

Klasifikasi Tingkat Keparahan Jerawat Menggunakan Metode K-Nearest Neighbor (KNN) dan Random Forest

Proposal Tugas Akhir

Kelas MK Penulisan Proposal (CAK4EAB2)

1301213102

GLORIA FLOURIN MAITIMU



Program Studi Sarjana Informatika

Fakultas Informatika

Universitas Telkom

Bandung

2024

Lembar Persetujuan

Klasifikasi Tingkat Keparahan Jerawat Menggunakan Metode K-Nearest Neighbor (KNN) dan Random Forest

Acne Severity Classification Using the K-Nearest Neighbor (KNN) Method and Random Forest

NIM : 1301213102

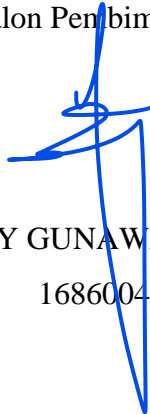
Gloria Flourin Maitimu

Proposal ini diajukan sebagai usulan pembuatan tugas akhir pada
Program Studi Sarjana Informatika
Fakultas Informatika Universitas Telkom

Bandung, 01 Januari 2025

Menyetujui

Calon Pembimbing 1



Dr. PUTU HARRY GUNAWAN, S.Si., M.Si., M.sc

16860043

DAFTAR ISI

LEMBAR PERSETUJUAN	1
DAFTAR ISI	2
DAFTAR TABEL	3
DAFTAR GAMBAR.....	4
ABSTRAK	5
1. PENDAHULUAN	6
1.1. Latar Belakang.....	6
1.2. Perumusan Masalah	7
1.3. Tujuan.....	8
1.4. Rencana Kegiatan	8
1.5. Jadwal Kegiatan.....	9
2. KAJIAN PUSTAKA	10
2.1. Studi Literatur	10
2.2. K-Nearest Neighbor	13
2.3. Random Forest	14
3. PERANCANGAN SISTEM	16
3.1. Perencanaan Sistem.....	16
3.2. Dataset.....	16
3.3. Preprocessing Dataset.....	17
3.4. Data Splitting.....	18
3.5. Object Detection (YOLOv8)	18
3.6. Pembangunan Model KNN dan Random Forest	19
3.7. Evaluasi Model	19
DAFTAR PUSTAKA	21

DAFTAR TABEL

<i>Tabel 1. Jadwal Kegiatan</i>	9
<i>Tabel 2. Tabel Studi Literatur.....</i>	10

DAFTAR GAMBAR

<i>Gambar 1. Random Forest</i>	14
<i>Gambar 2. Flowchart.....</i>	16
<i>Gambar 3. Contoh Dataset</i>	16
<i>Gambar 4. Pelabelan Dataset.....</i>	17
<i>Gambar 5. Arsitektur YOLOv8, visualization made by GitHub user RangeKin ..</i>	18

ABSTRAK

Jerawat atau *acne vulgaris* merupakan salah satu permasalahan kulit yang umum terjadi pada kalangan remaja baik laki-laki maupun perempuan. Penelitian ini bertujuan untuk mengembangkan model klasifikasi tingkat keparahan jerawat menggunakan algoritma *machine learning* yaitu *K-Nearest Neighbor* (KNN) dan *Random Forest* dengan bantuan YOLOv8 untuk mendeteksi objek. Tahapan yang dilakukan dalam penelitian ini yaitu pengumpulan dataset dari Kaggle yang terdiri dari 1833 gambar jerawat, melakukan *preprocessing dataset* dengan membersihkan, memberi label, mengubah ukuran gambar, melakukan augmentasi dan normalisasi, serta membagi dataset menjadi *data training* (80%) dan *data testing* (20%). Metode YOLOv8 diaplikasikan untuk mendeteksi jerawat dan selanjutnya metode KNN dan *Random Forest* diaplikasikan untuk mengklasifikasi jerawat menjadi 3 kategori yaitu ringan, sedang dan parah. Evaluasi model yang digunakan dalam penelitian ini adalah *Confusion Matrix* dengan membandingkan nilai *Accuracy*, *Precision*, *Recall* dan *F1-Score* dari kedua model yang telah dibangun. Hasil penelitian ini diharapkan dapat membantu para ahli di klinik kecantikan untuk mampu mengatasi masalah jerawat dengan memberikan penanganan yang tepat berdasarkan identifikasi tingkat keparahannya secara akurat.

Kata Kunci: Jerawat, Klasifikasi, Machine Learning, YOLOv8, K-Nearest Neighbor, Random Forest

1. PENDAHULUAN

1.1. Latar Belakang

Dalam era modern saat ini, dunia kecantikan mengalami perkembangan yang lumayan pesat. Hal ini ditandai dengan munculnya klinik-klinik kecantikan yang hadir di tengah-tengah masyarakat untuk merawat penampilan, menjaga kesehatan kulit dan membantu mengatasi masalah kulit khususnya pada bagian wajah. Salah satu masalah kulit yang sering dihadapi oleh klinik-klinik kecantikan adalah jerawat. Jerawat atau dalam istilah medis disebut sebagai *Acne Vulgaris* merupakan salah satu penyakit kulit yang sering terjadi pada laki-laki maupun perempuan terutama di kalangan remaja.

Dikutip dari laman halodoc.com, dr.Fadhli Rizal Makarim menjelaskan bahwa jerawat biasanya muncul di bagian tubuh yang memiliki kelenjer minyak terbanyak seperti wajah, leher, punggung dan dada bagian atas. Kondisi tersebut terjadi karena produksi sebum yang berlebihan sehingga mengakibatkan pori-pori menjadi tersumbat dan terbentuklah jerawat atau *acne vulgaris* tersebut. Selain itu juga, terdapat beberapa faktor lainnya yang dapat menyebabkan munculnya jerawat yaitu perubahan hormon yang terjadi saat hamil dan sebelum menstruasi, penggunaan kosmetik yang tidak sesuai, kebersihan kulit yang tidak dijaga dan juga stress yang berlebihan [6]. Selain itu, dr.Airindya Bella juga menjelaskan pada laman alodokter.com bahwa jerawat dibagi menjadi 3 jenis berdasarkan tingkat keparahannya yaitu tingkatan ringan yang ditandai dengan munculnya komedo hitam atau putih dan tidak meradang, tingkatan sedang yang ditandai dengan munculnya benjolan lunak yang berwarna merah dan bernanah yang biasa disebut dengan istilah jerawat papula dan pustula, dan tingkatan parah yang biasa disebut dengan jerawat nodul dan jerawat kista [4]. Jerawat bukan hanya masalah kulit yang terlihat secara fisik, namun jerawat juga berdampak pada kesehatan mental seseorang. Para penderita jerawat biasanya merasa kurang percaya diri bahkan dapat mengakibatkan depresi akibat permasalahan jerawat yang mereka hadapi.

Untuk mengatasi masalah ini, dibutuhkan teknologi yang mampu mengidentifikasi dan mengklasifikasi jerawat berdasarkan tingkat keparahannya.

Penelitian mengenai klasifikasi tingkat keparahan jerawat sudah pernah dilakukan sebelumnya seperti penelitian yang dilakukan oleh Adhi Guna, dkk yang menggunakan metode *Convolutional Neural Network* (CNN) untuk mengidentifikasi tingkat keparahan jerawat. Tingkat akurasi sistem tersebut mencapai 75% dengan menggunakan 1.106 dataset [1]. Selain itu, penelitian yang dilakukan oleh Rianto dan Risdho menyimpulkan bahwa nilai akurasi pada model yang dikembangkan untuk mengklasifikasi tingkat keparahan jerawat menggunakan metode CNN adalah sebesar 0.9234 atau 92% pada *epoch* 100 [11]. Terdapat juga penelitian mengenai implementasi model YOLOv5 dalam mendeteksi jerawat yang dilakukan oleh Yulia, dkk yang menyatakan bahwa model YOLOv5 berhasil mendeteksi jerawat pada wajah secara real-time dengan tingkat keakuratan yang memadai [13].

Berdasarkan penelitian-penelitian yang sudah dilakukan sebelumnya, maka penulis tertarik untuk melakukan penelitian dengan metode yang berbeda. Metode yang digunakan penulis adalah metode *K-Nearest Neighbor* (KNN) dan *Random Forest* dengan bantuan model YOLOv8 untuk mendeteksi dan mengidentifikasi daerah wajah yang memiliki jerawat. Penelitian ini diharapkan dapat membantu para ahli di klinik kecantikan untuk mampu mengatasi masalah jerawat dengan memberikan penanganan yang tepat berdasarkan identifikasi tingkat keparahannya secara akurat.

1.2. Perumusan Masalah

Berdasarkan latar belakang yang telah disampaikan, perumusan masalah yang akan di bahas dalam proposal TA ini adalah sebagai berikut :

1. Bagaimana tahapan melakukan *preprocessing* data yang dilakukan untuk mengklasifikasi tingkat keparahan jerawat ?
2. Bagaimana cara mengimplementasikan algoritma *K-Nearest Neighbor* (KNN) dan *Random Forest* dalam mengklasifikasi tingkat keparahan jerawat?
3. Bagaimana performasi antara metode *K-Nearest Neighbor* (KNN) dan *Random Forest* dalam mengklasifikasi tingkat keparahan jerawat?

1.3. Tujuan

Berdasarkan latar belakang dan rumusan masalah yang telah dibahas sebelumnya, maka tujuan dari penulisan proposal TA ini adalah sebagai berikut :

1. Menjelaskan tahapan dalam melakukan *preprocessing* data yang dilakukan untuk mengklasifikasi tingkat keparahan jerawat.
2. Mengimplementasikan algoritma *K-Nearest Neighbor* (KNN) dan *Random Forest* dalam mengklasifikasi tingkat keparahan jerawat.
3. Menganalisis dan membandingkan performansi metode *K-Nearest Neighbor* (KNN) dan *Random Forest* dalam mengklasifikasi tingkat keparahan jerawat.

1.4. Rencana Kegiatan

Rencana kegiatan yang akan dilakukan untuk menyelesaikan penelitian ini adalah sebagai berikut :

1. Studi Literatur

Melakukan tinjauan literatur terhadap penelitian-penelitian sebelumnya yang berkaitan dengan klasifikasi tingkat keparahan jerawat menggunakan algoritma *K-Nearest Neighbor* dan *Random Forest*.

2. Pengumpulan Data

Mengumpulkan dataset dalam bentuk gambar jerawat dengan berbagai level tingkat keparahan yang akan digunakan untuk menganalisis tingkat keparahan jerawat menggunakan algoritma *K-Nearest Neighbor* dan *Random Forest*.

3. *Preprocessing* Data

Menyiapkan dataset dengan membersihkan dan merapikan data agar data siap digunakan dalam tahap deteksi menggunakan metode YOLOv8 dan implementasi model menggunakan algoritma *K-Nearest Neighbor* dan *Random Forest*.

4. Implementasi Algoritma

Mengimplementasikan algoritma atau model yang digunakan untuk mengklasifikasi tingkat keparahan jerawat berdasarkan data yang telah disiapkan sebelumnya.

5. Analisis Hasil Implementasi

Menganalisis hasil implementasi dengan mengevaluasi kinerja masing-masing model yang digunakan untuk mengklasifikasi tingkat keparahan jerawat menggunakan metrik evaluasi yang relevan.

6. Penulisan Laporan

Merangkum seluruh proses penelitian dan didokumentasikan dalam bentuk laporan tugas akhir yang terstruktur dan komprehensif sehingga dapat penelitian ini dapat digunakan sebagai landasan untuk penelitian-penelitian lanjutan.

1.5. Jadwal Kegiatan

Tabel 1. Jadwal Kegiatan

Kegiatan	Bulan					
	1	2	3	4	5	6
Studi Literatur						
Pengumpulan Data						
<i>Preprocessing</i> Data						
Implementasi Algoritma						
Analisis Hasil Implementasi						
Penulisan Laporan						

2. KAJIAN PUSTAKA

2.1. Studi Literatur

Menurut Synder, *literature review* adalah sebuah dasar yang berperan penting dalam semua jenis penelitian. *Literature review* berfungsi sebagai dasar dalam pengembangan pengetahuan, membuat pedoman dan dapat memunculkan ide-ide baru untuk penelitiaasn dan teori di masa depan [12].

Pada bagian ini, penulis akan menjelaskan *review* dari jurnal-jurnal yang sudah dibaca dan dijadikan sebagai referensi dalam penelitian ini. *Review* tersebut akan disajikan dalam bentuk tabel dan akan berisikan informasi seperti judul, penulis, tahun terbit dan *review* singkat dari jurnal-jurnal tersebut. Tabel studi literatur ini menjadi dasar penulis dalam menulis latar belakang dan pemilihan metode dalam penelitian ini. Berikut adalah tabel studi literatur untuk penelitian ini:

Tabel 2. Tabel Studi Literatur

No.	Referensi	Review
1.	Adhi Guna, E., Fransiska Sihombing, E., Nico Pasaribu, M., Syahputra, H., & Ramadhani, F. (2024). Implementasi Algoritma CNN Dalam Mengidentifikasi Tingkat Keparahan Jerawat Pada Wajah.	Penelitian ini membahas tentang implementasi metode CNN untuk mengidentifikasi secara akurat tingkat keparahan jerawat diwajah sehingga diharapkan mampu untuk memberikan solusi dalam mengatasi masalah keparahan jerawat dan mengurangi kesalahan perawatan. Penelitian ini menggunakan 1.106 dataset dan dibagi menjadi 3 tingkatan level keparahan yaitu level 1, level 2 dan level 3. Selanjutnya CNN diimplementasikan menggunakan <i>framework</i> Keras dari TensorFlow untuk melakukan pemodelan klasifikasi dan model tersebut dilatih untuk meminimalkan fungsi

		kerugian selama fase pelatihan. Setelah pelatihan model, model dievaluasi dan mendapatkan tingkat akurasi yang cukup baik yaitu sebesar 75%. [1]
2.	Rianto, R., & Risdho Listianto, D. (2023). Convolutional Neural Network untuk mengklasifikasi tingkat keparahan jerawat.	Penelitian ini bertujuan untuk mengembangkan model klasifikasi jenis dan tingkat keparahan jerawat menggunakan metode <i>Convolutional Neural Network</i> (CNN). Penelitian ini menggunakan 1500 gambar yang terbagi dalam 3 level yaitu level 0, 1, dan 2. Pada tahap <i>preprocessing</i> data, dilakukan standarisasi pada data seperti ukuran, kualitas, augmentasi dan kontras gambar. Selanjutnya proses pelatihan model melibatkan penggunaan pengoptimal ADAM dan pengujian <i>Epoch</i> (50,80,100) dan memiliki nilai akurasi cukup tinggi yaitu sebesar 92% pada <i>Epoch</i> 100. [11]
3.	Yulia, N., Cahyaningtyas, R., & Djamain, Y. (2024). <i>Implementasi You Only Look Once Version 5 (Yolov5) Untuk Deteksi Jenis Jerawat Pada Wajah.</i>	Penelitian ini membahas implementasi YOLOv5 dalam mendeteksi jerawat dan diharapkan mampu untuk membantu para ahli dalam mendiagnosis jerawat lebih akurat. Penelitian ini menggunakan 1.707 gambar yang menggambarkan 10 variasi jenis jerawat. Penelitian ini juga mengadopsi metodologi CRISP-DM yang terdiri dari 6 tahap yaitu, <i>Problem Understanding, Data Understanding, Data Preparation, Modelling, Evaluation</i> dan <i>Deployment</i> . Evaluasi

		model menggunakan <i>confusion matrix</i> dan memiliki hasil <i>Accuracy</i> 99%, <i>Precision</i> 99%, <i>Recall</i> 100% dan <i>F1-Score</i> 99.5%. Dari hasil implementasi, dapat disimpulkan bahwa model YOLOv5 berhasil mendeteksi jenis jerawat pada wajah dengan baik dan secara <i>real-time</i> namun terdapat beberapa kelas jerawat yang memiliki nilai <i>Accuracy</i> , <i>Precision</i> , <i>Recall</i> dan <i>F1-Score</i> 0%. [13]
4.	Andrian, Steele, Salim, E. S., Bindan, H., Pranoto, E., & Dharma, A. (2020). Analisa Metode Random Forest Tree dan K-Nearest Neighbor dalam Mendeteksi Kanker Serviks.	Penelitian ini membahas tentang perbandingan kedua algoritma yaitu <i>Random Forest Tree</i> dan <i>K-Nearest Neighbor</i> dalam mendeteksi kanker serviks. Dataset terdiri dari 854 data total dan 214 digunakan untuk menguji model. Kumpulan data riwayat pasien dianalisis menggunakan 4 teknik skrining yaitu <i>Hinselmann</i> , <i>Schiller</i> , <i>Sitologi</i> dan juga <i>Biospi</i> . Dari hasil analisis, disimpulkan bahwa kedua model yang diujikan mendapat tingkat akurasi yang cukup tinggi. Nilai akurasi <i>Random Forest</i> sebesar 88.7% dan nilai akurasi KNN 90.6% sehingga keduanya terbukti mampu untuk menangani kasus <i>multilabel classification</i> . [2]
5.	Rahman, B., Fauzi, F., & Amri, S. (2023). Perbandingan Hasil Klasifikasi Data Iris	Penelitian ini membahas tentang perbandingan hasil klasifikasi data Iris menggunakan algoritma <i>K-Nearest Neighbor</i> (KNN) dan <i>Random Forest</i> .

	menggunakan Algoritma K-Nearest Neighbor dan Random Forest.	Data terbagi menjadi 2 yaitu data <i>training</i> sebesar 80% dan data <i>testing</i> sebesar 20%. Performasi kedua metode tersebut akan dilihat dari nilai <i>F1-Score</i> dan nilai akurasi. Pada hasil implementasi menunjukkan nilai <i>F1-Score</i> dari metode KNN adalah 0.98 dan tingkat akurasi 97%. Sedangkan <i>F1-Score</i> metode <i>Random Forest</i> adalah sebesar 1.00 dan tingkat akurasi 100% sehingga hasil klasifikasi menggunakan kedua metode tersebut disimpulkan bahwa metode <i>Random Forest</i> lebih baik dalam memprediksi data Iris tersebut. [9]
--	---	--

2.2. K-Nearest Neighbor

K-Nearest Neighbor (KNN) adalah algoritma *machine learning* yang mudah untuk diterapkan dalam menyelesaikan masalah klasifikasi dan regresi. Algoritma KNN juga digolongkan sebagai *supervised learning* karena hasil *query instance* yang baru, diklasifikasi berdasarkan mayoritas dari kategori k terdekat sehingga kelas yang paling banyak muncul nantinya akan menjadi kelas hasil dari klasifikasi tersebut [3]. Konsep dari algoritma *K-Nearest Neighbor* (KNN) yaitu melakukan klasifikasi berdasarkan jarak terdekat suatu data dengan data yang lain. Kedekatan jarak tersebut biasanya dihitung menggunakan perhitungan jarak seperti *Euclidean Distance* atau *jarak Euclidean* [3]. *Jarak Euclidean* berfungsi untuk menguji ukuran yang bisa digunakan untuk menginterpretasi jarak kedua objek tersebut [3]. Berikut adalah rumus perhitungan dari *jarak Euclidean* [8] :

$$d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \quad (1)$$

Keterangan :

d = Jarak

x_1 = Titik Koordinat Latitude 1

x_2 = Titik Koordinat Latitude 2

y_1 = Titik Koordinat Longitude 1

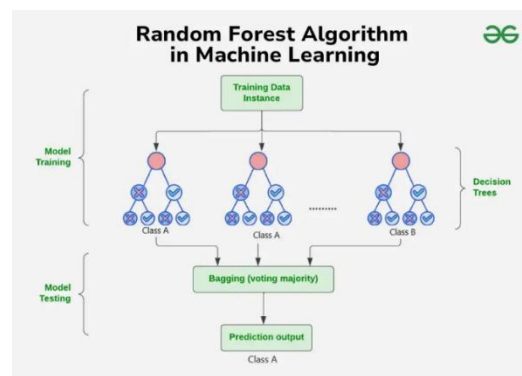
y_2 = Titik Koordinat Longitude 2

Berikut adalah langkah-langkah atau alur untuk menghitung metode *K-Nearest Neighbor* [3] yaitu :

1. Tentukan parameter K (jumlah nilai tetangga paling terdekat).
2. Hitung jarak dari jumlah tetangga K menggunakan *jarak Euclidean*
3. Mengurutkan objek-objek kedalam kelompok yang memiliki jarak *Euclidean* terkecil.
4. Mengumpulkan kategori Y (Klasifikasi Nearest Neighbor)
5. Dengan klasifikasi nearest neighbor paling banyak, maka dapat diprediksi *query instance* yang telah dihitung.

2.3. Random Forest

Random Forest adalah salah satu metode dalam *machine learning* yang juga diterapkan untuk menyelesaikan masalah klasifikasi dan regresi dalam ukuran yang besar. Random Forest termasuk dalam jenis *Ansamble Learning* menggunakan metode *bagging* atau dikenal sebagai *Bootstrap Aggregation* yang merupakan proses menggabungkan beberapa pengklasifikasi untuk menyelesaikan masalah yang kompleks dan meningkatkan kinerja model dan mengurangi *overfitting* [7] [10]. Konsep dari algoritma *Random Forest* yaitu mengkombinasikan masing-masing *tree* dari *decision tree* yang digabungkan hingga menjadi satu model. [5].



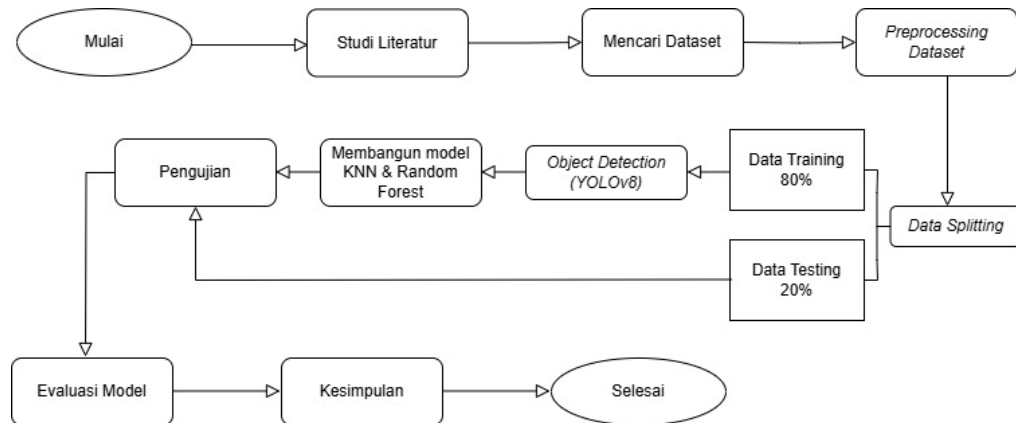
Gambar 1. Random Forest (sumber : [Random-Forest-Algorithm.webp \(1042×745\)](https://www.researchgate.net/publication/310427455))

Gambar.1 diatas merupakan gambaran cara kerja dari *random forest*. Pada *model training*, model menerima dataset dan membangun beberapa *decion tree* menggunakan subset *training data* yang dipilih secara acak. Setiap *decision tree* memilih subset fitur yang digunakan untuk membangun *tree* tersebut secara acak. Selanjutnya pada *model testing*, model menerima data baru yang ingin diprediksi. Setiap *decision tree* melakukan prediksi untuk data baru dan prediksi dari semua *decision tree* digabung menggunakan *bagging*. Selanjutnya prediksi yang paling sering muncul dari semua *decision tree* yang akan dipilih menjadi prediksi terakhir.

3. PERANCANGAN SISTEM

3.1. Perencanaan Sistem

Perencanaan sistem dalam penelitian klasifikasi tingkat keparahan jerawat menggunakan metode *K-Nearest Neighbor* (KNN) dan *Random Forest* akan ditunjukkan dalam *flowchart* berikut :



Gambar 2. Flowchart

Penelitian ini diawali dengan studi literatur untuk mencari sumber referensi dari penelitian-penelitian sebelumnya mengenai klasifikasi tingkat keparahan jerawat, metode KNN dan *Random Forest* dan juga metode YOLOv8 sebagai pendeteksi objek jerawat pada gambar.

3.2. Dataset

Dataset jerawat yang digunakan dalam penelitian ini bersumber dari website kaggle. Dataset tersebut bisa di akses melalui tautan berikut : [Acne dataset](#). Dataset terdiri dari 1833 gambar jerawat dengan berbagai tingkat keparahan dan memiliki ukuran dan rotasi gambar yang berbeda-beda. Berikut adalah contoh gambar jerawat yang terdapat dalam dataset tersebut



Gambar 3. Contoh Dataset

3.3. Preprocessing Dataset

Setelah proses pencarian dan pengumpulan dataset selesai, tahap selanjutnya yaitu melakukan *preprocessing* dataset. Tahap *preprocessing* dataset dilakukan dalam beberapa langkah yaitu pembersihan data, pelabelan data, mengubah ukuran gambar menjadi 420×420 piksel, melakukan augmentasi dan juga normalisasi pada dataset sehingga dapat meningkatkan kualitas dataset yang akan digunakan dalam penelitian ini.

Proses pembersihan data dilakukan dengan menghapus beberapa gambar karena memiliki resolusi yang kurang baik dan terdapat beberapa gambar yang sama sehingga jumlah gambar yang tersisa dalam dataset ini adalah 1265 gambar. Langkah selanjutnya yaitu dataset diberi label pada gambar jerawat menggunakan aplikasi *roboflow*. Pelabelan dibagi menjadi 3 level tingkat keparahan yaitu tingkat ringan yang ditandai dengan ‘acne 1’, tingkat keparahan sedang yang ditandai dengan ‘acne 2’ dan tingkat keparahan yang tinggi ditandai dengan ‘acne 3’. Dengan melakukan pelabelan yang tepat, model dapat dilatih untuk mengkasifikasi dan mengenali jerawat berdasarkan tingkat keparahannya.



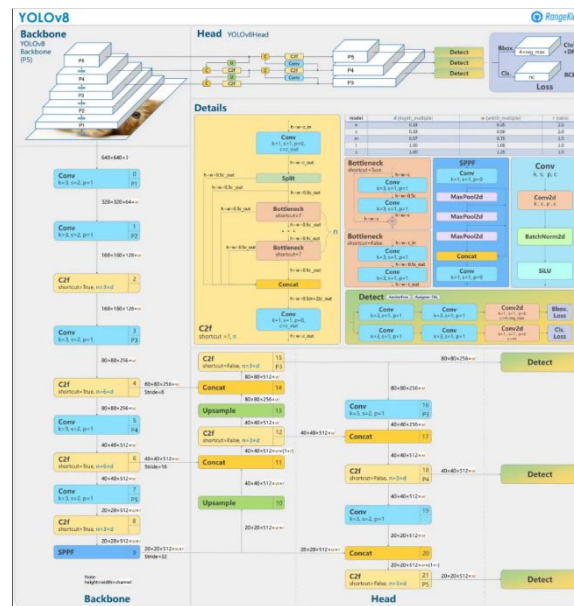
Gambar 4. Pelabelan Dataset

Gambar 4 merupakan contoh pelabelan pada dataset dengan berbagai level tingkat keparahannya. Jerawat dengan kategori ‘acne 1’ ditandai dengan *bounding box* berwarna merah, kemudian kategori ‘acne 2’ ditandai dengan *bounding box* berwarna hijau dan kategori ‘acne 3’ ditandai dengan *bounding box* berwarna ungu. Setelah memberi label, dataset diekspor kedalam format YOLOv8 agar dapat digunakan untuk melatih metode YOLOv8 dalam mendeteksi objek jerawat pada gambar.

3.4. Data Splitting

Setelah melakukan *preprocessing*, data akan dibagi menjadi dua bagian yaitu *data training* sebesar 80% dan juga *data testing* sebesar 20% dalam tahap *data splitting*. *Data training* berguna untuk melatih model *K-Nearest Neighbor* dan *Random Forest* sehingga mampu mengenali dan mempelajari pola dan karakteristik objek pada dataset yang akan dideteksi menggunakan metode YOLOv8. *Data testing* digunakan untuk menguji model yang sudah dilatih. *Data testing* merupakan data yang belum pernah dilihat sebelumnya oleh model sehingga dapat mengukur kemampuan model dalam menggeneralisasi data. Proses *data splitting* berguna untuk mengurangi *overfitting* dan juga dapat mengevaluasi kinerja model secara akurat.

3.5. Object Detection (YOLOv8)



Gambar 5. Arsitektur YOLOv8, visualization made by GitHub user RangeKin
(sumber : [image-1799.webp \(1000x1047\)](https://img1799.webp (1000x1047)))

Model YOLOv8 yang digunakan dalam penelitian ini bertujuan untuk mendeteksi objek jerawat dalam gambar. Dataset yang sebelumnya sudah diberi label pada *roboflow* akan diproses melalui 3 komponen utama dalam YOLOv8 yaitu *backbone*, *neck* dan *head*. Proses pertama terjadi di *backbone*, gambar diinput kemudian dilakukan proses ekstraksi fitur pada layer Conv dari P1-P5 yang akan mendeteksi fitur dari yang sederhana hingga yang kompleks dan juga C2f yang

bertujuan untuk efisiensi komputasi. Proses selanjutnya terjadi di *neck* yang merupakan jembatan antara *backbone* dan juga *head* dan terdiri dari beberapa blok yaitu blok *Spatial Pyramid Pooling – Fast* (SPPF), *Layer Concat*, *Layer Upsample* dan juga *Skip-Connection*. Fitur yang telah diekstrak pada *backbone* akan dioptimalkan menggunakan SPPF untuk *pooling* dengan ukuran karnel yang berbeda atau dalam berbagai skala. Selanjutnya pada *layer concat*, fitur akan digabungkan dari berbagai level *backbone* kemudian *layer upsample* akan membantu meningkatkan resolusi fitur dan *skip-connection* untuk preservasi detail fitur yang penting. Proses yang terakhir pada model YOLOv8 ini terjadi di *head*. *Layer Detection* akan menghasilkan *bounding box* dan *class prediction* sesuai label yang sebelumnya sudah dibuat di *roboflow*. Hasil dari model YOLOv8 akan digunakan sebagai input untuk mengimplementasikan model KNN (*K-Nearest Neighbor*) dan *Random Forest* pada tahap selanjutnya.

3.6. Pembangunan Model KNN dan Random Forest

Dalam penelitian ini, model yang dibangun untuk mengklasifikasi tingkat keparahan jerawat adalah *K-Nearest Neighbor* dan *Random Forest*. Kedua metode tersebut merupakan metode dalam *machine learning* yang diterapkan untuk menyelesaikan masalah klasifikasi dan regresi. KNN merupakan algoritma yang konsep dasarnya adalah mencari ‘K’ tetangga terdekat dari titik data yang ingin diprediksi dan diklasifikasi menurut mayoritas kelas dari tetangga tersebut. Sedangkan *random forest* merupakan algoritma yang konsep dasarnya adalah menggabungkan beberapa *decision tree* yang berguna untuk mengurangi overfitting dan dapat meningkatkan akurasi. Selanjutnya model akan dilatih menggunakan *data training* yang telah disiapkan sebelumnya agar dapat mempelajari pola dan karakteristik objek dengan baik sehingga model mampu untuk mengklasifikasi jerawat berdasarkan tingkat keparahannya.

3.7. Evaluasi Model

Evaluasi model akan dilakukan dengan menggunakan *confusion matrix* untuk menganalisis hasil pengujian model serta melihat akurasi dan kinerja dari model yang sudah dibangun. Elemen utama dari *confusion matrix* adalah *True Positive* (TP), *True Negative* (TN), *False Positive* (FP) dan juga *False Negative*

(FN). Selanjutnya *confusion matrix* digunakan untuk menghitung nilai *Accuracy*, *Precision*, *Recall* dan *F1-Score* dengan menggunakan persamaan sebagai berikut :

- *Accuracy*

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \times 100\% \quad (2)$$

- *Precision*

$$Precision = \frac{TP}{TP + FP} \times 100\% \quad (3)$$

- *Recall*

$$Recall = \frac{TP}{TP + FN} \times 100\% \quad (4)$$

- *F1-Score*

$$F1\ Score = 2 \times \frac{Recall \times Precision}{Recall + Precision} \times 100\% \quad (5)$$

Tingkat akurasi yang diperoleh dari masing-masing nilai *Accuracy*, *Precision*, *Recall* dan *F1-Score* antar kedua model tersebut digunakan sebagai pembandingan untuk mencari model yang terbaik dalam melakukan klasifikasi tingkat keparahan jerawat sehingga hal tersebut dapat menjadi kesimpulan untuk mengakhiri penelitian ini dan memberi rekomendasi untuk pengembangan sistem pada penelitian selanjutnya.

DAFTAR PUSTAKA

- [1] Adhi Guna E, Fransiska Sihombing E, Nico Pasaribu M, Syahputra H, Ramadhani F. Implementasi Algoritma Cnn Dalam Mengidentifikasi Tingkat Keparahan Jerawat Pada Wajah. JATI (Jurnal Mhs Tek Inform. 2024;8(4):5778–82.
- [2] Andrian, Steele, Salim ES, Bindan H, Pranoto E, Dharma A. Analisa Metode Random Forest Tree dan K-Nearest Neighbor dalam Mendeteksi Kanker Serviks. J Ilmu Komput dan Sist Inf [Internet]. 2020;3(2):97–101. Available from: <https://ejournal.sisfokomtek.org/index.php/jikom/article/view/73>
- [3] Avelita B. A. Klasifikasi K-Nearest Neighbor. Usd [Internet]. 2014 Nov 4; Available from: https://www.academia.edu/9131959/A. Klasifikasi_KNearest_Neighbor
- [4] Bella A. Acne Vulgaris, Kenali Penyebab dan Langkah Penanganannya [Internet]. Alodokter. 2023. Available from: <https://www.alodokter.com/acne-vulgaris-jenis-penyebab-dan-penanganan>
- [5] Bunga. Cara kerja Algoritma Random Forest [Internet]. Algoritma. 2022. Available from: <https://algorit.ma/blog/cara-kerja-algoritma-random-forest-2022/>
- [6] Fadli R. Jerawat [Internet]. Halodoc. 2024. Available from: <https://www.halodoc.com/kesehatan/jerawat?srsId=AfmBOOpUktOKhMLODTa rztziQlYH9xQugPmgOJ-5ZdRx00KZXXKL-t3v>
- [7] Machine Learning Random Forest Algorithm - JavatPoint [Internet]. www.javatpoint.com. Available from: <https://www.javatpoint.com/machine-learning-random-forest-algorithm>
- [8] Miftahuddin Y, Umaroh S, Karim FR. Perbandingan Metode Perhitungan Jarak Euclidean, Haversine, Dan Manhattan Dalam Penentuan Posisi Karyawan. J Tekno Insentif. 2020;14(2):69–77.
- [9] Rahman B, Fauzi F, Amri S. Perbandingan Hasil Klasifikasi Data Iris menggunakan Algoritma K-Nearest Neighbor dan Random Forest. J Data Insights. 2023;1(1):19–26.

- [10] Random Forest [Internet]. IBM. 2024. Available from: <https://www.ibm.com/topics/random-forest#:~:text=Random%20forest%20is%20a%20commonly,Decision%20trees>
- [11] Rianto R, Risdho Listianto D. Convolutional Neural Network untuk mengklasifikasi tingkat keparahan jerawat. Aiti. 2023;20(2):167–76.
- [12] Snyder H. Literature review as a research methodology: An overview and guidelines. J Bus Res [Internet]. 2019;104(March):333–9. Available from: <https://doi.org/10.1016/j.jbusres.2019.07.039>
- [13] Yulia N, Cahyaningtyas R, Djamain Y. Implementasi You Only Look Once Version 5 (Yolov5) Untuk Deteksi Jenis Jerawat Pada Wajah. 2024;17(1):34–43.