mailänder, P. (2010). The biology of burn injury. *Experimental dermatology*, 19(9), 777-783.
- Gupta, N., & Nusbaum, J. (2018). Points & Pearls: Emergency department management of patients with thermal burns. *Emergency medicine practice*, 20(2), e1-e2.
- Ghoury, Shekofa & Sungur, Cemil & Durdu, Akif. (2019). Real-Time Diseases Detection of Grape and Grape Leaves using Faster R-CNN and SSD MobileNet Architectures.