

Pengembangan Metode Identifikasi untuk Pra-Diagnosis  
Penyakit Kulit Manusia melalui Pemanfaatan Algoritma  
Kecerdasan Buatan dalam Optimalisasi Aplikasi  
Medical Dermatology

# **BAB 1**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Kulit adalah organ terbesar dalam tubuh, dan memainkan peran penting dalam pertahanan terhadap ancaman lingkungan seperti bakteri, virus, dan zat berbahaya. Penyakit kulit merupakan masalah kesehatan yang tersebar luas dan menyerang orang-orang dari segala usia, dan penyakit ini dapat disebabkan oleh berbagai faktor seperti genetika, gaya hidup, dan paparan lingkungan. Kelainan kulit merupakan hal yang umum terjadi di seluruh dunia, termasuk di Indonesia dimana penyakit ini menimbulkan tantangan besar terhadap kesehatan masyarakat karena tingginya angka kejadiannya (J. Zhang et al., 2023).

Indonesia adalah rumah bagi lebih dari 270 juta orang yang terkena berbagai penyakit kulit, menjadikannya salah satu negara yang paling terkena dampaknya secara global. Penyakit kulit yang umum termasuk psoriasis, eksim, acne vulgaris, vitiligo, dan melanoma. Penyakit-penyakit ini tidak hanya menimbulkan gejala fisik tetapi juga mempunyai dampak psikologis yang mendalam, mempengaruhi kualitas hidup pasien dan interaksi sosial. Diagnosis yang tepat dan efisien dari penyakit kulit menjadi kunci dalam penanganan yang efektif. Namun, proses diagnosis yang tergantung pada pengalaman klinis dan pengetahuan medis yang mendalam dari dermatolog sering kali membatasi akses terhadap perawatan yang berkualitas.

Diagnosis penyakit kulit secara tradisional memerlukan pemeriksaan langsung oleh seorang dokter spesialis kulit, yang kemudian dianalisis berdasarkan pengalaman klinis dan pengetahuan medis yang dimilikinya. Dengan perkembangan teknologi, terutama di bidang kecerdasan buatan (artificial intelligence atau AI), ada potensi untuk mengoptimalkan proses diagnosis penyakit kulit. Dalam beberapa tahun terakhir, kemajuan dalam bidang kecerdasan buatan

telah membuka peluang baru dalam peningkatan diagnosis penyakit kulit. Pemanfaatan algoritma kecerdasan buatan telah terbukti efektif dalam mendukung identifikasi penyakit kulit, yang dapat meningkatkan akurasi dan kecepatan diagnosis. Namun, tantangan utama yang masih dihadapi adalah integrasi teknologi ini ke dalam praktik klinis sehari-hari dengan cara yang efisien dan aman.

Penelitian sebelumnya telah menyoroti potensi pengembangan metode identifikasi pra-diagnosis penyakit kulit melalui pemanfaatan algoritma kecerdasan buatan. Pengenalan metode ini diharapkan dapat mengoptimalkan aplikasi medical dermatology dengan memperkenalkan pendekatan yang sistematis dan terstruktur dalam pengenalan gejala penyakit kulit. Pada penelitian (Wei et al., 2018) membahas tentang pengenalan penyakit kulit berbasis Image Color dan fitur tekstur. Metode yang digunakan meliputi GLCM untuk analisis tekstur, SVM untuk klasifikasi. Hasil akhir fitur warna dan tekstur yang digabungkan meningkatkan akurasi pengenalan hingga lebih dari 90%.

Pada penelitian (Fauzi Dzulfikar Wibowo et.al, 2022) membahas tentang Deteksi penyakit kulit di Indonesia menggunakan CNN dan pembelajaran ensemble. Dataset mencakup 1203 gambar penyakit kulit untuk klasifikasi. Metode Pembelajaran CNN dan Ensemble untuk deteksi gambar penyakit kulit. Validasi Silang K-Fold untuk pemisahan data dan hasil yang optimal. Pra-pemrosesan data dengan arsitektur CNN Resnet50 untuk pelatihan. Hasil akhir adalah Gambar RGB memiliki akurasi 49%, gambar Grayscale memiliki akurasi 47%.

Pada Penelitian (Raghav Agarwal et.al, 2023) membahas tentang Artificial Intelligence untuk deteksi gangguan kulit menggunakan model Deep Learning Convolutional Neural Network. Dataset mencakup 25000 gambar warna dari delapan gangguan kulit umum. Metode yang digunakan merupakan ResNet 152 turunan CNN dalam pemrosesan klasifikasi citra. Hasil akhir yang didapat pada penelitian ini dengan menggunakan ResNet152 adalah 74.24% pada validation and 73.01% pada test dataset.

Pada Penelitian (Yue Shen et.al, 2024) membahas tentang mengoptimalkan diagnosis penyakit kulit: memanfaatkan data komunitas daring dengan teknik ontrastive learning dan clustering. Metode Pendekatan contrastive learning untuk mempelajari representasi umum dari gambar yang tidak berlabel. Menggunakan metode clustering dengan gambar validasi standar untuk meningkatkan anotasi. Hasil akhir adalah model dapat ditransfer ke tugas baru dengan akurasi 61,76%.

Dan pada Penelitian (Archana et.al, 2019) membahas klasifikasi model deteksi penyakit kulit menggunakan pemrosesan gambar untuk aksesibilitas pedesaan. Transformasi DCT, DWT, dan SVD digunakan untuk deteksi penyakit kulit. Hasil akhir akurasi adalah 80 persen.

Pada penelitian ini mengusulkan pendekatan baru terhadap pra-diagnosis penyakit kulit manusia di Indonesia melalui pengembangan metode identifikasi berbasis AI yang memanfaatkan data klinis dan non-klinis. Dengan mengintegrasikan data dari berbagai sumber, termasuk rekam medis pasien, informasi genetik, paparan lingkungan, dan kebiasaan gaya hidup, metode yang diusulkan ini bertujuan untuk meningkatkan akurasi dan kecepatan diagnosis penyakit kulit, sehingga memberikan hasil yang lebih baik bagi pasien. Metode yang diusulkan juga mempertimbangkan karakteristik unik penduduk Indonesia, termasuk variasi penyakit kulit dan preferensi pasien, untuk memastikan relevansi dan efektivitasnya dalam konteks ini. Pendekatan ini merupakan perubahan signifikan dari metode diagnostik tradisional dan berpotensi merevolusi bidang dermatologi medis di Indonesia dan berbagai negara lain.

Dengan demikian, penelitian ini bertujuan untuk mengisi kesenjangan pengetahuan dengan mengembangkan pendekatan ilmiah yang lebih sistematis dalam pra-diagnosis penyakit kulit menggunakan algoritma kecerdasan buatan. Melalui investigasi yang komprehensif, diharapkan dapat dihasilkan metode identifikasi yang efisien dan akurat, yang mampu mendukung praktik klinis dan non-klinis dalam menghadapi berbagai tantangan dalam diagnosis penyakit kulit.

Pengembangan metode identifikasi ini diharapkan dapat meningkatkan aksesibilitas terhadap perawatan kesehatan kulit, mengurangi waktu tunggu untuk konsultasi, serta memberikan kontribusi signifikan dalam peningkatan kualitas perawatan pasien. Selain itu, penelitian ini juga diharapkan dapat memberikan landasan bagi pengembangan teknologi medis yang lebih lanjut dalam bidang dermatologi, serta memberikan dampak positif dalam perluasan cakupan layanan kesehatan kulit secara global.

### **1.3 Rumusan Masalah Penelitian**

Berbagai penelitian yang telah dilakukan belum terdapat identifikasi penyakit kulit pada manusia dengan teknis pendekatan data klinis dan non-klinis. Dengan

mengintegrasikan data dari berbagai sumber, termasuk rekam medis pasien, informasi genetik, paparan lingkungan, dan kebiasaan gaya hidup serta image dari kulit yang terinfeksi virus penyakit kulit dengan bantuan analisis pengolahan citra berbasis komputer, padahal dalam mendiagnosa penyakit kulit pada manusia para ahli sering mengalami kesulitan untuk mendeteksi adanya perbedaan dimana setiap kali terdeteksi selalu menunjukkan hasil yang berbeda dengan diagnosis sebenarnya. Maka terdapat peluang untuk melakukan penelitian pada permasalahan ini yang menjadi topik penelitian. Merujuk pada penelitian yang telah banyak dilakukan, proses kerja rangkaian pembangunan sistem dalam mengidentifikasi adanya penyakit kulit untuk mengenalinya sebagai suatu jenis penyakit maupun kelainan pada kulit menggunakan pengolahan citra akan berisi tiga langkah utama yaitu; segmentasi citra, ekstraksi fitur dan klasifikasi objek. Problem yang ada pada segmentasi citra penyakit kulit adalah; (1) segmentasi pada wilayah terlihat tidak jelas karena citra penyakit kulit direpresentasikan dalam citra sangat kecil yang memiliki karakter yang khas yaitu hanya memiliki intensitas warna lebih terang sehingga sulit membedakan kemiripannya dengan beberapa penyakit kulit, dengan demikian akan sulit mendefinisikan bentuk penyakit kulit secara spesifik. (2) Ketidakteraturan bentuk dari kelainan kulit dan ukuran serta rendahnya intensitas kontras antar penyakit kulit yang juga mirip dengan jaringan kulit di sekitarnya. (3) Citra penyakit kulit memiliki resolusi rendah, kontras yang lemah, dan memiliki banyak noise yang dapat mengubah intensitas piksel sehingga klasifikasi maupun identifikasi menjadi tidak pasti. (4) Warna kulit pada manusia terbilang sangat berbeda antara manusia yang satu dengan lainnya sehingga warna pada kelainan kulit juga dapat berubah yang menyebabkan identifikasi penyakit kulit menjadi sangat sulit untuk dibedakan. (5) Identifikasi penyakit kulit di banyak penelitian hanya menunjukkan jenis penyakit kulit tanpa adanya deskripsi yang jelas untuk dilakukan penanganan dini. (6) Sistem identifikasi penyakit kulit pada umumnya hanya menggunakan data citra sebagai acuan identifikasi jenis penyakit kulit namun tidak menggunakan keluhan maupun rekam medis pasien sebagai identifikasi penyakit kulit secara dini. Padahal pra-diagnosis serta penanganan dini merupakan tindakan

pencegahan dan pengobatan yang dapat menyelamatkan pasien-pasien yang terkena kelainan pada kulit sebelum dilakukan tindakan lebih lanjut.

Maka diperlukan model algoritma identifikasi handal yang dapat mengidentifikasi penyakit kulit sebagai pra-diagnosis dengan menggabungkan dua metode pengolahan data yaitu pengolahan data citra dan pengolahan data teks untuk dapat menangani permasalahan tersebut. Berdasarkan batasan masalah yang telah ditetapkan, uraian rumusan masalah agar fokus penelitian lebih terarah dan sesuai dengan yang diharapkan dengan berbagai pendekatan metode yang telah diusulkan oleh beberapa peneliti dapat diuraikan sebagai berikut;

1. Bagaimana mengembangkan algoritma identifikasi Image dan Teks agar lebih baik dan dapat melakukan prediksi identifikasi terhadap image untuk mengenali teks hasil sebagai penyakit kulit serta penanganannya maupun sebaliknya teks untuk mengenali image sebagai penyakit kulit serta penanganannya?
2. Bagaimana membangun model identifikasi penyakit kulit berdasarkan gabungan pengolahan image dan teks agar tercipta hybrid modelling untuk mengidentifikasi penyakit kulit melalui image-teks maupun sebaliknya teks-image (Usulan Bi-Directional Image-Text Matching)?
3. Bagaimana membangun prototype system aplikasi berbantuan komputer yang tepat dan akurat untuk identifikasi adanya kelainan kulit yang diderita pasien sebagai tindakan pra-diagnosis sehingga dapat diidentifikasi berdasarkan jenis penyakit kulit menggunakan usulan Bi-Directional Image-Text Matching?

#### **1.4 Tujuan Penelitian**

Tujuan umum penelitian ini adalah mengembangkan prototype sistem identifikasi berbasis aplikasi yang dapat mengidentifikasi penyakit kelainan pada kulit manusia dan mengenalinya sebagai jenis penyakit kulit serta dapat

memberikan keterangan dan penanganan dalam pengobatan menggunakan metode pengolahan citra dan teks. Tujuan khusus penelitian adalah;

1. Mengembangkan metode identifikasi penyakit kulit sebagai pra-diagnosis dengan menggabungkan dua metode pengolahan data yaitu pengolahan gambar dan pengolahan teks menjadi Bi-Directional Image-Text Matching untuk mendapatkan hasil yang lebih akurat dalam identifikasi image to teks maupun sebaliknya teks to image.
2. Menghasilkan model identifikasi Bi-Directional Image-Teks Matching yang dapat digunakan untuk identifikasi penyakit kulit yang dapat digunakan untuk prototype system identifikasi penyakit kulit berbasis aplikasi sebagai pra-diagnosis pasien.



#### **2.14.1. Perbandingan Tinjauan Penelitian**

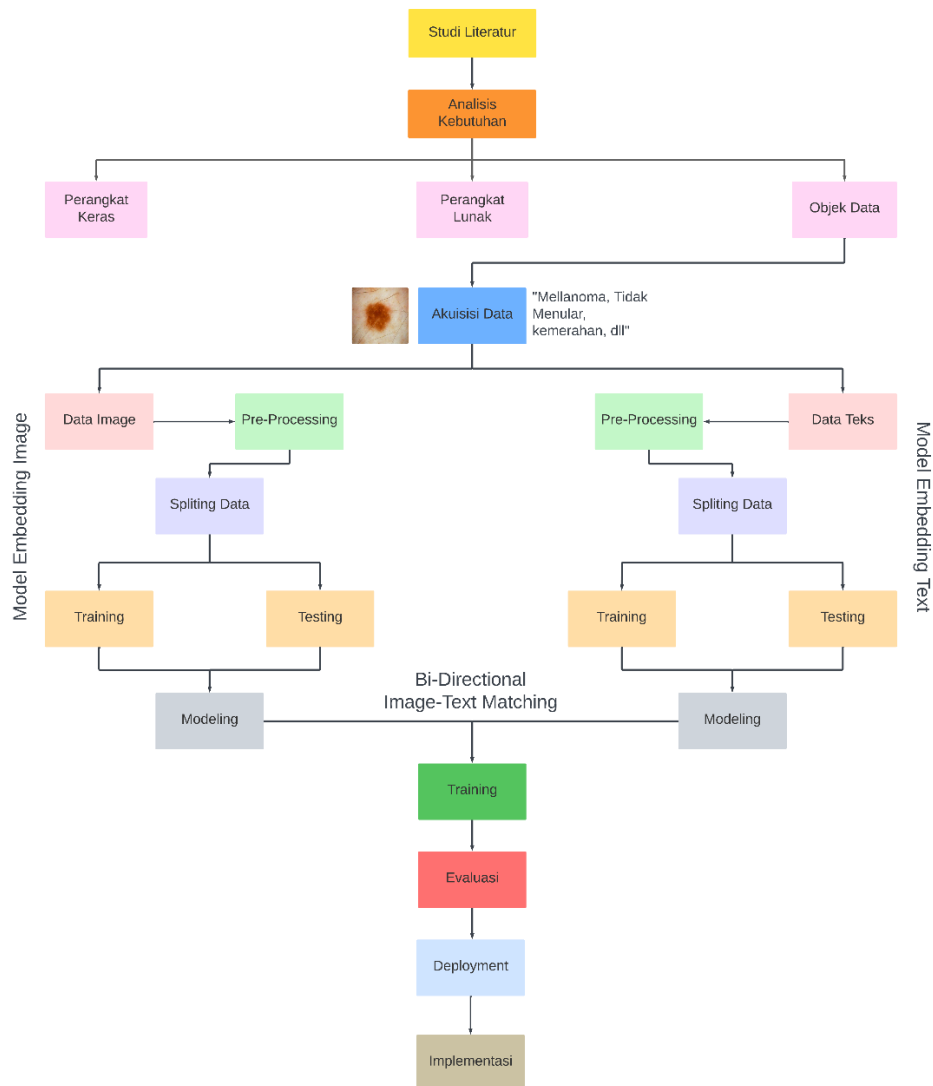
Dalam tinjauan penelitian ini, terdapat perbandingan mendalam antara berbagai pendekatan yang digunakan untuk pencocokan gambar-teks berbasis pembelajaran mendalam. Penelitian ini mengidentifikasi dan mengklasifikasikan pendekatan-pendekatan utama berdasarkan keselarasan antara gambar dan teks serta metode pembelajarannya. Beberapa penelitian terkait tentang pengolahan citra dan teks penyakit kulit untuk menentukan klasifikasi jenis penyakit pada kulit yang terdeteksi pada organ kulit menggunakan metode algoritma pengolahan image, teks maupun keduanya yang menjadi rujukan penelitian. Dalam keseluruhan perbandingan ini, penelitian ini memberikan wawasan yang komprehensif tentang kelebihan dan kekurangan masing-masing pendekatan, serta tantangan dan peluang pengembangan di masa depan, memberikan dasar yang kuat bagi peneliti baru untuk mengeksplorasi dan berkontribusi dalam bidang pencocokan gambar-teks berbasis pembelajaran mendalam. diuraikan pada tabel 2.4 berikut ini yaitu peneliti (Ebaid et al., 2023b; Mohammed & Al-Tuwaijari, 2021; Shen et al., 2024; Wei et al., 2018)

## BAB 3

# METODOLOGI PENELITIAN

Pada metodologi penelitian ini menjelaskan mengenai bagaimana proses dari analisis system, perancangan, dan analisis program yang dilakukan pada penelitian ini. Berikut analisis dan perancangan pada penelitian ini.

### 3.1 Tahapan Penelitian



Gambar 3. 1 Tahapan Penelitian (Dokumentasi Peneliti)

Tahapan penelitian dapat dilihat pada gambar 3.1. Tahapan penelitian yang dilakukan terdiri dari 9 tahapan, yaitu dimulai dari studi literatur sebagai dasar penelitian, analisis kebutuhan pada system yang akan dibangun, Pengumpulan dataset, preprocessing data, membangun model, training model, evaluasi model, deployment model, dan implementasi model yang telah dibuat ke dalam smartphone. Saat program telah dijalankan, program akan mengakuisisi dataset kemudian dataset akan melalui tahap preprocessing untuk menormalkan data kemudian setelah melalui tahap preprocessing selanjutnya mentraining dataset yang sudah didapatkan jika dataset berhasil dilatih dan juga divalidasi maka berlanjut ke tahap berikutnya yaitu tahapan testing dengan menerapkan model yang dibuat kedalam mobile phone atau smartphone. tahap selanjutnya jika camera telah menyala maka artinya sudah siap untuk mendeteksi objek jenis penyakit kulit. Pada tahap terakhir yaitu saat ada objek jenis penyakit kulit yang masuk atau terdeteksi oleh camera, maka citra tersebut sudah dapat dilakukan proses klasifikasi kemudian divalidasikan bahwa data tersebut sama dengan yang ada pada database untuk memunculkan label nama pada dataset serta memunculkan nilai confidence pada citra jenis penyakit kulit yang terdeteksi.

### **3.2 Analisis Kebutuhan**

Analisis kebutuhan merupakan menganalisis komponen yang diperlukan dalam pembuatan dan menjalankan program, Proses ini mencakup evaluasi, identifikasi, dan pemetaan kebutuhan dari berbagai perangkat yang terlibat dalam pembuatan system dan program pada penelitian ini. berikut analisis kebutuhan dari penelitian yang dibuat.

#### **3.2.1. Analisis Kebutuhan Perangkat Keras**

Perangkat keras yang digunakan dalam penelitian identifikasi penyakit kulit pada manusia menggunakan laptop Acer Predator Helios Neo 16 dan mobile phone atau smartphone Xiaomi Redmi Note 7 dengan bahasa pemrograman python, dengan spesifikasi yang dapat dilihat pada Tabel 3.1.

*Tabel 3. 1 Daftar Perangkat Keras*

No	Perangkat	Qty	Spesifikasi
1	Laptop Acer Predator Helios Neo 16	1	13 <sup>th</sup> Gen Intel(R) Core (TM) i7-13700HX (24 CPUs), ~ 2.1GHz. Random Acces Memory 8GB. Graphics Card Nvidia GeForce RTX 4060 8GB Solid State Drive 2TB.
2	Mobile Phone / Smartphone	1	Camera HD 48MP 16:9 (1280x720) f/1.8 (wide) Dual-LED flash, HDR, panorama RGB (Red, Green, Blue)

### 3.2.2. Analisis Kebutuhan Perangkat Lunak

Perangkat lunak yang digunakan dalam penelitian identifikasi penyakit kulit pada manusia menggunakan Operating System Windows Jupyter Lab dengan bahasa pemrograman python, dan Visual Studio sebagai text editor, yang dapat dilihat pada tabel 3.2.

*Tabel 3. 2 Daftar Perangkat Keras*

No	Perangkat Lunak	Version
1	Operating System	Windows 11 Pro Single Language 64-bit (10.0, Build 22631)
2	Python	3.7.0
3	Jupyter Notebook / labs	7.2.1 / 4.2.2
4	Visual Studio Code	May 2024 (version 1.90)

### 3.2.3. Analisis Objek

Program dengan menggunakan Metode Bi-Directional Image-Text Matching Deep Learning ini mempunyai beberapa objek yang diterapkan pada penelitian ini, yaitu:

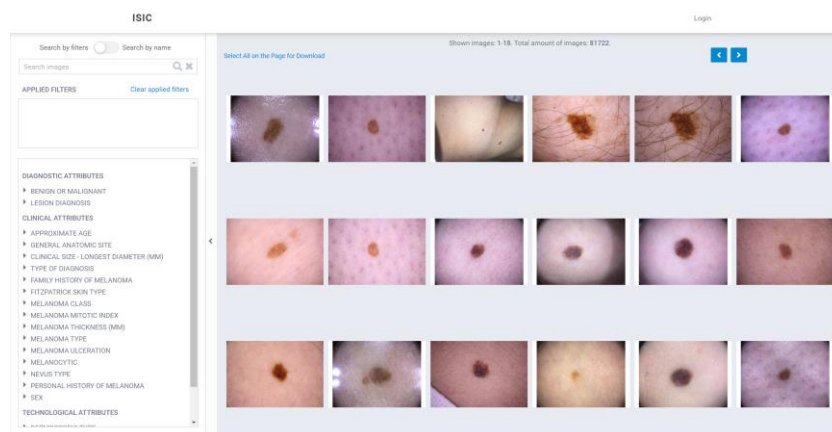
1. Identifikasi berbagai macam jenis penyakit kulit dengan memunculkan citra gambar yang didapat dan deskripsi mengenai penyakit kulit yang teridentifikasi dibawah citra gambar untuk setiap objek penyakit kulit yang terdeteksi, data yang digunakan memiliki variasi jenis penyakit kulit dengan kategori 2 penyakit kulit menular (Candidiasis dan Molluscum) dan 2 penyakit kulit tidak

menular (Eczhema dan Melanoma) dengan masing masing kelas memiliki 1000 citra penyakit kulit yang di dapat pada website international Dermnet NZ ([dermnetnz.org](http://dermnetnz.org), 2024) dan The International Skin Imaging Collaboration (ISIC) ([isic-archive.com](http://isic-archive.com), 2024).

2. Program identifikasi berbagai macam objek penyakit kulit pada manusia ditampilkan secara real-time menggunakan file upload kamera mobile phone.

### 3.3 Akuisisi Dataset

Proses akuisisi citra dilakukan dengan melakukan pengunduhan data dari berbagai sumber online international skin disease, seperti pada website **dermnetnz.org** dan **www.isic-archive.com** yang merupakan referensi gratis berbasis website untuk informasi tentang berbagai kondisi kulit. Website ini menyediakan gambar-gambar resolusi tinggi dari berbagai penyakit kulit, baik yang menular maupun tidak menular, serta memberikan deskripsi lengkap tentang penyakit tersebut meliputi gejala dan pengobatan. Citra yang diperoleh kemudian diseleksi berdasarkan fokus penelitian, yaitu identifikasi penyakit kulit menular (Candidiasis dan Molluscum) dan tidak menular (Eczhema dan Melanoma). Data citra yang digunakan berasal dari pasien dewasa dan anak-anak dengan kondisi kulit yang jelas menunjukkan gejala atau kelainan, seperti lesi atau ruam. Contoh citra yang akan digunakan pada penelitian seperti terlihat pada Gambar 3.2



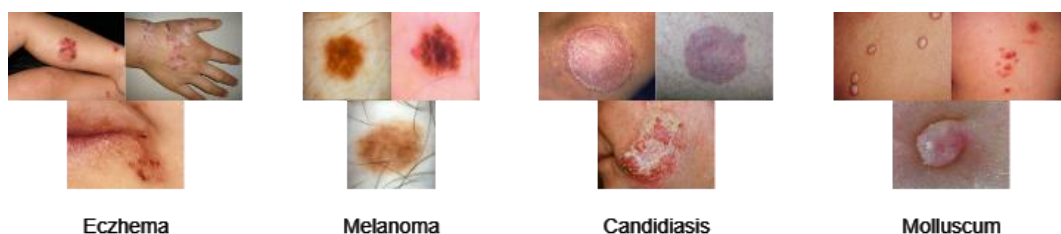
Gambar 3. 2: Citra penyakit kulit yang berasal dari website ISIC ([isic-archive.com](http://isic-archive.com), 2024)

### 3.3.1. Dataset Penyakit Kulit

Dataset pada penelitian ini dibagi menjadi 2 bagian yaitu 80% data training, dan 20% data testing objek jenis penyakit kulit. Dataset bersumber dari citra (data image) dan deskripsi (data teks) beberapa jenis penyakit kulit sejumlah 4000 citra dengan 4 jenis penyakit kulit yang terdiri dari Eczema, Melanoma, Candidiasis, dan Molluscum dengan memiliki 1000 citra berbeda setiap jenis penyakit kulit. Dari keempat jenis penyakit kulit tersebut dibagi menjadi 2 kelompok sebagai penyakit kulit menular dan tidak menular.

#### 3.3.1.1. Data Gambar

Data image ini mencakup berbagai jenis gambar yang menampilkan gejala dan karakteristik penyakit kulit yang digunakan pada penelitian ini (Eczema, Melanoma, Candidiasis, dan Molluscum) seperti ruam, bintik-bintik, lepuhan, atau lesi kulit lainnya. Ukuran citra asli yang didapat berukuran 294 x 222 yang akan diproses menjadi 256 x 256 sehingga ukuran gambar menjadi presisi dan pengambilan gambar diambil dari berbagai posisi yang berbeda sehingga posisi dalam proses training data akan mendapat banyak posisi pengenalan 1 jenis penyakit kulit dengan format citra JPEG (Joint Photographic Experts Group) serta pengambilan gambar dengan kamera. Penggunaan data gambar sangat penting dalam penelitian ini untuk membandingkan dan mempelajari pola visual yang terkait dengan berbagai penyakit kulit. Data image pada penelitian ini terdiri 4000 gambar dari 4 jenis penyakit kulit yaitu Eczema, Melanoma, Candidiasis, dan Molluscum yang dibagi menjadi 2 kelompok menular dan tidak menular. Data gambar dapat dilihat pada Gambar 3.3.



*Gambar 3. 3 Data gambar penyakit kulit*

### 3.3.1.2. Data Teks

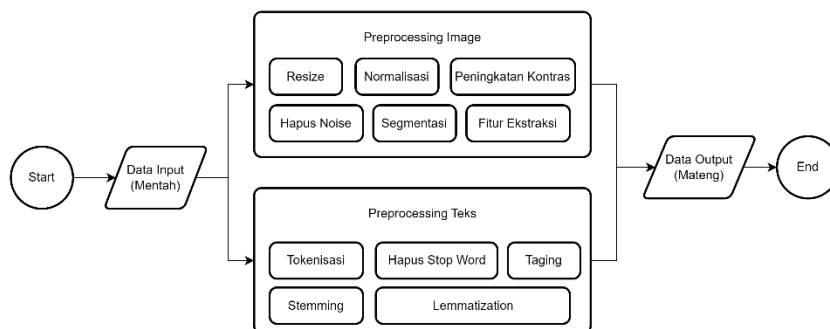
Data teks penyakit kulit merujuk kepada informasi tertulis yang berisi deskripsi dan karakteristik berbagai kondisi dermatologis. Data pada penelitian ini meliputi penjelasan tentang gejala-gejala khas seperti gatal-gatal, perubahan warna kulit, tekstur, dan lokasi lesi serta penjelasan mengenai cara penanganan maupun pengobatan yang dapat dilakukan pasien. Informasi ini penting untuk diagnosis dan pemahaman lebih lanjut tentang berbagai penyakit kulit seperti dermatitis, eksim, psoriasis, dan infeksi jamur kulit. Pada penelitian ini data teks diproses menggunakan teknik pengolahan bahasa alami atau natural language processing (NLP) untuk mengidentifikasi kata kunci dan pola yang terkait dengan setiap kondisi kulit. Berikut data teks yang digunakan pada penelitian ini dapat dilihat pada Tabel 3.3

*Tabel 3. 3 Data teks penyakit kulit*

Jenis	Tipe	Daerah + Populasi	Penanganan + Pengobatan	Obat	Karakteristik	Sebab/Akibat
Eczhema	Tidak Menular	Daerah perkotaan dengan polusi tinggi  Anak-anak, Orang dengan riwayat alergi	<b>Medis:</b> 1. Konsultasi dengan dokter kulit 2. Terapi kortikosteroid topikal 3. Fototerapi  <b>Non Medis:</b> 1. Menghindari alergen dan iritan 2. Menjaga kelembapan kulit	1. Kortikosteroid topikal 2. Emolien 3. Moisturizer 4. Antihistamin	1. Kulit kering dan bersisik 2. Kemerahan dan peradangan 3. Gatal intens 4. Ruam di wajah, leher, tangan, kaki 5. Lepuhan kecil mungkin pecah dan mengeluarkan cairan	<b>Sebab:</b> 1. Faktor genetic 2. Lingkungan (polusi, iklim) 3. Sistem imun  <b>Akibat:</b> 1. Gangguan tidur 2. Infeksi kulit sekunder 3. Gangguan psikologis (stres, kecemasan)
Melanoma	Tidak Menular	Daerah dengan paparan sinar matahari tinggi  Orang dengan kulit putih	<b>Medis:</b> 1. Operasi pengangkatan tumor 2. Terapi radiasi 3. Kemoterapi 4. Terapi target 5. Imunoterapi  <b>Non Medis:</b> 1. Menghindari paparan sinar matahari langsung 2. Menggunakan tabir surya	1. Kemoterapi (Dacarbazine) 2. Imunoterapi (Pembrolizumab, Nivolumab) 3. Terapi target (Vemurafenib)	1. Munculnya tahi lalat baru atau perubahan pada tahi lalat yang ada 2. Tepi tahi lalat tidak rata 3. Warna tahi lalat tidak seragam (campuran hitam, coklat, merah, biru, atau putih) 4. Diameter lebih dari 6 mm 5. Tahi lalat gatal atau berdarah	<b>Sebab:</b> 1. Paparan sinar UV berlebihan 2. Usia lanjut  <b>Akibat:</b> 1. Penurunan estetika wajah 2. Risiko metastasis
Candidiasis	Menular	Umumnya tersebar di seluruh wilayah Indonesia  Bayi (terutama bayi baru lahir)	<b>Medis:</b> 1. Krim atau salep antijamur (klotrimazol, mikonazol) 2. Obat antijamur oral dalam kasus yang parah  <b>Non Medis:</b> Menjaga kebersihan dan kekeringan area yang terkena	1. Krim antijamur (klotrimazol, mikonazol) 2. Obat antijamur oral (flukonazol) dalam kasus yang parah	1. Ruam merah, berlekuk, atau berbintik-bintik di area lipatan kulit 2. Gatal intens 3. Kelembapan yang berlebihan dan kerak putih pada kulit	<b>Sebab:</b> 1. Kelembapan berlebihan di area lipatan kulit 2. Faktor genetic  <b>Akibat:</b> 1. Infeksi sekunder jika tidak ditangani dengan baik 2. Ketidaknyamanan yang signifikan pada bayi atau individu yang rentan
Molluscum	Menular	Umumnya tersebar di seluruh wilayah Indonesia  Anak-anak, terutama usia 1-10 tahun	<b>Medis:</b> 1. Pemakaian kuretase atau pemetikan 2. Krioterapi (pengobatan dingin)  <b>Non Medis:</b> Monitoring dan kebersihan kulit	1. Krioterapi (pengobatan dingin) 2. Kuretase (pengangkatan dengan pisau bedah) 3. Obat antiviral dalam kasus yang parah	1. Lesi bulat atau kubah, berwarna putih atau merah muda 2. Ukuran berkisar dari 2-5 mm 3. Biasanya tidak menimbulkan rasa sakit atau gatal	<b>Sebab:</b> Infeksi virus Molluscum contagiosum  <b>Akibat:</b> 1. Potensi untuk penyebaran ke area kulit lain jika tidak ditangani dengan baik 2. Kecemasan atau ketidaknyamanan estetika pada pasien dan orang tua anak-anak

### 3.4 Pre-Processing Data

Pada tahapan ini data gambar penyakit kulit, preprocessing mencakup berbagai teknik seperti pengubahan ukuran gambar, normalisasi piksel, peningkatan kontras, penghapusan noise serta melakukan segmentasi dan fitur ekstraksi. Teknik ini bertujuan untuk meningkatkan kualitas gambar dan memastikan konsistensi data, sehingga fitur-fitur penting dapat diekstraksi dengan lebih efektif oleh algoritma analisis atau model kecerdasan buatan. Sedangkan pada data teks penyakit kulit, preprocessing melibatkan beberapa tahap seperti tokenisasi, penghapusan stop words, stemming, lemmatization, dan tagging. Langkah-langkah ini membantu dalam menyederhanakan teks, mengurangi dimensionalitas, dan meningkatkan efisiensi analisis teks. Dengan preprocessing yang tepat, data gambar dan teks menjadi lebih bersih dan terstruktur, memungkinkan model machine learning untuk menghasilkan prediksi yang lebih akurat dan andal. Tahapan preprocessing dapat dilihat pada Gambar 3.4.



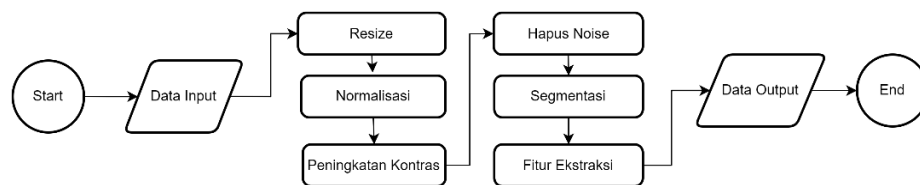
*Gambar 3. 4. Tahapan Preprocessing Data*

#### 3.4.1. Preprocessing Data Gambar

Proses ini melibatkan beberapa teknik utama. Pertama, pengubahan ukuran (resizing) gambar dilakukan untuk memastikan bahwa semua gambar memiliki dimensi yang seragam yaitu 256 x 256, yang penting untuk pengolahan batch dan integrasi dalam model. Kedua, normalisasi piksel diterapkan untuk mengatur nilai piksel dalam rentang tertentu, biasanya antara 0 dan 1, guna meningkatkan stabilitas dan kecepatan konvergensi model. Ketiga, peningkatan kontras (contrast



enhancement) dan penghapusan noise bertujuan untuk memperjelas fitur-fitur penting dalam gambar, seperti tepi atau tekstur, yang mungkin relevan untuk diagnosis penyakit kulit. Keempat, segmentasi data untuk memisahkan area kulit yang terkena penyakit dari bagian yang sehat. Kelima, fitur ekstraksi memungkinkan identifikasi karakteristik spesifik dari kondisi kulit, seperti ukuran dan bentuk lesi, distribusi warna, dan tekstur permukaan kulit. Preprocessing data gambar dapat dilihat pada Gambar 3.5.



*Gambar 3. 5 Preprocessing data gambar*

#### 3.4.1.1. Resizing Data

Pada tahap resize data ini bertujuan untuk mengubah ukuran citra penyakit kulit menjadi resolusi tetap 256x256 piksel. Langkah ini penting untuk memastikan bahwa semua citra memiliki ukuran yang konsisten sebelum digunakan dalam proses analisis data atau pelatihan model pembelajaran mesin. Skrip ini menggunakan pustaka OpenCV untuk memuat, mengubah ukuran, dan menyimpan citra. Dapat dilihat pada Algoritma 3.1.

---

##### **Algoritma 3.1** Algoritma Resize Citra

---

Input:

- Citra penyakit kulit dengan ukuran asli

Ouput:

- Citra penyakit kulit dengan ukuran sama 256x256

Proses:

1. Inisialisasi citra
  2. Periksa dan buat direktori output
  3. Iterasi Melalui Citra dalam Direktori Input
  4. Muat Citra
  5. Ubah Ukuran Citra
  6. Simpan Citra yang Telah Diubah Ukurannya
-

Ukuran dan bentuk citra hasil resizing disimpan pada folder output masing-masing penyakit kulit, yang selanjutnya akan diproses pada tahap berikutnya. Algoritma 3.1 diatas dapat dikonversi kedalam Pseudo-code 1 yang dapat diimplementasikan pada pemrograman Python.

---

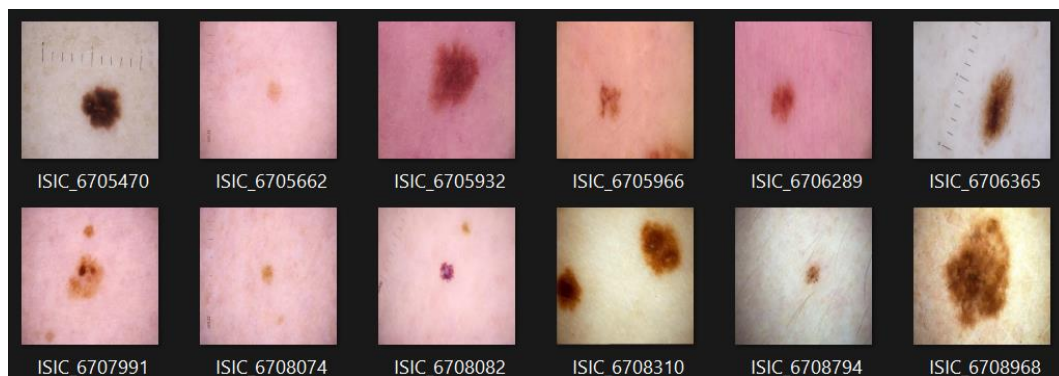
**Pseudocode 1. Resize Citra**

---

```
def resize_image(image, size=(256, 256)):  
    resized_image = cv2.resize(image, size,  
    interpolation=cv2.INTER_AREA)  
    return resized_image
```

---

Sehingga tampilan hasil program terlihat pada gambar 3.6 berikut. Seperti terlihat pada gambar, proses resize ditujukan pada ukuran gambar yang terlihat presisi dan sama yaitu 256x256.



*Gambar 3. 6. Hasil Resize Data Gambar*

#### **3.4.1.2. Normalisasi Data**

Pada tahapan ini data yang telah di resize pada tahap sebelumnya dinormalisasi. Melalui tahap normalisasi data bertujuan untuk mengubah nilai piksel citra ke dalam rentang yang konsisten, biasanya antara 0 dan 1 atau -1 dan 1. proses ini membantu dalam mengurangi variasi yang tidak diinginkan antar citra, seperti perbedaan pencahayaan dan kontras, sehingga fitur yang relevan menjadi lebih menonjol. Normalisasi dilakukan dengan membagi nilai piksel setiap citra dengan nilai maksimum piksel (biasanya 255 untuk citra 8-bit), sehingga setiap piksel memiliki nilai yang proporsional dalam rentang yang diinginkan. Langkah-langkah normalisasi data dapat di lihat pada Algoritma 3.2.

---

### Algoritma 3.2 Algoritma Normalisasi Citra

---

Input:

- Citra penyakit kulit hasil resize

Ouput:

- Citra penyakit kulit dengan hasil normalisasi

Proses:

1. Inisialisasi citra
  2. Muat data citra
  3. Ubah tipe data citra
  4. Normalisasi nilai piksel
  5. Simpan dan gunakan hasil normalisasi
- 

Citra hasil normalisasi disimpan, yang selanjutnya akan diproses pada tahap berikutnya. Algoritma 3.2 diatas dapat dikonversi kedalam Pseudo-code 2 yang dapat diimplementasikan pada pemrograman Python.

---

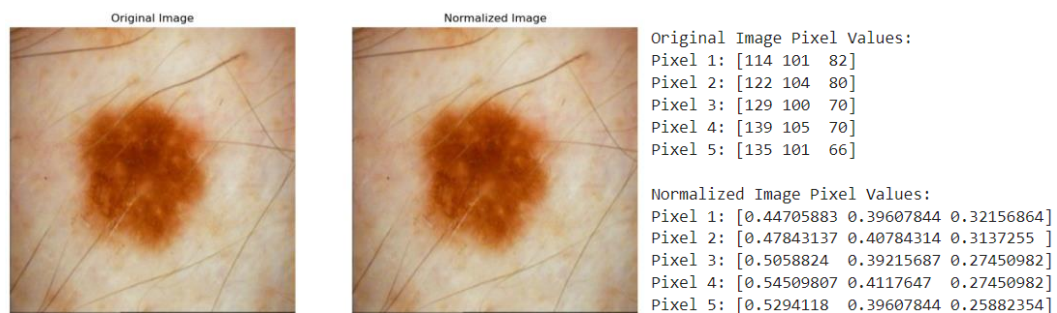
### Pseudocode 2. Normalisasi Citra

---

```
def normalize_image(image):  
    # Convert image to float32 type for normalization  
    image = image.astype(np.float32)  
    # Normalize the image  
    normalized_image = image / 255.0  
    return normalized_image
```

---

Sehingga tampilan hasil program terlihat pada gambar 3.7 berikut. Seperti terlihat pada gambar, proses normalisasi ditujukan mengubah nilai piksel citra ke dalam rentang yang konsisten, biasanya antara 0 dan 1 atau -1 dan 1.



*Gambar 3. 7 Hasil Normalisasi Data Citra*

### 3.4.1.3. Peningkatan Kontras Data

Pada tahap ini dilakuakn peningkatan kontras pada data citra yang telah di normalisasi bertujuan untuk meningkatkan perbedaan antara nilai intensitas piksel yang berdekatan. Dengan meningkatkan perbedaan antara nilai intensitas piksel, proses ini membantu dalam meningkatkan ketajaman citra dan membuatnya lebih mudah untuk dianalisis. Proses ini tidak hanya membuat citra lebih tajam dan lebih jelas, tetapi juga dapat meningkatkan kemampuan sistem analisis citra, seperti deteksi objek atau segmentasi yang lebih baik. Langkah-langkah peningkatan kontras dapat dilihat pada Algoritma 3.3.

---

**Algoritma 3.3** Algoritma Peningkatan Kontras

---

Input:

- Citra penyakit kulit hasil normalisasi

Ouput:

- Citra penyakit kulit dengan peningkatan kontras

Proses:

1. Inisialisasi citra
  2. Muat data citra
  3. Ubah tipe data citra
  4. Hitung Rata-rata Intensitas Piksel
  5. Peningkatan Kontras
  6. Simpan hasil Peningkatan Kontras
- 

Citra hasil peningkatan kontras disimpan, yang selanjutnya akan diproses pada tahap berikutnya. Algoritma 3.3 diatas dapat dikonversi kedalam Pseudo-code 3 yang dapat diimplementasikan pada pemrograman Python.

---

**Pseudocode 3.** Peningkatan Kontras Citra

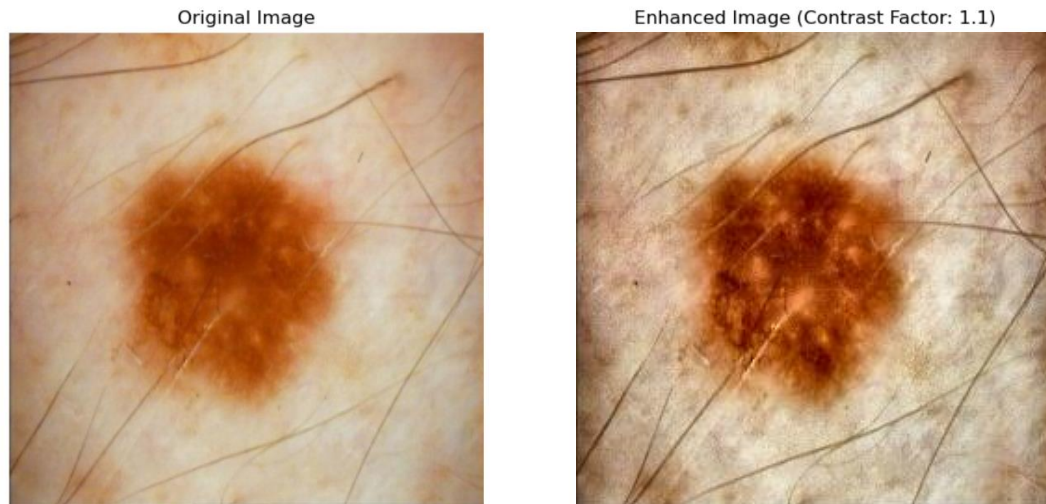
---

```
def normalize_image(image):  
    # Convert image to float32 type for normalization  
    image = image.astype(np.float32)  
    # Normalize the image  
    normalized_image = image / 255.0  
    return normalized_image
```

---

Sehingga tampilan hasil program terlihat pada gambar 3.8 berikut. Seperti terlihat pada gambar, proses peningkatan kontras ditujukan untuk meningkatkan perbedaan

antara nilai intensitas piksel yang berdekatan. Dengan meningkatkan perbedaan antara nilai intensitas piksel, proses ini membantu dalam meningkatkan ketajaman citra dan membuatnya lebih mudah untuk dianalisis.



*Gambar 3. 8 Hasil Peningkatan Kontras*

#### **3.4.1.4. Penghapusan Noise Data**

Pada tahap ini dilakukan penghapusan noise yang bertujuan untuk menghilangkan noise pada citra. Noise pada citra kulit dapat muncul karena berbagai alasan, seperti kualitas kamera yang rendah, kondisi pencahayaan yang buruk, atau bahkan gangguan selama pengambilan gambar. Untuk membersihkan gambar dari gangguan ini, digunakan berbagai teknik penghapusan noise. Filter median, misalnya, sangat baik untuk mengatasi noise jenis salt-and-pepper dengan menggantikan nilai setiap piksel dengan median dari piksel-piksel sekitarnya, sementara filter Gaussian menghaluskan gambar dengan mempertahankan tepi dan detail penting. Dengan menghilangkan noise, gambar kulit menjadi lebih bersih dan detail penting seperti warna, bentuk, dan tekstur lesi menjadi lebih jelas. Ini sangat membantu dokter atau sistem analisis otomatis untuk mengidentifikasi dan mengevaluasi kondisi kulit dengan lebih akurat, memastikan diagnosis dan rencana perawatan yang lebih efektif. Langkah-langkah penghapusan noise menggunakan median dan gaussian filter dapat dilihat pada Algoritma 3.4.

---

**Algoritma 3.4** Algoritma Penghapusan Noise

---

Input:

- Citra penyakit kulit hasil peningkatan kontras

Ouput:

- Citra penyakit kulit dengan penghapusan noise

Proses:

1. Inisialisasi citra
  2. Muat data citra
  3. Penghapusan Noise Menggunakan Filter Median
  4. Penghapusan Noise Menggunakan Filter Gaussian
  5. Tampilkan dan Simpan hasil
- 

Citra hasil penghapusan noise menggunakan median filter dan gaussian filter disimpan, yang selanjutnya akan diproses pada tahap berikutnya. Algoritma 3.4 diatas dapat dikonversi kedalam Pseudo-code 4 yang dapat diimplementasikan pada pemrograman Python.

---

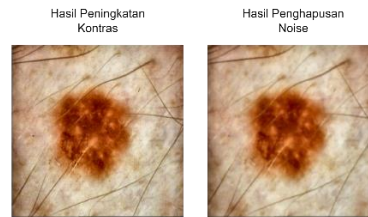
**Pseudocode 4.** Penghapusan Noise

---

```
def denoise_median(image, kernel_size=3):  
    denoised_image = cv2.medianBlur(image, kernel_size)  
    return denoised_image  
  
def denoise_gaussian(image, kernel_size=3):  
    denoised_image = cv2.GaussianBlur(image, (kernel_size,  
    kernel_size), 0)  
    return denoised_image
```

---

Sehingga tampilan hasil program terlihat pada gambar 3.9 berikut. Seperti terlihat pada gambar, proses penghapusan noise menggunakan gabungan median filter dan gaussian filter ditujukan untuk menghilangkan objek-objek yang tidak terpakai dengan menggunakan kernel rendah citra yang dihasilkan tidak terlalu mendapatkan blur yang sangat signifikan, sehingga objek suatu penyakit kulit masih dapat terlihat jelas tanpa adanya noise yang tidak terpakai. Dengan menghapus noise maka citra yang dihasilkan menjadi lebih bersih, proses ini membantu dalam meningkatkan fokus citra terhadap penyakit kulit dan membuatnya lebih mudah untuk dianalisis.



*Gambar 3. 9 Hasil Penghapusan Noise*

#### **3.4.1.5. Segmentasi Data**

Pada tahap ini dilakukan Segmentasi dengan thresholding atau penghapusan bagian yang tidak diperlukan seperti background untuk mendapatkan objek penyakit kulit yang digunakan pada penelitian serta menambahkan active contour untuk mendapatkan objek yang ditandai sebagai penyakit kulit. Proses ini melibatkan beberapa tahapan penting. Pertama, citra awal dimuat dan mungkin diubah menjadi citra skala abu-abu untuk mempermudah analisis intensitas piksel. Selanjutnya, nilai ambang dipilih atau dihitung berdasarkan karakteristik citra seperti histogram intensitas piksel. Pada tahap thresholding, piksel dalam citra yang melebihi nilai ambang akan diberi warna atau nilai putih (255), sementara piksel yang lebih rendah akan diberi warna atau nilai hitam (0), menghasilkan citra biner. Langkah-langkah segmentasi menggunakan thresholding atau penghapusan bagian yang tidak dibutuhkan dapat dilihat pada Algoritma 3.5.

---

#### **Algoritma 3.5** Algoritma Segmentasi

---

Input:

- Citra penyakit kulit hasil penghapusan noise

Ouput:

- Citra penyakit kulit hasil segmentasi thresholding

Proses:

1. Inisialisasi citra
  2. Muat data citra
  3. Konversi ke citra grayscale
  4. Tentukan nilai threshold
  5. Segmentasi dengan thresholding
  6. Inversi citra hasil thresholding
  7. Pemulihan warna asli
  8. Simpan Hasil
-

Citra hasil segmentasi menggunakan thresholding disimpan, yang selanjutnya akan diproses pada tahap berikutnya. Algoritma 3.5 diatas dapat dikonversi kedalam Pseudo-code 5 yang dapat diimplementasikan pada pemrograman Python.

---

**Pseudocode 5. Segmentasi**

---

```
def segment_with_threshold(image, threshold_value):
    _, segmented_image = cv2.threshold(image, threshold_value,
    255, cv2.THRESH_BINARY)
    return segmented_image

def invert_image(image):
    inverted_image = cv2.bitwise_not(image)
    return inverted_image

def find_contours(image):
    contours, _ = cv2.findContours(image, cv2.RETR_EXTERNAL,
    cv2.CHAIN_APPROX_SIMPLE)
    return contours

def draw_contours(image, contours):

    image_with_contours = image.copy()
    cv2.drawContours(image_with_contours, contours, -1, (0, 255,
    0), 2)
    return image_with_contours

def restore_color(original_image, inverted_segmented_image):
    mask = cv2.merge([inverted_segmented_image,
    inverted_segmented_image, inverted_segmented_image])

    inverted_mask = cv2.bitwise_not(mask)

    restored_image = cv2.bitwise_or(original_image,
    inverted_mask)

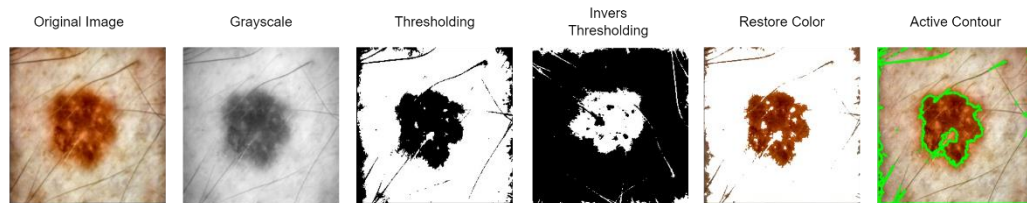
    return restored_image
```

---

Sehingga tampilan hasil program terlihat pada gambar 3.10 berikut. Seperti terlihat pada gambar, proses segmentasi menggunakan thresholding dan active contour ditujukan untuk menghilangkan objek-objek yang tidak digunakan dan memberi tanda pada objek yang digunakan untuk proses selanjutnya. Dengan menghapus nilai-nilai pada citra yang tidak terpakai maka citra yang dihasilkan menjadi lebih



bersih, proses ini membantu dalam menentukan focus objek terhadap penyakit kulit dan membuatnya lebih mudah untuk dianalisis.



*Gambar 3. 10 Hasil Segmentasi*

#### 3.4.1.6. Ekstraksi Fitur

Tahapan ini melibatkan pengambilan informasi relevan dari citra yang dapat digunakan untuk mengklasifikasikan dan mendiagnosis kondisi kulit. Setelah citra tersegmentasi dengan baik, langkah berikutnya yaitu mengekstraksi fitur-fitur yang relevan dari setiap area tersegmentasi. Fitur-fitur ini berupa tekstur, bentuk, dan warna yang dapat membedakan antara lesi kulit yang berbeda. Dalam beberapa kasus, tidak semua fitur yang diekstraksi diperlukan. Proses seleksi fitur membantu dalam memilih subset fitur terbaik yang paling bermakna untuk klasifikasi atau diagnosa yang akurat.

##### 3.4.1.6.1. Ekstraksi Fitur Warna

Tahap ini dimulai dengan memuat citra dalam format yang sesuai, seperti JPEG atau PNG, dan memisahkan informasi warna menjadi tiga kanal utama: merah (Red), hijau (Green), dan biru (Blue). Setiap kanal ini mewakili intensitas cahaya pada panjang gelombang yang berbeda dan memiliki rentang nilai dari 0 hingga 255 dalam skala 8-bit. Langkah-langkah ekstraksi fitur warna dapat dilihat pada Algoritma 3.6.

---

#### **Algoritma 3.6** Ekstraksi Fitur Warna

---

Input:

- Citra penyakit kulit hasil segmentasi

Ouput:

- Nilai Fitur Ekstraksi Warna

Proses:

---

- 
1. Inisialisasi citra
  2. Muat data citra
  3. Pisahkan Kanal Warna (R, G, B)
  4. Hitung Statistik Kanal
    - a) Rata-rata (Mean)
    - b) Standar Deviasi (Standard Deviation)
  5. Simpan Fitur
- 

Nilai hasil Ekstraksi fitur warna menggunakan RGB disimpan, yang selanjutnya akan diproses pada tahap berikutnya. Algoritma 3.6 diatas dapat dikonversi kedalam Pseudo-code 6 yang dapat diimplementasikan pada pemrograman Python.

---

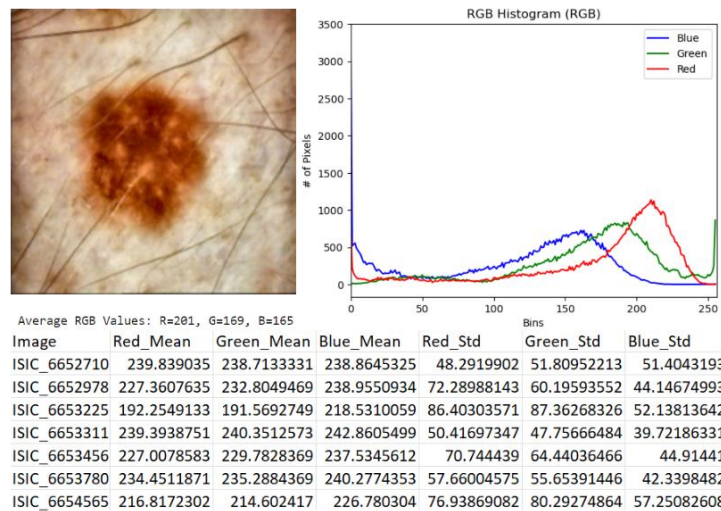
**Pseudocode 6. Ekstraksi Fitur Warna**

---

```
def extract_rgb_features(image):  
    # Split the image into RGB channels  
    b, g, r = cv2.split(image)  
  
    # Calculate mean and standard deviation for each channel  
    r_mean = b.mean()  
    g_mean = g.mean()  
    b_mean = r.mean()  
    r_std = b.std()  
    g_std = g.std()  
    b_std = r.std()  
  
    return r_mean, g_mean, b_mean, r_std, g_std, b_std
```

---

Sehingga tampilan hasil program terlihat pada gambar 3.11 berikut. Seperti terlihat pada gambar, proses ekstraksi fitur menggunakan RGB dan menunjukkan hasil histogram ditujukan untuk memisahkan informasi warna menjadi tiga kanal utama: merah (Red), hijau (Green), dan biru (Blue). Dengan mendapatkan nilai-nilai pada setiap kanal RGB maka informasi yang didapat akan semakin kompleks, proses ini membantu dalam menentukan setiap warna yang paling dominan pada objek terhadap penyakit kulit dan membuatnya lebih mudah untuk dianalisis.



*Gambar 3. 11 Hasil Ekstraksi Fitur Warna*

#### 3.4.1.6.2. Ekstraksi Fitur Bentuk

Tahapan ini dimulai dengan pra-pemrosesan citra untuk meningkatkan kualitas dan mempersiapkannya untuk ekstraksi fitur. Langkah pertama biasanya melibatkan segmentasi objek dari latar belakang, yang dapat dilakukan dengan metode seperti thresholding atau deteksi tepi. Setelah objek tersegmentasi, berbagai fitur geometris seperti luas, keliling, bentuk, dan orientasi dapat diekstraksi. Langkah-langkah ekstraksi fitur bentuk dapat dilihat pada Algoritma 3.7.

---

#### **Algoritma 3.7** Ekstraksi Fitur Bentuk

---

Input:

- Citra penyakit kulit hasil segmentasi

Ouput:

- Nilai Fitur Ekstraksi Bentuk

Proses:

1. Inisialisasi citra
  2. Muat data citra
  3. Ekstraksi Kontur
  4. Ekstraksi Fitur Geometris
  5. Simpan Hasil
-

Nilai hasil Ekstraksi fitur bentuk menggunakan Contour dan Geometris disimpan, yang selanjutnya akan diproses pada tahap berikutnya. Algoritma 3.7 diatas dapat dikonversi kedalam Pseudo-code 7 yang dapat diimplementasikan pada pemrograman Python.

---

**Pseudocode 7. Ekstraksi Fitur Bentuk**

---

```
def extract_shape_features(image):
    gray = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)

    _, thresh = cv2.threshold(gray, 0, 255, cv2.THRESH_BINARY +
cv2.THRESH_OTSU)

    contours, _ = cv2.findContours(thresh, cv2.RETR_EXTERNAL,
cv2.CHAIN_APPROX_SIMPLE)

    areas = []
    perimeters = []
    circularities = []
    eccentricities = []

    for contour in contours:
        area = cv2.contourArea(contour)
        perimeter = cv2.arcLength(contour, True)

        circularity = 4 * np.pi * area / (perimeter ** 2) if
perimeter > 0 else 0

        if len(contour) >= 5:
            ellipse = cv2.fitEllipse(contour)
            (center, axes, orientation) = ellipse
            major_axis = max(axes)
            minor_axis = min(axes)
            eccentricity = np.sqrt(1 - (minor_axis ** 2) /
(major_axis ** 2)) if major_axis > 0 else 0
        else:
            eccentricity = 0

        areas.append(area)
        perimeters.append(perimeter)
        circularities.append(circularity)
        eccentricities.append(eccentricity)

    avg_area = np.mean(areas)
    avg_perimeter = np.mean(perimeters)
```

---

---

```

avg_circularity = np.mean(circularities)
avg_eccentricity = np.mean(eccentricities)

return avg_area, avg_perimeter, avg_circularity,
avg_eccentricity

```

---

Sehingga tampilan hasil program terlihat pada gambar 3.12 berikut. Seperti terlihat pada gambar, proses ekstraksi fitur menggunakan bentuk contour dan geometris menunjukkan hasil nilai untuk setiap citra ditujukan untuk memisahkan informasi bentuk menjadi area, perimeter, circularity, dan exccentricity. Dengan mendapatkan nilai-nilai bentuk maka informasi yang didapat akan semakin kompleks, proses ini membantu dalam menentukan setiap bentuk yang paling dominan pada objek terhadap penyakit kulit dan membuatnya lebih mudah untuk dianalisis.

Image	Area	Perimeter	Circularity	Eccentricity
ISIC_6652710	630.9414894	28.13814949	0.098386711	0.210447387
ISIC_6652978	3038.8	71.68305145	0.111515514	0.094880682
ISIC_6653225	503.5934579	37.26525368	0.086272686	0.179074318
ISIC_6653311	1727.041667	44.64887538	0.158894203	0.095374179
ISIC_6653456	21619.66667	347.1323229	0.251475665	0.273191407
ISIC_6653780	967.7142857	36.20865767	0.081031771	0.110748532
ISIC_6654565	1919.981481	168.057358	0.256298259	0.605321865
ISIC_6654619	65024	1020.828427	0.784111882	0.985304283
ISIC_6654875	805.3472222	46.39103696	0.113139165	0.172580208
ISIC_6655383	933.8103448	98.2960011	0.233751268	0.41060552

*Gambar 3. 12 Hasil Ekstraksi Fitur Bentuk*

#### **3.4.1.6.3. Ekstraksi Fitur Tekstur**

Pada Tahapan ekstraksi fitur tekstur melibatkan beberapa langkah kunci untuk menggambarkan dan menganalisis tekstur citra secara sistematis. Tahap awal mencakup pemilihan GLCM sebagai metode utama untuk mengekstraksi fitur tekstur. Setelah GLCM terbentuk, tahap selanjutnya adalah ekstraksi fitur-fitur statistik dari matriks GLCM. Fitur-fitur ini mungkin mencakup energi, kontras, homogenitas, dan korelasi, yang masing-masing memberikan informasi tentang struktur dan pola tekstur dalam citra yang dianalisis. Langkah-langkah ekstraksi fitur tekstur dengan menggunakan metode GLCM sebagai acuan tekstur dapat dilihat pada Algoritma 3.8.

---

**Algoritma 3.8** Ekstraksi Fitur Tekstur

---

Input:

- Citra penyakit kulit hasil segmentasi

Ouput:

- Nilai Fitur Ekstraksi Tekstur

Proses:

1. Inisialisasi citra
  2. Muat data citra
  3. Pembentukan GLCM
  4. Normalisasi GLCM
  5. Ekstraksi Fitur Statistik
  6. Simpan Hasil
- 

Nilai hasil Ekstraksi fitur tekstur menggunakan GLCM disimpan, yang selanjutnya akan diproses pada tahap berikutnya. Algoritma 3.8 diatas dapat dikonversi kedalam Pseudo-code 8 yang dapat diimplementasikan pada pemrograman Python.

---

**Pseudocode 8.** Ekstraksi Fitur Tekstur

---

```
def calculate_glcm_features(image_path):  
    # Load gambar dan konversi ke grayscale  
    image = io.imread(image_path)  
    gray_image = color.rgb2gray(image)  
    gray_image = img_as_ubyte(gray_image) # Konversi ke tipe  
    data uint8  
  
    # Hitung GLCM dengan jarak dan arah yang ditentukan  
    distances = [1, 2, 3] # Jarak (d)  
    angles = [0, np.pi/4, np.pi/2, 3*np.pi/4] # Arah (θ)  
    glcm = graycomatrix(gray_image, distances=distances,  
        angles=angles, symmetric=True, normed=True)  
  
    # Ekstraksi fitur tekstur dari GLCM  
    contrast = graycoprops(glcm, 'contrast')  
    dissimilarity = graycoprops(glcm, 'dissimilarity')  
    homogeneity = graycoprops(glcm, 'homogeneity')  
    energy = graycoprops(glcm, 'energy')  
    correlation = graycoprops(glcm, 'correlation')  
  
    # Mengembalikan hasil fitur sebagai tuple  
    return (contrast.mean(), dissimilarity.mean(),  
        homogeneity.mean(), energy.mean(), correlation.mean())
```

---

Sehingga tampilan hasil program terlihat pada gambar 3.13 berikut. Seperti terlihat pada gambar, proses ekstraksi fitur menggunakan GLCM menunjukkan hasil nilai untuk setiap citra ditujukan untuk memisahkan informasi tekstur menjadi contrast, dissimilarity, homogeneity, energy dan correlation. Dengan mendapatkan nilai-nilai tekstur maka informasi yang didapat akan semakin kompleks, proses ini membantu dalam menentukan setiap tekstur yang paling dominan pada objek terhadap penyakit kulit dan membuatnya lebih mudah untuk dianalisis.

Image	Contrast	Dissimilarity	Homogeneity	Energy	Correlation
ISIC_6652710.jpg	1231.65463	8.447259836	0.844691317	0.787181691	0.762224524
ISIC_6652978.jpg	944.1924863	7.08935974	0.791081669	0.686836622	0.850783147
ISIC_6653225.jpg	3604.234732	26.91911272	0.336917064	0.147973877	0.715209388
ISIC_6653311.jpg	857.3144884	6.386740505	0.815364722	0.715389835	0.791955224
ISIC_6653456.jpg	505.5689637	4.482941353	0.828913533	0.77733306	0.931810018
ISIC_6653780.jpg	1647.801379	11.42836698	0.700051823	0.532937453	0.698027801
ISIC_6654565.jpg	2451.078761	16.59014094	0.561026828	0.345212699	0.783161192
ISIC_6654619.jpg	255.1038341	1.813032452	0.941287488	0.89935909	0.738916829
ISIC_6654875.jpg	2896.625517	22.42471018	0.310133985	0.122962482	0.838327483
ISIC_6655383.jpg	3235.334625	20.20172999	0.62546341	0.452300406	0.61042744
ISIC_6655446.jpg	2937.584838	22.46001243	0.402843556	0.212558734	0.775834964

*Gambar 3. 13 Hasil Ekstraksi Fitur Tekstur dengan GLCM*

### 3.4.2. Preprocessing Data Teks

Tahap pre-processing data teks dilakukan serangkaian langkah penting dalam pengolahan informasi teks yang bertujuan untuk membersihkan, merapikan, dan mempersiapkan data sebelum dilakukan analisis lebih lanjut. Proses ini krusial karena data teks sering kali tidak terstruktur dan dapat mengandung berbagai jenis noise atau informasi yang tidak relevan yang dapat mempengaruhi hasil analisis. Beberapa tahap yang dilakukan pada preprocessing data teks ini meliputi Pertama, **Tokenisasi** dilakukan untuk memecah teks menjadi unit-unit yang lebih kecil seperti kata-kata atau kalimat. Setelah itu, langkah **Pembersihan (cleaning)** dilakukan untuk menghilangkan elemen-elemen yang tidak relevan seperti, karakter khusus, atau token seperti stopwords yang tidak memberikan banyak informasi. Selanjutnya, **Stemming atau Lemmatisasi** Proses ini mengubah kata-kata menjadi bentuk dasarnya (lemmas) atau akar kata (stems) untuk mengurangi variasi kata yang memiliki arti yang sama. Contohnya,