



# **Ekstraksi Ciri untuk Klasifikasi Tipe Kulit Wajah Menggunakan Citra Foto**

Shinta Dewi Kusumaningrum  
20917055

*Tesis diajukan sebagai syarat untuk meraih gelar Magister Komputer  
Konsentrasi Informatika Medis  
Program Studi Informatika Program Magister  
Fakultas Teknologi Industri  
Universitas Islam Indonesia  
2024*

**Lembar Pengesahan Pembimbing**

**Ekstraksi Ciri untuk Klasifikasi Tipe Kulit Wajah Menggunakan Citra Foto**

Shinta Dewi Kusumaningrum

20917055



Yogyakarta, Agustus 2024

Pembimbing

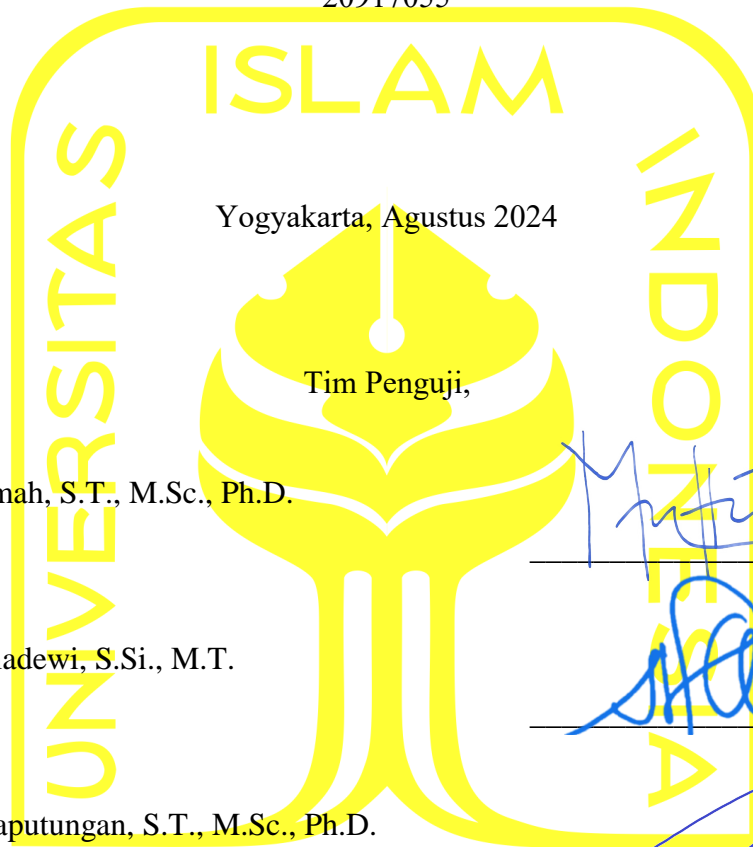
Izzati Muhimmah, S.T., M.Sc., Ph.D.

# Lembar Pengesahan Penguji

## Ekstraksi Ciri untuk Klasifikasi Tipe Kulit Wajah Menggunakan Citra Foto

Shinta Dewi Kusumaningrum

20917055



Yogyakarta, Agustus 2024

Tim Penguji,

Izzati Muhimmah, S.T., M.Sc., Ph.D.

Ketua

Dr. Sri Kusumadewi, S.Si., M.T.

Anggota I

Irving Vitra Paputungan, S.T., M.Sc., Ph.D.

Anggota II

08/10/2024



Mengetahui,

Ketua Program Studi Informatika Program Magister

Universitas Islam Indonesia



Irving Vitra Paputungan, S.T., M.Sc., Ph.D.

## Abstrak

### Ekstraksi Ciri untuk Klasifikasi Tipe Kulit Wajah Menggunakan Citra Foto

Kulit merupakan salah satu bagian terluar yang melapisi wajah dan menjadi bagian kulit paling sensitif dibanding lainnya. Berbagai permasalahan kulit seperti kusam, berjerawat, hingga komedo merupakan akibat dari adanya kesalahan dalam melakukan perawatan kulit wajah atau pada saat seseorang telah memasuki masa pubertas yang ditandai dengan mimpi basah bagi laki-laki dan *menarche* bagi perempuan. Langkah awal dalam perawatan kulit wajah adalah dengan mengetahui tipe kulit wajah yang dimiliki. Hal ini sebagai upaya untuk memilih produk perawatan kulit yang sesuai, mengatasi permasalahan yang dialami, hingga meminimalkan kesalahan dalam memilih melakukan perawatan kulit. Berkaitan dengan hal tersebut, pada penelitian ini dilakukan pengembangan model untuk klasifikasi tipe kulit wajah dengan menggunakan citra foto. Pemodelan menggunakan metode *Convolutional Neural Network* (CNN) dengan melakukan eksplorasi pada setiap arsitekturnya yaitu MobileNetV1, MobileNetV2, dan MobileNetV3Large. Data citra yang berhasil dikumpulkan sebanyak 588 citra yang diambil secara langsung dan telah dilakukan *labelling* oleh dokter. Pemodelan dilakukan dengan menggunakan data citra yang telah dibagi sebesar 80% untuk *data training* dan 20% *data testing*. Proses pemodelan dimulai dengan *preprocessing*, dilanjutkan *training*, *testing*, hingga *model evaluation*. Hasil pemodelan untuk setiap arsitektur menunjukkan bahwa arsitektur yang cukup baik untuk melakukan klasifikasi tipe kulit wajah adalah MobileNetV3Large dengan *validation accuracy* sebesar 45% dan *training accuracy* sebesar 49%. Setelah dilakukan perbaikan citra dan pemodelan ulang dengan arsitektur MobileNetV3Large, didapatkan hasil nilai *validation accuracy* sebesar 48% dan nilai *training accuracy* sebesar 51%, hasil tersebut menunjukkan adanya peningkatan setelah data citra diperbaiki.

### Kata kunci

cnn, mobilenet, klasifikasi, citra, kulit, wajah

## **Abstract**

### **Feature Extraction for Facial Skin Type Classification Using Photo Images**

The skin is one of the outermost layers that covers the face and is the most sensitive part of the skin compared to others. Various skin problems such as dullness, acne, and blackheads are the result of errors in facial skin care or when someone enters puberty, marked by wet dreams for males and menarche for females. The initial step in facial skin care is to determine the type of facial skin one has. This is an effort to choose suitable skin care products, address problems experienced, and minimize mistakes in choosing skin care. In this study, a model for classifying facial skin types using photo images was developed. Modelling used the Convolutional Neural Network (CNN) method by exploring each of its architectures, namely MobileNetV1, MobileNetV2, and MobileNetV3Large. The image data successfully collected amounted to 588 images taken directly and labelled by a doctor. The modeling results for each architecture indicate that the most suitable architecture for classifying facial skin types is MobileNetV3Large, with a validation accuracy of 45% and a training accuracy of 49%. After image enhancement and re-modelling with the MobileNetV3Large architecture, the results showed a validation accuracy of 48% and a training accuracy of 51%. These results indicate an improvement after image data enhancement.

#### **Keywords**

cnn, mobilenet, classification, image, skin, face

## **Pernyataan Keaslian Tulisan**

Dengan ini saya menyatakan bahwa tesis ini merupakan tulisan asli dari penulis, dan tidak berisi material yang telah diterbitkan sebelumnya atau tulisan dari penulis lain terkecuali referensi atas material tersebut telah disebutkan dalam tesis. Apabila ada kontribusi dari penulis lain dalam tesis ini, maka penulis lain tersebut secara eksplisit telah disebutkan dalam tesis ini.

Dengan ini saya juga menyatakan bahwa segala kontribusi dari pihak lain terhadap tesis ini, termasuk bantuan analisis statistik, desain survei, analisis data, prosedur teknis yang bersifat signifikan, dan segala bentuk aktivitas penelitian yang dipergunakan atau dilaporkan dalam tesis ini telah secara eksplisit disebutkan dalam tesis ini.

Segala bentuk hak cipta yang terdapat dalam material dokumen tesis ini berada dalam kepemilikan pemilik hak cipta masing-masing. Apabila dibutuhkan, penulis juga telah mendapatkan izin dari pemilik hak cipta untuk menggunakan ulang materialnya dalam tesis ini.

Yogyakarta, 9 Oktober 2024



Handwritten signature of Shinta Dewi Kusumaningrum in black ink.

Shinta Dewi Kusumaningrum, S.Kom.

## Daftar Publikasi

### Publikasi yang menjadi bagian dari tesis

Analisis Faktor dan Metode untuk Menentukan Tipe Kulit Wajah: Tinjauan Literatur  
(Kusumaningrum et al., 2023).

Kontributor	Jenis Kontribusi
Izzati Muhimmah	Mendesain eksperimen Menulis dan merevisi <i>paper</i>
Shinta Dewi Kusumaningrum	Mendesain eksperimen Menulis dan memperbaiki <i>paper</i>

## **Halaman Kontribusi**

Dalam penulisan tesis ini, dosen pembimbing memberikan arah penelitian yang jelas, memfasilitasi pencarian data, serta memberikan saran terkait pengolahan data. Selain itu, pembimbing juga memberikan masukan untuk perbaikan baik dalam hal pemodelan, analisis, hingga penulisan. Dalam proses penelitian dan pengerjaan laporan, dosen pembimbing memberikan banyak dukungan dan masukan sehingga penulis mampu menghadapi kesulitan yang dialami.

## **Halaman Persembahan**

Tesis ini penulis persembahkan untuk orang-orang yang berkontribusi dalam menyelesaikan studi antara lain untuk:

1. Mami tercinta yang tidak pernah lelah mendoakan dan memberikan motivasi yang luar biasa.
2. Kakak-kakak tersayang yang selalu memberikan semangat untuk menjalani studi hingga menyelesaikan tesis.
3. Dio Agus Nofrizal yang selalu memberikan dukungan dan meyakinkan bahwa penulis mampu untuk menyelesaikan studi dengan baik.
4. Ibu Izzati Muhimmah sebagai dosen pembimbing yang selalu memberikan solusi atas kendala yang dialami penulis, memberikan arahan untuk menyelesaikan tesis ini.
5. Almamater Universitas Islam Indonesia, yang telah memberikan kesempatan bagi penulis untuk bisa menuntut ilmu dan mendapatkan wawasan yang berharga.

## **Kata Pengantar**

*Bismillahirrahmaanirrahiim,*

*Assalaamu'alaikum warahmatullaahi wabarokaatuh*

*Alhamdulillah Rabbil 'Alamin*, segala puji bagi Allah, Tuhan semesta alam. Atas berkat rahmat Allah SWT, penulis mampu menyelesaikan tesis ini.

Keberhasilan dalam menyelesaikan tesis ini tidak lepas dari adanya bimbingan dan dorongan dari dosen pembimbing. Untuk itu, penulis menyampaikan terima kasih terbesar kepada Ibu Izzati Muhimmah yang selalu sabar dalam membimbing penulis. Terima kasih juga dipersembahkan oleh penulis kepada seluruh dosen Magister Informatika UII yang telah memberikan wawasan dan ilmu yang sangat bermanfaat bagi penulis. Tidak lupa, penulis juga menyampaikan terima kasih untuk dosen penguji yang memberikan kritik dan saran perbaikan untuk tesis ini.

Penulis menyadari bahwa masih banyak kekurangan dari tesis ini dan jauh dari kata sempurna. Namun, besar harapan penulis agar tesis ini dapat memberikan manfaat bagi orang lain yang membacanya.

*Wassalaamu'alaikum warahmatullaahi wabarokaatuh*

## Daftar Isi

Lembar Pengesahan Pembimbing .....	i
Lembar Pengesahan Penguji.....	ii
Abstrak.....	iii
Abstract.....	iv
Pernyataan Keaslian Tulisan .....	v
Daftar Publikasi .....	vi
Halaman Kontribusi.....	vii
Halaman Persembahan .....	viii
Kata Pengantar.....	ix
Daftar Isi.....	x
Daftar Tabel.....	xiii
Daftar Gambar .....	xiv
Glosarium .....	xvi
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	2
1.3 Tujuan Penelitian .....	3
1.4 Batasan Penelitian.....	3
1.5 Sistematika Penulisan .....	3
2.1 Kerangka Teori .....	5
2.1.1 Kulit Normal.....	7
2.1.2 Kulit Berminyak .....	7
2.1.3 Kulit Kering.....	7
2.1.4 Kulit Kombinasi .....	7
2.1.5 Kulit Sensitif.....	7
2.2 Penelitian Terdahulu .....	8

2.3	Konsep Pengetahuan yang Digunakan.....	19
2.3.1	<i>Convolutional Neural Network (CNN)</i> .....	19
2.3.2	MobileNet.....	19
2.3.3	Kualitas dan Kuantitas Data Citra .....	20
3.1	Tahapan Penelitian.....	22
3.2	Analisis Masalah.....	22
3.2.1	Studi Literatur.....	22
3.2.2	Analisis Kebutuhan.....	23
3.3	<i>Preprocessing Data</i> .....	24
3.3.1	Pengumpulan Data.....	24
3.3.2	Labelling.....	27
3.3.3	<i>Resize</i> atau <i>Rescaling</i> Citra.....	28
3.3.4	Pembagian Data Citra.....	28
3.3.5	Augmentasi Data .....	29
3.4	Training Model .....	29
3.5	<i>Model Evaluation</i> .....	31
4.1	<i>Dataset</i> .....	33
4.2	<i>Preprocessing</i> .....	34
4.3	Membangun Model.....	36
4.4	<i>Training Model</i> .....	38
4.5	<i>Model Evaluation</i> .....	41
4.5.1	<i>Plotting</i> Akurasi dan <i>Loss</i> .....	41
4.5.2	Perbandingan <i>Accuracy</i> dan <i>Loss</i> .....	44
4.5.3	<i>Confusion Matrix</i> .....	44
4.5.4	<i>Classification Report</i> .....	47
4.5.5	Hasil Evaluasi .....	49
4.6	Plot Pencahayaan .....	49

4.7	Hubungan Antara Kelembaban dan Tipe Kulit Wajah.....	53
4.8	Pemodelan Setelah Perbaikan Citra.....	54
4.8.1	Perbaikan Citra.....	54
4.8.2	<i>Dataset</i> .....	56
4.8.3	<i>Preprocessing</i> .....	56
4.8.4	<i>Training Model</i> .....	57
4.8.5	Hasil.....	58
5.1	Kesimpulan.....	64
5.2	Saran.....	65

## Daftar Tabel

Tabel 2.1 Klasifikasi Tipe Kulit Wajah Berdasarkan Ciri.....	5
Tabel 2.2 Klasifikasi Tipe Kulit Wajah dengan Citra Mikroskopis .....	13
Tabel 2.3 Klasifikasi Tipe Kulit Wajah dengan Citra Foto .....	15
Tabel 2.4 Klasifikasi Tipe Kulit Wajah Berdasarkan Kadar Minyak dan Penyakit.....	17
Tabel 3.1 Perincian Data Citra <i>Full</i> Wajah .....	26
Tabel 3.2 Perincian Data Citra <i>Full</i> Wajah dan <i>Patch</i> .....	26
Tabel 3.3 Perincian <i>Data Training</i> dan <i>Data Testing</i> .....	29
Tabel 4.1 Perbandingan Validasi <i>Accuracy</i> dan <i>Loss</i> .....	44
Tabel 4.2 Perbandingan <i>Precision</i> , <i>Recall</i> , dan <i>F1-Score</i> .....	48
Tabel 4.3 Pembagian Waktu Plot Pencahayaan .....	50
Tabel 4.4 Hasil Plot Pencahayaan .....	52
Tabel 4.5 Rerata Kelembaban Kulit .....	53
Tabel 4.6 Hasil <i>Confusion Matrix</i> .....	59
Tabel 4.7 Hasil TP, TN, FP, FN, dan IP .....	61
Tabel 4.8 Nilai <i>Precision</i> , <i>Recall</i> , dan <i>F1-Score</i> Data <i>Training</i> .....	62
Tabel 4.9 Nilai <i>Precision</i> , <i>Recall</i> , dan <i>F1-Score</i> Data <i>Testing</i> .....	62

## Daftar Gambar

Gambar 1.1 Pertumbuhan Industri <i>Beauty and Personal Care</i> .....	2
Gambar 3.1 Tahapan Penelitian.....	22
Gambar 3.2 Alur Pencarian <i>Dataset</i> .....	24
Gambar 3.3 Alur Pengumpulan Data Citra .....	27
Gambar 3.4 Contoh <i>Dataset</i> yang Dikumpulkan .....	27
Gambar 3.5 Struktur Folder Pengelompokan Data .....	28
Gambar 3.6 Alur <i>Convolutional Neural Network</i> (CNN).....	30
Gambar 3.7 Alur <i>Data Training</i> .....	30
Gambar 3.8 Alur <i>Data Training</i> .....	31
Gambar 4.1 <i>Source code</i> Pembagian Data .....	33
Gambar 4.2 <i>Source code</i> Augmentasi Data 1 .....	34
Gambar 4.3 <i>Source code</i> Augmentasi Data 2.....	35
Gambar 4.4 <i>Source code</i> Model .....	36
Gambar 4.5 Hasil Model .....	38
Gambar 4.6 <i>Training</i> Model MobileNetV1 .....	39
Gambar 4.7 <i>Training</i> Model MobileNetV2.....	39
Gambar 4.8 <i>Training</i> Model MobileNetV3Large .....	40
Gambar 4.9 <i>Plotting</i> Akurasi dan <i>Loss</i> .....	42
Gambar 4.10 Hasil Akurasi dan <i>Loss</i> MobileNetV1 .....	43
Gambar 4.11 Hasil Akurasi dan <i>Loss</i> MobileNetV2.....	43
Gambar 4.12 Hasil Akurasi dan <i>Loss</i> MobileNetV3Large.....	44
Gambar 4.13 <i>Source Code Confusion Matrix</i> .....	46
Gambar 4.14 <i>Confusion Matrix</i> MobileNetV1 .....	46
Gambar 4.15 <i>Confusion Matrix</i> MobileNetV2.....	47
Gambar 4.16 <i>Confusion Matrix</i> MobileNetV3Large .....	47
Gambar 4.17 <i>Source Code</i> Hasil Model.....	47
Gambar 4.18 <i>Source Code</i> Plot Pencahayaan 1 .....	50
Gambar 4.19 <i>Source Code</i> Plot Pencahayaan 2 .....	50
Gambar 4.20 <i>Source Code</i> Plot Pencahayaan 3 .....	51
Gambar 4.21 Hasil Plot Pencahayaan.....	51
Gambar 4.22 <i>Source Code</i> Standar Deviasi .....	52

Gambar 4.23 <i>Source Code</i> Perbaikan Citra 1 .....	54
Gambar 4.24 <i>Source Code</i> Perbaikan Citra 2.....	55
Gambar 4.25 <i>Source Code</i> Perbaikan Citra 3.....	56
Gambar 4.26 <i>Source Code</i> Augmentasi Data.....	57
Gambar 4.27 <i>Source Code</i> Training Model .....	57
Gambar 4.28 Hasil Akurasi dan <i>Loss</i> .....	58
Gambar 4.29 Hasil <i>Confusion Matrix</i> .....	59

## Glosarium

CNN	- Convolutional Neural Network
KNN	- K Nearest Neighbor
GLCM	- Gray Level Co-Occurrence Matrix
SVM	- Support Vector Machine
LDA	- Linear Discriminant Analysis
PCA	- Principal Component Analysis
DWT	- Discrete Wavelet Transform
HOG	- Histogram of Oriented Gradient
ROS	- Rosacea
AK	- Actinic Keratosis
SK	- Seborrhea Keratosis
LE	- Lupus Eritematosa
SCC	- Karsinoma Sel Skuamosa
BCC	- Karsinoma Sel Basal

# BAB 1

## Pendahuluan

### 1.1 Latar Belakang

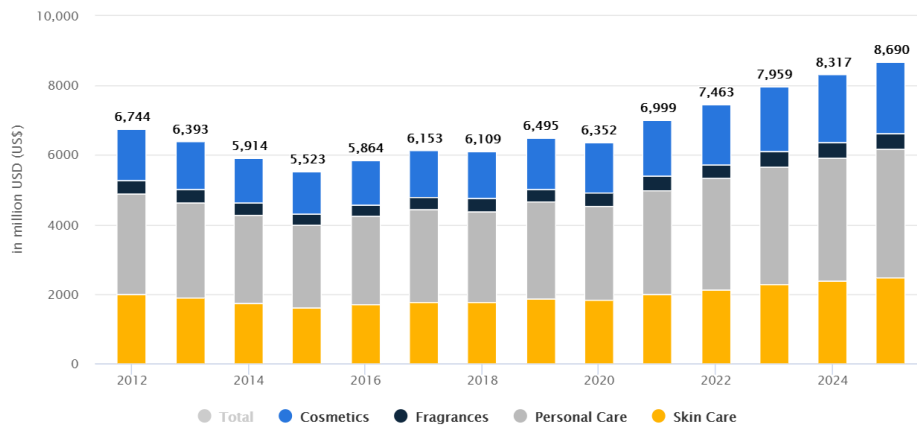
Kulit merupakan salah satu bagian terluar yang melapisi wajah dan menjadi bagian kulit paling sensitif dibanding lainnya (Susanti, 2014). Berbagai permasalahan kulit seperti kusam, berjerawat, hingga komedo merupakan akibat dari adanya kesalahan dalam melakukan perawatan kulit wajah atau pada saat seseorang telah memasuki masa pubertas yang ditandai dengan mimpi basah bagi laki-laki dan *menarche* bagi perempuan (Triyanto, 2010).

Survei yang dilakukan pada tahun 2018 oleh Mark Plus Inc dan Zap Beauty Clinic terhadap 17.889 wanita di Indonesia dengan usia 18-23 tahun menunjukkan bahwa sebesar 59,7% wanita mengalami permasalahan kulit wajah seperti berjerawat dan komedo, 57,7% mengalami permasalahan bekas jerawat dan flek hitam, 36,9% mengalami permasalahan kulit kusam, 36,8% mengalami permasalahan kulit pori-pori besar, serta sebesar 34,1% mengalami permasalahan kadar minyak berlebih (ZAP Beauty Clinic & Markplus Inc, 2018).

Setiap orang bisa mengalami permasalahan kulit wajah yang berbeda serta penyebab yang berbeda pula. Pengaruh produk perawatan kulit dan kosmetik terhadap kulit wajah memberikan reaksi yang berbeda bagi setiap orang tergantung tipe kulit yang dimilikinya. Survei terhadap  $\pm$  6000 responden pada tahun 2021 oleh Mark Plus Inc dan ZAP Beauty Clinic memberikan hasil bahwa sebanyak 86,2% Wanita di Indonesia memilih produk perawatan kulit dengan kandungan yang sesuai dengan tipe kulit dan manfaat yang diperlukan (ZAP Beauty Clinic & Markplus Inc, 2021).

Dalam 10-15 tahun ke depan, Indonesia diprediksi menjadi lima besar pasar kosmetik terbesar di dunia dengan prediksi pertumbuhan sebesar 7,2% per tahun (Swiss Business Hub Indonesia, 2018). Menurut data dari Euromonitor International yang berjudul *The Future of Skin Care*, saat ini Indonesia berada pada peringkat dua dunia sebagai penyumbang pertumbuhan perawatan kulit terbesar (EU-Indonesia Business Network, 2019). Industri *beauty and personal care* diperkirakan terus mengalami perkembangan sebesar 5,56% setiap tahunnya. Perkembangan pasar *beauty and personal care* di Indonesia yang ditunjukkan dalam Gambar 1.1 diperkirakan terus mengalami kenaikan hingga 2024

nanti. Segmen pasar terbesar dalam industri ini adalah *personal care*, disusul oleh *skincare*, dan *cosmetics*.



Gambar 1.1 Pertumbuhan Industri *Beauty and Personal Care*.

Sumber: (Statista, 2020)

Besarnya pertumbuhan *skincare* dan kosmetik di Indonesia tentu saja seiringan dengan banyaknya produk yang tersedia di pasaran. Banyaknya produk kecantikan yang dijual di pasaran tentu saja membuat masyarakat semakin bingung. Membeli produk yang sesuai tentu saja memberikan efek positif terhadap kulit, tetapi di sisi lain jika salah dalam membeli produk, dapat memperparah permasalahan kulit atau bahkan menambah permasalahan baru, sehingga dengan adanya sistem ini dapat membantu untuk menentukan tipe kulit sebagai langkah awal membeli produk.

Langkah awal dalam perawatan kulit wajah adalah dengan mengetahui tipe kulit wajah yang dimiliki. Hal ini sebagai upaya untuk memilih produk perawatan kulit yang sesuai, mengatasi permasalahan yang dialami, hingga meminimalkan kesalahan dalam memilih melakukan perawatan kulit. Berkaitan dengan hal tersebut, diperlukan adanya sistem untuk ekstraksi ciri untuk menentukan tipe kulit wajah. Untuk mempermudah penggunaan, citra yang digunakan merupakan citra foto digital yang dapat diambil menggunakan kamera digital maupun *handphone*. Hal ini akan membantu masyarakat untuk mengetahui tipe kulit wajah yang dimiliki dengan lebih mudah, hemat biaya, tenaga, dan waktu karena tidak perlu melakukan konsultasi secara langsung dengan dokter.

## 1.2 Rumusan Masalah

Berdasarkan permasalahan yang telah dijelaskan sebelumnya, rumusan masalah yang akan diselesaikan dalam penelitian ini antara lain.

1. Bagaimana proses untuk mengklasifikasikan tipe kulit wajah menggunakan citra foto?
2. Bagaimana hasil perbandingan hasil ekstraksi ciri untuk klasifikasi setiap tipe kulit wajah?

### **1.3 Tujuan Penelitian**

Tujuan yang akan dicapai dalam penelitian ini antara lain:

1. Mendapatkan model matematis atau algoritma untuk klasifikasi tipe kulit berdasarkan citra foto.
2. Mengeksplorasi arsitektur pada CNN dengan melakukan ekstraksi ciri untuk klasifikasi tipe kulit wajah.

### **1.4 Batasan Penelitian**

Batasan masalah yang berguna untuk tetap menjaga fokus penelitian agar tidak menyimpang dari tujuan antara lain:

1. Citra foto diambil menggunakan kamera digital atau *handphone* dengan kualitas yang baik.
2. Tipe kulit wajah diklasifikasikan menjadi empat kelompok saja yaitu kulit normal, kering, berminyak, dan kombinasi.
3. Sistem hanya dapat menentukan tipe kulit berdasarkan foto wajah manusia.

### **1.5 Sistematika Penulisan**

Sistematika penulisan dalam penelitian ini antara lain:

#### **Bab 1 Pendahuluan**

Bab 1 berisi latar belakang penelitian, rumusan masalah, tujuan penelitian, batasan penelitian, dan sistematika penulisan.

#### **Bab 2 Tinjauan Pustaka**

Bab 2 berisi tinjauan pustaka dari literatur sebelumnya yang berkaitan dan sesuai dengan penelitian yang dilakukan.

#### **Bab 3 Metodologi Penelitian**

Bab 3 berisi metodologi yang digunakan dalam penelitian.

#### **Bab 4 Hasil dan Pembahasan**

Bab 4 berisi hasil dan pembahasan dari penelitian yang telah dilakukan.

## **Bab 5 Kesimpulan dan Saran**

Bab 5 berisi kesimpulan dan saran berdasarkan hasil dari penelitian yang telah dilakukan.

## BAB 2

### Tinjauan Pustaka

#### 2.1 Kerangka Teori

Tipe kulit wajah manusia diklasifikasikan menjadi lima jenis, yaitu tipe kulit normal, kering, berminyak, kombinasi, dan sensitif (Baumann, 2015). Setiap tipe kulit wajah memiliki ciri-ciri yang berbeda. Kulit normal merupakan tipe kulit yang terlihat kusam, memiliki tekstur yang cukup kasar, terdapat benjolan-benjolan kecil (*ridges*), serta hampir tidak memiliki permasalahan. Sementara itu, tipe kulit berminyak biasanya lebih mudah berjerawat dan pori-pori yang terlihat besar diakibatkan adanya produksi sebum yang berlebihan. Sebum berkontribusi cukup besar terhadap perkembangan jerawat dibandingkan dengan tipe kulit lainnya (Baumann, 2008). Selanjutnya, tipe kulit kombinasi merupakan tipe kulit yang ada di antara tipe kulit berminyak dan kering. Beberapa area pada wajah seperti jidat, hidung, dan dagu (*T-zone*) cenderung berminyak dan area lain cenderung normal hingga kering. Terakhir, tipe kulit sensitif yang merupakan tipe kulit paling rumit di antara tipe kulit yang lain baik dalam hal karakteristik maupun perawatannya. Kulit sensitif sangat mudah bereaksi terhadap sinar matahari, debu, ataupun bahan kimia tertentu yang mengakibatkan reaksi seperti gatal bitnik, jerawat, hingga ruam kemerahan (Farage, 2019).

Klasifikasi tipe kulit wajah dibedakan berdasarkan ciri yang ditunjukkan pada kulit wajah. Beberapa penelitian telah dilakukan untuk mengetahui pembagian kelompok tipe kulit wajah seperti yang ditunjukkan pada Tabel 2.1.

Tabel 2.1 Klasifikasi Tipe Kulit Wajah Berdasarkan Ciri

No.	Sub Tema	Kata Kunci	Ulasan Kritis	Pustaka
1	Klasifikasi Tipe Kulit Wajah	Ciri tipe kulit wajah	Pengklasifikasian tipe kulit wajah berdasarkan ciri dan perawatannya. Terdapat empat tipe kulit yaitu normal, kering, berminyak, dan kombinasi. Ciri-ciri dari setiap tipe kulit	(Rostamailis, 2005)

No.	Sub Tema	Kata Kunci	Ulasan Kritis	Pustaka
			wajah masih minim dan kurang kuat.	
2	Klasifikasi Tipe Kulit Wajah	<i>Skin care tips</i>	Menyajikan panduan untuk mengetahui tipe kulit wajah beserta perawatannya agar tetap sehat. Terdapat lima tipe kulit wajah yaitu normal, kering, berminyak, kombinasi, dan sensitif. Setiap tipe kulit wajah dan ciri-ciri yang disajikan cukup lengkap.	(Cho, 2015)
3	Klasifikasi Tipe Kulit Wajah	<i>Skin type</i>	Pendeskripsian tipe kulit wajah menggunakan perbandingan seperti tipe kulit berminyak vs kering, sensitif vs tahan, berpigmen vs tidak berpigmen, dan keriput vs ketat. Penjelasan ini kurang bisa mendeskripsikan ciri dari masing-masing tipe.	(Baumann, 2010)
4	Klasifikasi Tipe Kulit Wajah	Ciri tipe kulit wajah	Penelitian ini menggunakan basis data dari pakar secara langsung. Menyajikan lima tipe kulit yaitu	(Santi & Andari, 2019)

No.	Sub Tema	Kata Kunci	Ulasan Kritis	Pustaka
			normal, berminyak, kering, kombinasi, dan sensitif. Ciri atau gejala dari setiap tipe kulit lengkap dan diperkuat dengan adanya bobot setiap gejala.	

Dari berbagai penelitian yang telah dilakukan, rata-rata pengelompokan tipe kulit yang dibedakan berdasarkan ciri-cirinya dapat dibagi menjadi lima kategori antara lain (Indriyani & Made Sudarma, 2020):

#### **2.1.1 Kulit Normal**

Kulit wajah normal memiliki ciri seperti terlihat segar dan sehat, tidak berminyak, tidak berjerawat, tekstur dan pori-pori yang samar atau halus. Kulit Normal cenderung mudah dalam memilih kosmetik dan mudah menempel pada kulit.

#### **2.1.2 Kulit Berminyak**

Kulit wajah berminyak memiliki ciri seperti pori-pori kulit besar terutama pada area *T-zone* seperti hidung, pipi, dan dagu. Kulit di bagian wajah terlihat mengilap dan kadang berjerawat atau sering ditumbuhi jerawat.

#### **2.1.3 Kulit Kering**

Kulit wajah kering memiliki ciri seperti kulit yang tidak berminyak, terlihat kering, pori-pori halus, tidak berjerawat, dan tidak lembab. Kulit kering cenderung tipis sehingga lebih mudah berkeriput dibandingkan tipe kulit yang lainnya.

#### **2.1.4 Kulit Kombinasi**

Kulit wajah kombinasi memiliki ciri seperti pori-pori kulit terlihat besar terutama pada area *T-zone* seperti dahi, hidung, dagu, dan pipi. Sebagian kulit terlihat berminyak dan sebagian lainnya terlihat kering, terkadang berjerawat, dan sulit mendapat hasil polesan kosmetik yang sempurna karena tekstur yang tidak merata.

#### **2.1.5 Kulit Sensitif**

Kulit wajah sensitif biasanya ditandai dengan kulit yang kemerahan pada area tertentu karena mudah mengalami alergi. Ciri lain kulit sensitif antara lain seperti tekstur kulit wajah yang tipis, rentan, mudah iritasi, dan mudah terluka.

## 2.2 Penelitian Terdahulu

Penelitian ini akan berfokus pada pengolahan citra untuk menentukan tipe kulit wajah berdasarkan ciri yang ada. Citra yang digunakan adalah citra foto wajah dengan kriteria tertentu.

Penelitian analisis tekstur yang bertujuan untuk klasifikasi jenis kulit wajah dilakukan menggunakan data citra wajah wanita yang diambil dengan menggunakan kamera digital. Subjek data citra merupakan wanita dengan rentang usia 17-25 tahun dan pengambilan foto dari 5 sisi wajah. Pengolahan citra diawali dengan *image acquisition* yang bertujuan untuk mendapatkan data citra yang sesuai, lalu dilakukan *preprocessing* untuk memperbaiki data citra agar siap dijadikan sebagai data input untuk proses selanjutnya. Tahap selanjutnya adalah *feature extraction* yang bertujuan untuk mendapatkan karakteristik yang pada citra. Pada proses *feature extraction* dilakukan penipisan pola, perubahan citra biner, dan sebagainya. Setelah berhasil mendapatkan karakteristik fitur, selanjutnya dilakukan klasifikasi untuk mengelompokkan data ke dalam kelompok *dry skin*, *normal to oily skin*, dan *oily skin* dengan menggunakan *K-Nearest Neighbor* (KNN). Dari penelitian yang dilakukan, akurasi terbaik yang berhasil didapatkan adalah sebesar 92,85% (Sartika et al., 2012a).

Perancangan klasifikasi dengan tujuan untuk mengklasifikasikan tipe kulit wajah telah dikembangkan. Klasifikasi menggunakan data citra wajah pembesaran 40x pada bagian dagu, hidung, pipi, dan dahi dengan menggunakan kamera mikroskopis. Setelah data citra didapatkan, selanjutnya dilakukan *preprocessing* dengan mengubah citra dari RGB menjadi *grayscale*. Selanjutnya, klasifikasi dilakukan dengan menggunakan citra pada setiap daerah wajah serta klasifikasi menggunakan citra seluruh wajah. Metode *Gray Level Co-Occurrence Matrix* (GLCM) diterapkan untuk ekstraksi ciri dan selanjutnya menggunakan metode *Support Vector Machine* (SVM) untuk mengklasifikasikan tipe kulit wajah ke dalam tiga kelompok yaitu kombinasi, berminyak, atau kering. Akurasi terbaik yang berhasil didapatkan dari penelitian ini adalah sebesar 88,89% (Firaz et al., 2018a).

Sebanyak 60 data citra mikroskopis dibagi menjadi dua yaitu sebanyak 45 citra digunakan sebagai data latih serta 15 citra digunakan sebagai data uji. Data citra ini digunakan dengan tujuan untuk mengelompokkan data citra agar dapat mengklasifikasikan antara jenis kulit normal, kering, atau berminyak. Penelitian ini diawali dengan proses menyiapkan data citra, dilanjutkan dengan *preprocessing* dengan HPF, konversi citra, serta *image enhancement*. Selanjutnya, data citra melalui proses *Fuzzy C-Means* yang berfungsi untuk pengenalan pola agar dapat mengetahui titik-titik dari data *input* serta titik tengah

kelompok. Setelah itu, dilanjutkan dengan klasifikasi citra dengan menggunakan metode *Linear Discriminant Analysis* (LDA) dan *Principal Component Analysis* (PCA). Hasil dari pengujian, ekstraksi ciri dengan metode LDA memberikan akurasi konektivitas 8 dengan nilai tertinggi yaitu sebesar 55,57%, sedangkan ekstraksi ciri dengan PCA konektivitas 4 mendapatkan nilai akurasi terendah yaitu sebesar 33,33% (Wulandari, 2019).

Penelitian untuk klasifikasi menjadi dua kelompok yaitu kulit wajah berminyak dan non berminyak telah berhasil dilakukan. Data yang digunakan adalah citra wajah yang diambil menggunakan kamera mikroskopis dengan pembesaran 100x. Data citra tersebut selanjutnya harus melalui *preprocessing* dengan cara merubah citra asli RGB menjadi *grayscale*. Setelah didapatkan citra *grayscale*, selanjutnya citra diproses menggunakan metode *Haar Wavelet* untuk ekstraksi ciri. Setelah berhasil mendapatkan ciri dari masing-masing citra, selanjutnya dilakukan klasifikasi citra dengan menggunakan metode *Support Vector Machine* (SVM) yang menghasilkan *output* berupa hasil klasifikasi citra termasuk ke dalam kelompok berminyak atau non berminyak. Hasil penelitian menunjukkan akurasi terbaik yang berhasil didapatkan sebesar 90% (Farhan et al., 2019a).

Penelitian untuk klasifikasi kulit wajah menjadi dua jenis yaitu berminyak dan kering dilakukan dengan menggunakan data citra. Citra yang diambil merupakan citra mikroskopis digital yang harus melalui proses *preprocessing* dengan tujuan untuk mempermudah proses selanjutnya dengan cara memperbaiki kualitas citra. Data citra selanjutnya melalui proses ekstraksi ciri untuk mengubah citra gambar menjadi biner serta penipisan pola dengan menggunakan *Discrete Wavelet Transform* (DWT). Setelah mendapatkan ciri dari citra, klasifikasi ke dalam kelompok berminyak atau kering dilakukan dengan menggunakan metode *Backpropagation*. Hasil dari pengujian, penelitian ini mendapatkan hasil uji akurasi sebesar 95% (Amelia et al., 2019a).

Sejumlah 30 citra mikroskopis kulit digunakan sebagai data citra dalam penelitian dengan tujuan untuk klasifikasi jenis kulit ke dalam kelompok berminyak, normal, kering, atau kombinasi. Data citra berformat jpg diambil menggunakan kamera *Microscope Digital Wi-Fi* dan *Dinolite*. Dari 30 citra tersebut, sebanyak 15 data citra digunakan sebagai data latih dan 15 data citra sebagai data uji. Selanjutnya, data citra melalui proses akuisisi data, *preprocessing*, serta ekstraksi ciri dengan menggunakan metode *Gabor Wavelet*. Setelah berhasil melakukan ekstraksi ciri, data citra selanjutnya diklasifikasikan dengan menggunakan metode *Naïve Bayes*. Berdasarkan pengujian akurasi, penelitian ini mampu mendapatkan nilai sebesar 92% (Safira et al., 2020).

Penelitian untuk melakukan klasifikasi data citra foto wajah ke dalam kelompok tipe kulit normal, berminyak, kering atau kombinasi telah dilakukan. Citra foto wajah diambil dengan menggunakan kamera digital. Data citra yang dikumpulkan tersebut selanjutnya melalui tahap *preprocessing* untuk memperbaiki kualitas citra dengan mengubah citra RGB menjadi *grayscale*, mengurangi *noise*, dan membuang bagian foto yang tidak digunakan. Setelah itu, dilakukan *feature extraction* terhadap data citra dengan menggunakan metode *Gray-level Co-Occurrence Matrix* (GLCM) untuk mendapatkan nilai *contrast* dari intensitas warna kulit. Selanjutnya, *feature extraction* untuk mengetahui tekstur kulit. Proses ini harus melalui dua tahap yaitu menggunakan *Discrete Wavelet Transform* (DWT) untuk proses dekomposisi setelah itu dilanjutkan dengan *Local Binary Pattern* (LBP) untuk ekstraksi. Setelah mendapatkan hasil ekstraksi ciri, selanjutnya data citra diklasifikasikan untuk mengelompokkan ke dalam tipe kulit wajah normal, kering, berminyak, atau kombinasi dengan menggunakan metode *Support Vector Machine* (SVM). Berdasarkan pengujian, menunjukkan hasil terbaik akurasi sebesar 91,66% (Indriyani & Made Sudarma, 2020a).

Klasifikasi tipe kulit wajah menggunakan sebanyak 1600 data citra yang diambil secara mikroskopis. Dari data citra tersebut kemudian dibagi menjadi dua yaitu sebanyak 1200 citra sebagai data latih dan 400 citra sebagai data uji. Selanjutnya, data citra harus melalui *preprocessing* untuk mengubah citra menjadi *grayscale*. Data citra *grayscale* selanjutnya melalui proses *training* dan *testing* agar dapat melakukan klasifikasi dengan menggunakan metode *Convolutional Neural Network* (CNN). Hasil klasifikasi citra menjadi empat yaitu kulit normal, kering, berminyak, dan kombinasi. Berdasarkan pengujian, akurasi terbaik yang berhasil didapatkan adalah sebesar 99,5% (Saidah et al., 2021a).

Jenis kulit wajah jika dilihat dari kadar minyaknya, dapat dibagi menjadi empat jenis yaitu kulit berminyak, kering, normal, dan kombinasi. Penelitian dengan menggunakan 1600 data citra digital yang masing-masing jenis kulitnya terdapat 400 data citra diawali dengan *preprocessing* untuk memperbaiki citra. Selanjutnya, data citra digunakan sebagai data latih menggunakan metode *GoogLeNet* dengan *22 layer deep CNN* dan dilakukan validasi untuk mengumpulkan citra baru. Dari hasil pengujian yang dilakukan, nilai akurasi terbaik adalah 99,69% dengan nilai *loss* terkecil sebesar 1,6496 (Zakiah et al., 2021).

Penelitian untuk klasifikasi menggunakan citra kulit wajah menjadi tiga kelompok yaitu tipe kulit normal, kering, dan berminyak. Data citra menggunakan foto wajah yang diambil pada bagian tertentu. Selanjutnya, data citra melalui *preprocessing* untuk mengubah citra asli menjadi *grayscale*. Setelah didapatkan citra *grayscale*, selanjutnya citra diekstraksi menggunakan metode *Gray Level Co-occurrence Matrix* (GLCM). Setelah nilai ekstraksi

didapatkan, selanjutnya proses klasifikasi menggunakan metode *Naïve Bayes*. Setelah mendapatkan model klasifikasi tipe kulit wajah menggunakan citra wajah, selanjutnya model diaplikasikan ke aplikasi android. Setelah dilakukan pengujian, hasil akurasi sebesar 85% (Alaska & Hidayat, 2021a).

Penelitian dengan tujuan untuk klasifikasi tipe kulit dengan menerapkan metode *MobileNet* telah dilakukan. Penelitian ini melakukan *transfer learning* pada tujuh penyakit kulit antara lain *psoriasis*, *acne*, *vitiligo*, *eczema*, *pityriasis rosea*, *tinea corporis*, dan *varicella (chickenpox)*. Sebanyak 3.406 data citra digunakan sebagai *dataset*, tetapi data ini dianggap tidak seimbang karena data citra pada setiap jenis penyakit memiliki jumlah yang berbeda. Data yang digunakan merupakan gabungan antara *dataset* yang diambil secara langsung dan telah divalidasi oleh dermatolog, ditambah dengan *dataset* yang didapatkan dari repositori dermatologi yang bisa diakses publik, serta *dataset* dari atlas berwarna dermatologi. Semua *dataset* yang digunakan menunjukkan lesi (jaringan abnormal) secara jelas dan berekstensi jpeg. Semua *dataset* yang dikumpulkan dibagi menjadi dua yaitu sebagai data latih sebesar 80% dan 20% sebagai data uji. Data tersebut kemudian diaplikasikan dengan model CNN *MobileNet* untuk klasifikasi. Dari hasil *preprocessing* data input, akurasi yang didapat sebesar 84,28% dan untuk *dataset* yang tidak seimbang berhasil mendapatkan akurasi sebesar 93,6%. Setelah melakukan eksplorasi pada *oversampling dataset* dan modelnya, akurasi yang berhasil didapatkan mencapai 91,8%. Selain itu, hasil dari menggunakan teknik *over sampling* dan augmentasi data pada *preprocessing*, memberikan hasil akurasi sebesar 94,4%. Model yang telah dikembangkan kemudian diaplikasikan ke dalam android.

Penelitian dengan memanfaatkan algoritma CNN untuk klasifikasi penyakit kulit wajah telah berhasil dilakukan. *Dataset* yang digunakan merupakan data citra yang diperoleh dari Xiangya-Derm dan telah divalidasi oleh tiga dokter kulit yang berpengalaman. Dari 2656 citra wajah, terdiri dari enam penyakit kulit antara lain *rosacea (ROS)*, *actinic keratosis (AK)*, *seborrhea keratosis (SK)*, *lupus eritematosa (LE)*, karsinoma sel skuamosa (SCC), dan karsinoma sel basal (BCC). Data citra diambil menggunakan kamera digital dan telah dikelompokkan berdasarkan patologi dan riwayat medis yang sesuai. Selanjutnya, dilakukan *preprocessing* dengan melakukan *crop* dan *random reserve*. Citra yang sudah melalui *preprocessing* selanjutnya melalui proses klasifikasi dengan algoritma CNN untuk dilatih dan dimodelkan. Berdasarkan hasil pengujian menggunakan 388 data citra, akurasi terbaik yang berhasil didapatkan mencapai 98,43% untuk SK, 2,9% untuk LE, 89,2% untuk BCC, serta rata-rata *recall* dan presisi mencapai 77,0% dan 70,8% (Wu et al., 2019).

Penelitian dengan metode CNN untuk klasifikasi penyakit kulit *Guttate Psoriasis* telah berhasil dilakukan. Sebanyak 187 data citra yang digunakan sebagai *dataset* didapatkan dari *psoriasis image library* yang terdapat pada situs web *International Psoriasis Council* (IPC) serta 82 data citra *Plaque Psoriasis* dan 105 data citra *Guttate Psoriasis* yang didapatkan dari DermNet MZ. *Dataset* yang berhasil dikumpulkan selanjutnya melalui *preprocessing* untuk mengubah ukuran citra agar seragam. Setelah itu, dengan memanfaatkan metode CNN, data citra kemudian dilatih dan dimodelkan yang kemudian menghasilkan *output* berupa klasifikasi penyakit kulit. Berdasarkan pengujian, akurasi yang didapatkan sebesar 72,4% (Roslan et al., 2020).

Citra kulit wajah dapat dikelompokkan berdasarkan permasalahannya menjadi jerawat, flek, dan kerutan. Sebanyak 40 data citra yang telah dikumpulkan dari masing-masing permasalahan kulit, digunakan sebagai *dataset* dan kemudian telah melalui proses *labelling*. Selanjutnya, *dataset* melalui *preprocessing* untuk mengubah citra asli menjadi *grayscale* serta *resize* citra menjadi 80 pixel. Setelah itu, data citra melalui proses ekstraksi fitur dengan memanfaatkan metode *Histogram of Oriented Gradient* (HOG). Berdasarkan hasil ekstraksi, data citra kemudian melalui proses klasifikasi dengan memanfaatkan metode SVM. Berdasarkan pengujian, akurasi terbaik yang berhasil didapatkan sebesar 98% (Sinaulan & Hantara, 2021).

Beberapa penyakit kulit wajah yang sering dijumpai antara lain jerawat, milia, herpes, eksim, dan melasma. Berdasarkan beberapa penyakit tersebut, penelitian berhasil dilakukan untuk mengelompokkan penyakit serta mengetahui perbedaannya. Dalam penelitiannya, data citra yang digunakan sebanyak 500 citra yang kemudian dibagi menjadi data latih sebanyak 325 citra dengan pembagian jumlah data citra yang merata pada setiap kelompoknya, serta sebanyak 125 data citra digunakan untuk data validasi yang masing-masing kelompoknya memiliki jumlah yang rata. Tahap awal pengolahan data citra adalah *preprocessing* untuk menyamakan posisi citra (*rotate*) dan augmentasi data untuk mendapatkan *batch* dari citra. Selanjutnya, data citra yang sudah melalui proses melalui pelatihan dan pengelompokan dengan memanfaatkan metode CNN. Dari hasil pengujian yang dilakukan, berhasil didapatkan akurasi sebesar 90% (Nurkhasanah & Murinto, 2021).

Metode *deep learning* dapat dimanfaatkan untuk klasifikasi penyakit kulit. Sebanyak 1.137 data citra didapatkan dari dr. Gerbi dan sebanyak 239 data citra didapatkan dari Boru Meda General Hospital yang kemudian dikumpulkan sebagai *dataset*. Selanjutnya, *dataset* tersebut harus melalui *preprocessing* untuk menyamakan ketajaman warna dan ukuran citra. Setelah itu, *dataset* dibagi menjadi tiga bagian yaitu sebanyak 80% digunakan sebagai data

latih, 10% data validasi, dan 10% lainnya digunakan sebagai data uji. Data latih kemudian dilakukan *training data* dengan memanfaatkan metode *MobileNet-v2* untuk mengenali ciri dari kulit normal atau kulit abnormal. Klasifikasi jenis kulit ke dalam kelompok normal atau abnormal memberikan hasil akurasi sebesar 97,5% (Muhaba et al., 2022).

Penelitian yang bertujuan untuk deteksi kanker kulit dengan memanfaatkan citra foto telah dilakukan. Data citra foto digital berjumlah 132 yang terdiri dari kanker kulit ganas sebanyak 71 data citra dan 61 lainnya merupakan data citra kanker kulit jinak. *Dataset* selanjutnya melalui pemrosesan data dengan *resizing* untuk mengubah ukuran citra agar sama. Selanjutnya, dilakukan ekstraksi fitur untuk mengetahui ciri dari citra dan dilanjutkan dengan klasifikasi citra untuk mengelompokkan ke dalam kanker kulit ganas atau jinak dengan memanfaatkan metode CNN. Akurasi terbaik yang bisa didapatkan adalah sebesar 97,73% dengan *loss* sebesar 1,7063 (Sofia Saidah et al., 2022).

Berdasarkan penelitian-penelitian terdahulu yang telah diuraikan, dapat diketahui bahwa faktor-faktor yang mendukung hasil akurasi yang baik dalam pengolahan citra adalah kualitas data citra yang dapat didukung dengan *preprocessing* yang baik, serta kuantitas data citra yang digunakan. Semakin banyak data citra yang digunakan, maka semakin banyak pula suatu mesin dapat belajar. Semakin banyak mesin belajar, maka hasil akurasi dari model akan semakin tinggi. Di sisi lain, penggunaan metode yang tepat dalam ekstraksi ciri serta klasifikasi dalam pemodelan merupakan salah satu faktor yang berpengaruh pada hasil akurasi. Ringkasan penelitian terdahulu disajikan pada Tabel 2.2, dan Tabel 2.3.

Tabel 2.2 Klasifikasi Tipe Kulit Wajah dengan Citra Mikroskopis

No.	Sub Tema	Kata Kunci	Ulasan Kritis	Pustaka
1	Deteksi Tipe Kulit Wajah dengan Citra Mikroskopis	Tipe kulit wajah, <i>image processing</i> , deteksi citra	Pengelompokan tipe kulit dalam penelitian ini hanya dibagi ke dalam kulit berminyak atau tidak berminyak saja. Akurasi terbaik yang dihasilkan sistem adalah 92,85%.	(Sartika et al., 2012b)
2	Deteksi Tipe Kulit Wajah	Tipe kulit wajah, <i>image</i>	Pengklasifikasian citra dibagi menjadi tiga kelompok saja yaitu	(Firaz et al., 2018b)

No.	Sub Tema	Kata Kunci	Ulasan Kritis	Pustaka
	dengan Citra Mikroskopis	<i>processing</i> , deteksi citra	tipe kulit berminyak, kering, dan kombinasi. Data latih dan data uji masih minim, tetapi dari pengujian sistem menghasilkan akurasi yang cukup baik yaitu sebesar 88,89%.	
3	Deteksi Tipe Kulit Wajah dengan Citra Mikroskopis	Tipe kulit wajah, <i>image processing</i> , deteksi citra	Tipe kulit hanya dibagi menjadi dua kelompok yaitu kulit berminyak dan kering. Melalui ekstraksi ciri dan klasifikasi yang dilakukan memberikan hasil yang cukup baik. Berdasarkan pengujian, akurasi maksimal yang didapatkan sebesar 90%.	(Farhan et al., 2019b)
4	Deteksi Tipe Kulit Wajah dengan Citra Mikroskopis	Tipe kulit wajah, <i>image processing</i> , deteksi citra	Klasifikasi hanya dibagi menjadi dua kelompok yaitu berminyak atau kering, sehingga kurang lengkap. Data latih dan data uji memberikan hasil cukup baik sehingga melalui pengujian bisa didapatkan hasil akurasi sebesar 95%.	(Amelia et al., 2019b)

No.	Sub Tema	Kata Kunci	Ulasan Kritis	Pustaka
5	Deteksi Tipe Kulit Wajah dengan Citra Mikroskopis	<i>Face skin type, image processing</i>	Klasifikasi citra cukup lengkap karena terbagi menjadi empat kelompok yaitu kulit normal, berminyak, kering, atau kombinasi. Dengan adanya data <i>training</i> dan <i>testing</i> yang digunakan dalam metode SVM memberikan hasil akurasi terbaik sebesar 99,5%.	(Saidah et al., 2021b)

Tabel 2.3 Klasifikasi Tipe Kulit Wajah dengan Citra Foto

No.	Sub Tema	Kata Kunci	Ulasan Kritis	Pustaka
1	Deteksi Tipe Kulit Wajah dengan Citra Foto	<i>Face skin type, image processing</i>	Tujuan klasifikasi citra wajah cukup lengkap karena dibagi menjadi empat kelompok yaitu kulit normal, berminyak, kering, atau kombinasi. Sampel wajah yang digunakan kurang merata dari segi usia, hanya untuk dewasa saja. Metode yang digunakan memberikan hasil akurasi klasifikasi rata-rata sebesar 91,66%.	(Indriyani & Made Sudarma, 2020b)

No.	Sub Tema	Kata Kunci	Ulasan Kritis	Pustaka
2	Deteksi Tipe Kulit Wajah dengan Citra Foto	<i>Face skin type, image processing, skin tone clustering</i>	Pengklasifikasian citra menggunakan metode <i>Naive bayes</i> . Klasifikasi tipe kulit kurang lengkap karena hanya dibagi menjadi tiga kelompok yaitu normal, kering, berminyak. Lebih berfokus pada pengembangan aplikasi android daripada pengolahan citra. Tidak dijelaskan berapa banyak data <i>training</i> yang digunakan. Berdasarkan 20 data uji, memberikan hasil akurasi sebesar 85%.	(Molina et al., 2020)
3	Deteksi Tipe Kulit Wajah dengan Citra Foto	<i>Skin type classification</i>	Penelitian yang dilakukan memiliki tujuan untuk mengklasifikasikan warna kulit melalui citra foto wajah sehingga penelitian ini kurang sejalan dengan tujuan penelitian yang akan dilakukan. Tetapi, konsep pemrosesan citra yang dilakukan dapat digunakan	(Alaska & Hidayat, 2021b)

No.	Sub Tema	Kata Kunci	Ulasan Kritis	Pustaka
			sebagai acuan untuk penelitian.	

Dari berbagai penelitian terdahulu yang telah dipaparkan, dapat dilihat bahwa penelitian dilakukan dengan memanfaatkan teknik *image classification* menggunakan data citra mikroskopis maupun citra foto. Secara umum, penelitian terdahulu mengelompokkan citra kulit wajah ke dalam beberapa tipe kulit seperti normal, tidak normal, berminyak, kering, atau kombinasi. Metode yang digunakan bervariasi dengan menyesuaikan dengan data citra dan tujuan dari masing-masing penelitian. Secara umum, proses penelitian melalui *preprocessing* untuk menstandarkan dan memperbaiki kualitas data citra lalu dilanjutkan dengan klasifikasi ke dalam beberapa kelompok tipe kulit wajah.

Tabel 2.4 Klasifikasi Tipe Kulit Wajah Berdasarkan Kadar Minyak dan Penyakit

Jumlah Dataset	Metode	Kelas	Hasil Akurasi	Faktor Tinggi/Rendah Akurasi
60	SVM	Normal, Berminyak, Kering, Kombinasi	91,66%	Data yang dikumpulkan masih cenderung minim, namun data cukup seimbang (Indriyani & Made Sudarma, 2020a).
140	<i>K-Nearest Neighbor</i>	Berminyak, Kering, Kombinasi	92,85%	Ekstraksi ciri tekstur dan gabungan antara orde 1 dan orde 2 (Sartika et al., 2012a).
100	SVM	Berminyak, Kering, Kombinasi	88,89%	Data latih masih minim, tetapi hasil cukup baik (Firaz et al., 2018a).
60	LDA	Normal, Berminyak, Kering	55,57%	Data citra minim, ekstraksi ciri LDA memberikan akurasi lebih baik dibandingkan hasil akurasi PCA (Wulandari, 2019).
112	SVM	Berminyak, Non Berminyak	90%	Dari pemodelan memberikan akurasi terbaik pada level 2 koefisien HH (Farhan et al., 2019a).
80	<i>Backpropagation</i>	Berminyak, Kering	95%	Jumlah data, parameter, nilai <i>epoch</i> , <i>learning rate</i> , dan <i>goal number</i> pada klasifikasi memengaruhi hasil akurasi (Amelia et al., 2019a).
30	<i>Naïve Bayes</i>	Normal, Berminyak, Kering, Kombinasi	92%	Hasil akurasi dan klasifikasi bergantung pada kualitas citra (sudut pengambilan gambar, kestabilan objek, dan kondisi lingkungan) (Safira et al., 2020).
1600	CNN	Normal, Berminyak,	99,5%	Banyaknya data citra yang digunakan serta metode yang

		Kering, Kombinasi		digunakan sangat sesuai (Saidah et al., 2021a).
1600	CNN	Normal, Berminyak, Kering, Kombinasi	99,69%	Kuantitas dan kualitas data citra yang digunakan (Zakiah et al., 2021).
20	<i>Naïve Bayes</i>	Normal, Berminyak, Kering	85%	Kualitas data citra yang baik seperti pencahayaan gambar, fokus, sudut pengambilan, derau, dan perangkat untuk pengambilan citra (Alaska & Hidayat, 2021a).
3406	CNN	<i>Acne, Eczema, Pityriasis Rosea, Psoriasis, Tinea Corporis, Varicella (Chickenpox), Vitiligo</i>	94,4%	Pengambilan dan pemrosesan data citra menggunakan <i>oversampling</i> serta augmentasi (Velasco et al., 2019).
2656	CNN	<i>Seborrhea Keratosis (SK), Actinic Keratosis (AK), Rosacea (ROS), Lupus Eritematosa (LE), Karsinoma Sel Basal (BCC), Karsinoma Sel Skuamosa (SCC)</i>	92,9%	Kuantitas yang cukup dan kualitas data citra yang digunakan sangat baik (Reddy & Manohar, 2019).
187	CNN	<i>Plaque Psoriasis</i> atau <i>Gutatte Psoriasis</i>	72,4%	Parameter yang digunakan dalam pemodelan metode CNN cocok untuk data minim (Roslan et al., 2020).
40	SVM	Flek, Jerawat, Kerutan	98%	Model yang digunakan untuk klasifikasi sesuai dengan data yang dimiliki (Sinaulan & Hantara, 2021).
500	CNN	Jerawat, Eksim, Melasma, Herpes, Milia	90%	Kuantitas data citra yang digunakan cukup untuk pemodelan dengan metode yang dipilih (Nurkhasanah & Murinto, 2021).
1137	CNN	Normal, Abnormal	97,5%	Model yang digunakan untuk klasifikasi sesuai dengan data yang dimiliki (Muhaba et al., 2022).
132	CNN	Kanker Kulit, Ganas, Jinak	97,73%	Model yang digunakan untuk klasifikasi cocok dengan kualitas data (Sofia Saidah et al., 2022).

Dapat dilihat pada Tabel 2.4 bahwa mayoritas pemodelan untuk klasifikasi citra sebagian besar menggunakan metode CNN. Dari hasil penelitian-penelitian yang sudah

dilakukan sebelumnya, dapat dilihat bahwa metode CNN mampu memberikan hasil pemodelan yang baik pada data citra yang minim hingga yang besar. Hal ini dapat dibuktikan dengan hasil akurasi yang cukup tinggi pada pemodelan yang menggunakan CNN. Sedangkan pada metode SVM, memberikan hasil akurasi yang baik tetapi memiliki kekurangan yaitu ketika menggunakan data yang besar maka akurasi akan semakin menurun. Berdasarkan hasil analisis tabel perbandingan, metode yang paling potensial untuk klasifikasi data citra adalah metode CNN yang sudah banyak digunakan serta dibuktikan oleh peneliti sebelumnya. Rata-rata hasil pemodelan CNN memberikan hasil akurasi yang baik walaupun dengan menggunakan data citra yang sedikit ataupun banyak sekaligus. Tetapi, baik atau tidaknya akurasi tidak hanya ditentukan dari metode yang digunakan, tetapi juga dari kuantitas dan kualitas data citra yang digunakan (Kusumaningrum et al., 2023).

### **2.3 Konsep Pengetahuan yang Digunakan**

Konsep pengetahuan yang digunakan dalam penelitian ini adalah dengan memanfaatkan metode *Convolutional Neural Network* (CNN). Arsitektur jaringan saraf tiruan yang digunakan pada metode CNN yaitu MobileNetV1, MobileNetV2, dan MobileNetV3-Large.

#### **2.3.1 *Convolutional Neural Network* (CNN)**

*Convolutional Neural Network* (CNN) merupakan salah satu metode klasifikasi dalam jaringan syaraf tiruan. CNN dapat dimanfaatkan untuk melakukan klasifikasi gambar dengan mengenali ciri-ciri dari setiap gambar pada suatu kelompok. Klasifikasi ini biasanya dilakukan untuk mengidentifikasi wajah, penyakit, individu, maupun objek gambar lainnya (Nissa et al., 2021). Berdasarkan beberapa penelitian serupa yang telah dilakukan sebelumnya, metode CNN dinilai sesuai untuk melakukan klasifikasi untuk menentukan tipe kulit wajah berdasarkan citra karena mampu memberikan hasil akurasi yang baik (Kusumaningrum et al., 2023).

#### **2.3.2 MobileNet**

MobileNet merupakan model arsitektur yang dikembangkan dalam pemodelan dengan metode CNN. Model arsitektur ini biasa digunakan untuk deteksi objek, klasifikasi gambar, ataupun segmentasi semantik. Tujuan adanya arsitektur ini adalah untuk menyeimbangkan antara akurasi dan efisiensi dari model yang dikembangkan sehingga model dapat bekerja secara efektif walaupun pada perangkat dengan sumber daya terbatas (Howard et al., 2017). MobileNet memiliki tiga varian yang memiliki penyesuaian serta peningkatan dari masing-masing versi, berikut perbedaan masing-masing versi dari MobileNet:

a. MobileNetV1

Pada versi pertama, untuk mengurangi jumlah parameter serta meningkatkan efisiensinya, memanfaatkan operasi konvolusi *Depthwise Separable Convolution*. Selain itu, dalam MobileNetV1 memanfaatkan *linear bottleneck design* yang bertujuan untuk mengurangi beban komputasi dengan cara mengurangi dimensi ruang fitur baik sebelum maupun sesudah konvolusi. MobileNetV1 digunakan untuk pemodelan yang sederhana dan efisiensi komputasi.

b. MobileNetV2

MobileNetV2 merupakan perbaikan dari MobileNet versi sebelumnya. Arsitektur ini melakukan perbaikan untuk meningkatkan efisiensi dan kinerja model dengan menggunakan *bottleneck blocks* dengan *residual connections*. Selain itu, dalam MobileNetV2 juga terdapat konsep *inverted residual blocks* dengan *linear bottleneck* dengan tujuan untuk meningkatkan efisiensi dan akurasi model. MobileNetV2 digunakan untuk pemodelan yang lebih kompleks serta efisiensi dan akurasi yang lebih baik dibandingkan dengan MobileNetV1.

c. MobileNetV3-Large

MobileNetV3-Large merupakan perkembangan dan inovasi yang signifikan dari MobileNet versi sebelumnya. Perkembangan yang terlihat antara lain dari penggunaan teknik *squeeze-and-excitation* dan desain blok *residual* untuk menambahkan kapasitas yang mampu mewakili atau representatif. Arsitektur ini menggunakan lebih banyak komputasi dan parameter dengan tujuan untuk kinerja yang lebih tinggi atau kompleks. MobileNetV3-Large digunakan untuk pemodelan dengan *dataset* yang lebih banyak dan kompleks serta komputasi yang lebih berat dibandingkan MobilNetV1 dan MobileNetV3-Large.

### 2.3.3 Kualitas dan Kuantitas Data Citra

Kualitas dan kuantitas data citra mempengaruhi keberhasilan pemodelan serta hasil akurasi yang didapatkan model. Citra dengan kualitas yang buruk cenderung memberikan pengaruh negatif terhadap performa model yang dikembangkan (Hanjaya, 2023). Kualitas data citra yang rendah dapat disebabkan beberapa faktor seperti teknis maupun lingkungan yang memberikan pengaruh negatif terhadap pengolahan citra (Syahrul & Putra, n.d.). Salah satu penyebab kurangnya kualitas data citra dikarenakan minimnya cahaya dan perbedaan cahaya yang diterima pada citra. Intensitas cahaya yang diterima citra pada pengambilan gambar di luar ruangan berpengaruh dalam pemodelan yang melakukan pengelompokan berdasarkan ciri warna (Putro & Hermawan, 2021). Berdasarkan hasil pengambilan citra dengan

mempertimbangkan cahaya, semakin terang intensitas cahaya yang diterima objek maka persentase tingkat pengenalan citra semakin tinggi dan sebaliknya, jika intensitas cahaya minim maka persentase tingkat pengenalan citra semakin kecil (Wijanarko A.P, 2014). Hasil pemodelan dengan menggunakan data gambar yang diambil pada pagi hari memiliki akurasi yang lebih tinggi dibandingkan dengan data gambar yang diambil pada sore hari, sehingga waktu pengambilan gambar berpengaruh pada kualitas gambar dan akurasi yang dihasilkan dari pemodelan (Putro & Hermawan, 2021).

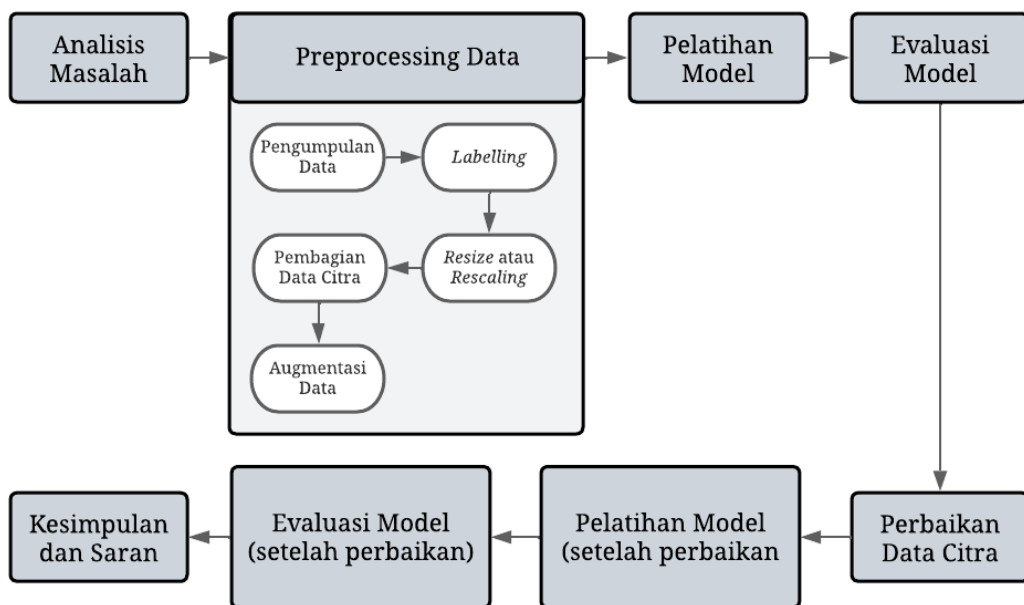
Selain kualitas, kuantitas juga mempengaruhi hasil kinerja model. Distribusi kuantitas data citra yang tidak seimbang untuk setiap kelas memberikan efek negatif bagi kinerja model. Kelas dengan data citra yang lebih minim menjadi salah satu sumber penurunan kinerja pada saat penerapan model (De Angeli et al., 2022). Ketika data antar kelas pada pemodelan tidak seimbang, maka dapat dilakukan teknik augmentasi data citra seperti (*shareRange*, *horizontalFlip*, *widthShift*, *verticalFlip* dan *rotationRange*) (Resa Arif Yudianto et al., 2022).

# BAB 3

## Metodologi

### 3.1 Tahapan Penelitian

Tahapan penelitian menunjukkan proses penelitian yang dilalui, mulai dari analisis masalah, *preprocessing data*, pelatihan model, dan yang terakhir evaluasi model sehingga mampu memberikan kesimpulan dan saran dari penelitian yang telah dilakukan. Detail tahap penelitian dapat dilihat pada Gambar 3.1.



Gambar 3.1 Tahapan Penelitian

### 3.2 Analisis Masalah

Tahap analisis masalah dilakukan untuk mengetahui permasalahan yang dijadikan sebagai dasar dilakukannya penelitian. Pada tahap ini dilakukan studi literatur yang berkaitan dengan penelitian serta menganalisis kebutuhan untuk pemodelan.

#### 3.2.1 Studi Literatur

Tahap studi literatur dilakukan untuk mencari literatur berupa buku, jurnal, laporan, atau beberapa sumber lain. Literatur yang digunakan merupakan literatur yang berkaitan dengan penelitian sehingga mampu menjadi dasar dilakukan penelitian ataupun sumber pengetahuan.

### 3.2.2 Analisis Kebutuhan

Tahap analisis kebutuhan dilakukan dengan tujuan untuk memahami kebutuhan dalam penelitian. Analisis kebutuhan yang diperlukan antara lain mengetahui sasaran pengguna, kriteria data citra yang dibutuhkan, dan kriteria model.

Pengukuran besaran sampel untuk menentukan banyaknya data citra yang dibutuhkan dalam penelitian ini menggunakan rumus dari Krejcie & Morgan. Rumus pada persamaan (3.1) menentukan besaran sampel berdasarkan jumlah populasi yang ada (Krejcie & Morgan, 1970).

$$S = \frac{\chi^2 NP(1 - P)}{d^2(N - 1) + \chi^2 P(1 - P)} \quad (3.1)$$

Keterangan:

- $S$  : Ukuran sampel
- $N$  : Ukuran populasi
- $\chi^2$  : Nilai Chi Square
- $P$  : Proporsi populasi
- $d$  : Galat

Populasi yang digunakan dalam penelitian ini adalah seluruh warga Indonesia berjenis kelamin laki-laki dan perempuan dengan usia 10 – 59 tahun. Berdasarkan data sensus penduduk tahun 2020 yang dilakukan oleh Badan Pusat Statistik (BPS), total populasi dari kelompok umur tersebut sebanyak 203.132.261 penduduk (Badan Pusat Statistik, 2021). Berdasarkan data dari BPS dan rumus dari Krejcie & Morgan sebagai acuan, besaran sampel yang dibutuhkan dapat dihitung seperti pada persamaan (3.2), persamaan (3.3), dan persamaan (3.4) sebagai berikut:

$$S = \frac{3,84 \times 203.132.261 \times 0,5(1 - 0,5)}{0,05^2(203.132.261 - 1) + 3,84 \times 0,5(1 - 0,5)} \quad (3.2)$$

$$S = \frac{195.006.970,56}{507.831,61} \quad (3.3)$$

$$S = 383,9 \text{ dibulatkan menjadi } 384 \quad (3.4)$$

Berdasarkan data-data yang telah dipaparkan beserta dengan perhitungan, untuk mendapatkan hasil yang maksimal dan tepat sasaran, diperlukan data citra dengan kriteria sebagai berikut:

- a. Usia 12 tahun (telah mengalami pubertas) - 59 tahun
- b. Total data citra yang dibutuhkan minimal 384 sampel. Kelompok rentang usia yang digunakan adalah usia < 20 tahun, 20-29 tahun, 30-39 tahun, 40-49 tahun,

dan 50-59 tahun. Untuk itu, masing-masing kelompok rentang usia memerlukan setidaknya 77 sampel.

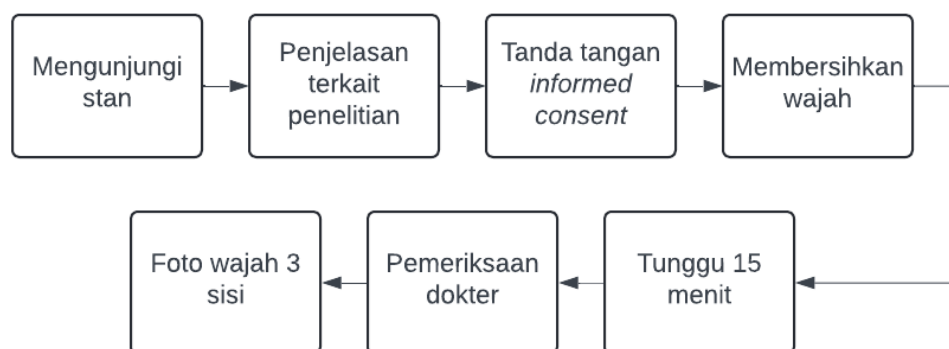
- c. Citra wajah diambil dari 3 sisi yaitu depan, samping kanan, dan samping kiri.
- d. Berformat PNG, JPG, atau JPEG.
- e. Kualitas citra yang baik diambil dengan ketentuan sebagai berikut:
  - Pencahayaan yang merata
  - Tidak blur (fokus)
  - Foto masih asli (tanpa filter dan tidak dimodifikasi)
  - Memiliki resolusi tinggi minimal 720p (1280 x 720)
  - Wajah bersih atau tidak menggunakan *makeup*
  - Wajah terlihat secara menyeluruh, tidak tertutup rambut, hijab, atau objek lainnya (Suryaningsih, 2018).

### 3.3 Preprocessing Data

*Dataset* yang digunakan dalam pemodelan adalah data citra kulit wajah. Pada tahap ini dilakukan *preprocessing* data mulai dari pengumpulan data citra yang dilakukan untuk mendapatkan *dataset* yang akan digunakan dalam pemodelan hingga augmentasi data.

#### 3.3.1 Pengumpulan Data

Pengumpulan data citra telah dilakukan pada tanggal 8 Oktober 2022 pada acara Avorestation yang dilaksanakan di GOR Universitas Negeri Yogyakarta bersamaan dengan diadakannya acara Avorestation. Proses pencarian *dataset* dilakukan dengan mendirikan stan pemeriksaan kulit wajah gratis bersama dengan dr. Rozan. Pengunjung yang tertarik dan bersedia untuk menjadi subjek penelitian dengan cara difoto wajahnya. Alur pencarian *dataset* ditunjukkan pada Gambar 3.2.



Gambar 3.2 Alur Pencarian *Dataset*

- a. Subjek mengunjungi stan  
Pengunjung yang berkenan untuk menjadi subjek diarahkan untuk datang ke stan pemeriksaan. Syarat pengunjung yang diperbolehkan menjadi subjek adalah sesuai dengan kriteria yang telah ditentukan pada analisis kebutuhan.
- b. Subjek diberikan penjelasan terkait penelitian  
Pengunjung diberikan penjelasan terkait penelitian dan apa saja yang akan dilakukan dalam pengambilan data. Detail penjelasan terkait penelitian yang disampaikan kepada calon subjek terdapat pada LAMPIRAN A.
- c. Subjek diminta menandatangani persetujuan untuk menjadi  
Pengunjung yang sudah bersedia menjadi subjek dan sudah memahami penjelasan terkait penelitian kemudian diarahkan untuk menandatangani *informed consent*. *Informed consent* sebagai tanda persetujuan sebagai subjek ditunjukkan pada LAMPIRAN B.
- d. Subjek diminta untuk membersihkan wajah agar bersih dari *skincare* dan *makeup*.  
Subjek yang telah menandatangani formulir persetujuan menjadi subjek penelitian kemudian diarahkan untuk membersihkan wajah menggunakan *micellar water*. Pembersihan ini dilakukan agar wajah subjek terbebas dari *makeup* dan debu sehingga wajah akan memperlihatkan kondisi asli kulit wajah. *Micellar water* *Micellar water* yang digunakan adalah merek Bioderma Sensibio H2O yang cocok untuk segala tipe kulit. Kandungan unsur molekul yang hampir menyerupai kulit, Bioderma Sensibio H2O *micellar water* mampu membantu melindungi lapisan pertahanan kulit yang disesuaikan dengan pH alami kulit yaitu 5,5 (Taieb et al., 2021).
- e. Subjek menunggu 15 menit  
Subjek diminta untuk menunggu selama 15 menit. Hal ini dilakukan agar kulit wajah dapat mengeluarkan minyak kulit secara alami setelah dibersihkan.
- f. Subjek melakukan pemeriksaan  
Subjek diarahkan untuk melakukan pemeriksaan bersama dengan dokter. Pemeriksaan berupa pengecekan terhadap kulit wajah yang dilihat dengan mata serta pengecekan kelembaban kulit pada lima titik yang berbeda yaitu jidat, pipi kanan, pipi kiri, hidung, dan dagu dengan menggunakan alat SK-IV Digital Moisture .
- g. Subjek melakukan foto wajah  
Pengambilan data citra menggunakan kamera digital dan kamera *handphone* dengan sudut pengambilan citra secara *eye level* dengan sudut sebesar 90° untuk

pengambilan dari depan, 45° untuk pengambilan citra dari samping kiri, serta 135° untuk pengambilan data citra dari samping kanan.

Dari pengumpulan data citra, didapatkan sebanyak 147 data citra yang diambil dari tiga sisi yang berbeda dari 49 orang yang bersedia menjadi subjek penelitian. Data citra yang dikumpulkan terdiri dari 12 orang laki-laki dan 37 orang perempuan dengan umur, jenis kelamin, dan tipe kulit yang beragam. Tabel 3.1 menunjukkan perincian data citra yang berhasil dikumpulkan.

Tabel 3.1 Perincian Data Citra *Full Wajah*

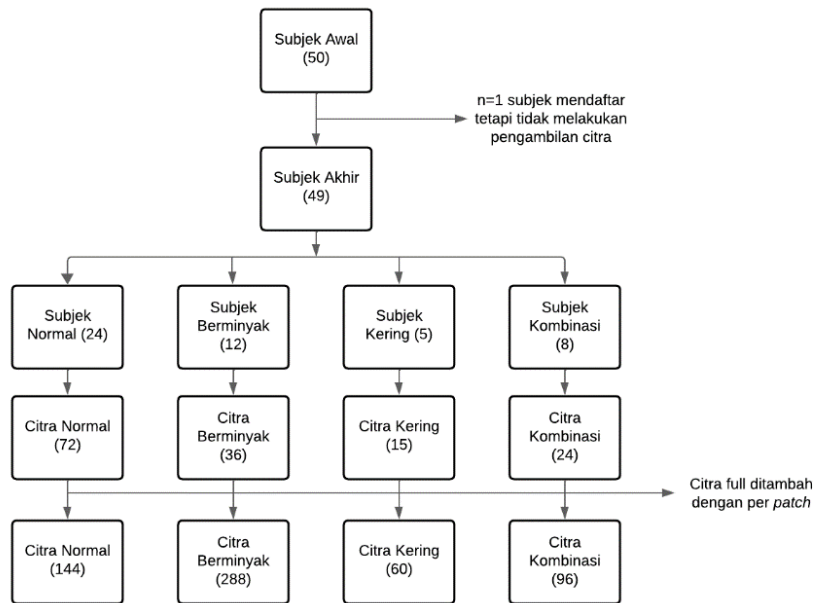
No.	Tipe Kulit	Jumlah Subjek	Jumlah Data Citra
1	Normal	24	72
2	Berminyak	12	36
3	Kering	5	15
4	Kombinasi	8	24
<b>Total</b>		49	147

Data citra yang dihasilkan selanjutnya dilakukan pemotongan area pada lima titik wajah yaitu jidat, hidung, dagu, pipi kiri, dan pipi kanan. Total data citra yang didapatkan adalah sebanyak 588 data citra. Perincian data citra ditunjukkan pada Tabel 3.2

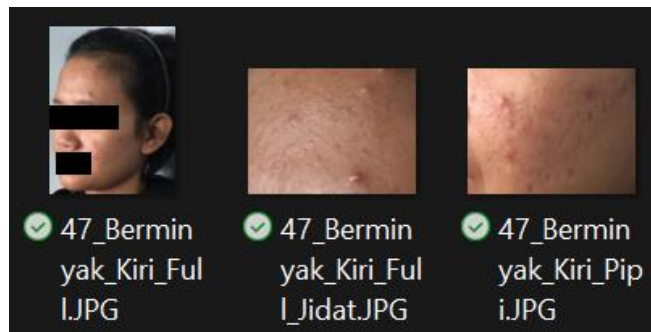
Tabel 3.2 Perincian Data Citra *Full Wajah dan Patch*

No.	Tipe Kulit	Jumlah Subjek	Jumlah Data Citra
1	Normal	24	144
2	Berminyak	12	288
3	Kering	5	60
4	Kombinasi	8	96
<b>Total</b>		49	588

Visualisasi alur pengumpulan data citra ditunjukkan pada Gambar 3.3. Alur pengumpulan data citra dimulai dari jumlah subjek awal, dilanjutkan jumlah subjek akhir yang didapatkan hingga jumlah data citra yang berhasil dikumpulkan setiap tipe kulit. Hasil pengumpulan data citra dicontohkan pada Gambar 3.4.



Gambar 3.3 Alur Pengumpulan Data Citra



Gambar 3.4 Contoh *Dataset* yang Dikumpulkan

Dari data citra yang telah didapatkan, selanjutnya perlu dilakukan persiapan data citra yang akan digunakan dalam pemodelan. Proses persiapan data antara lain:

### 3.3.2 Labelling

Data citra yang dikumpulkan selanjutnya dilakukan *labelling* oleh tenaga medis yang dalam hal ini adalah dr. Rozan. *Labelling* citra memiliki empat anotasi antara lain *id* citra, jenis kelamin, sisi pengambilan citra, serta tipe kulit yang ditentukan dari pengamatan dan ciri-ciri yang terlihat pada kulit wajah saat pemeriksaan oleh dokter yang masing-masing anotasi dipisahkan dengan simbol *underscore* “\_”. Tipe kulit wajah dibedakan ke dalam empat kelompok yaitu tipe kulit normal, berminyak, kering, atau kombinasi. Penjelasan untuk *labelling* antara lain sebagai berikut:

- *id* citra merupakan anotasi untuk nomor urut pengambilan gambar citra dengan menggunakan angka dimulai dari angka satu, contoh: 1\_...\_...\_....
- tipe kulit wajah dalam *labelling* menggunakan Normal, Berminyak, Kering, atau Kombinasi, contoh: 1\_Berminyak\_...\_....

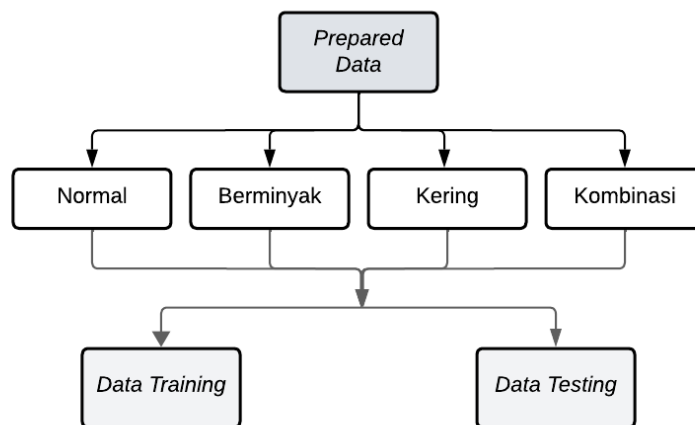
- sudut pengambilan gambar menggunakan “Depan”, “Kiri”, atau “Kanan”, contoh: 1\_Berminyak\_Kiri\_....
- bagian pengambilan citra dengan menggunakan “Full” untuk penuh satu wajah, “Jidat” untuk potongan area jidat, “Hidung” untuk potongan area hidung, “Dagu” untuk potongan area dagu, “PipiKiri” untuk potongan area pipi kiri, dan “PipiKanan” untuk potongan area pipi kanan, contoh: 1\_Berminyak\_Kiri\_PipiKanan

### 3.3.3 *Resize* atau *Rescaling* Citra

Data citra yang didapatkan memiliki ukuran yang beragam yang beragam, sehingga diperlukan *resize* citra ke dalam ukuran tertentu. Selain itu, adanya perbedaan dimensi perlu dilakukan *rescaling* intensitas pixel pada data citra sehingga data citra yang digunakan memiliki ukuran dan dimensi yang seragam.

### 3.3.4 Pembagian Data Citra

Pengelompokan data merupakan proses mempersiapkan data citra sebelum siap digunakan untuk pemodelan dengan membagi setiap kelas berdasarkan label yang telah diberikan. Struktur folder pengelompokan data ditunjukkan pada Gambar 3.5.



Gambar 3.5 Struktur Folder Pengelompokan Data

Berdasarkan label yang telah diberikan, data citra kemudian dibagi ke dalam empat folder terpisah sesuai dengan tipe kulit wajah yaitu “Normal”, “Berminyak”, “Kering”, dan “Kombinasi”. Setelah dilakukan pembagian data citra berdasarkan tipe kulit tersebut, dilakukan *split data* yang bertujuan untuk membagi data citra menjadi *data training* serta *data testing*. Dikarenakan minimnya data citra yang dimiliki, persentase pembagian data citra adalah 80% untuk *data training* dan 20% untuk *data testing*. Tabel 3.3 menunjukkan perincian jumlah data citra yang sudah dibagi ke dalam *data training* dan *data testing* yang selanjutnya digunakan dalam pemodelan.

Tabel 3.3 Perincian *Data Training* dan *Data Testing*

No.	Tipe Kulit	Jumlah <i>Data Training</i>	Jumlah <i>Data Testing</i>
1	Normal	231	57
2	Berminyak	116	28
3	Kering	48	12
4	Kombinasi	77	19
<b>Total</b>		472	116

Pada folder *data training* terdapat sebanyak 472 data citra serta pada folder data testing terdapat sebanyak 116 data citra, yang kemudian kedua folder tersebut terbagi lagi ke dalam folder masing-masing tipe kulit.

### 3.3.5 Augmentasi Data

Kondisi *dataset* yang belum seimbang mengharuskan *preprocessing* data yang berfungsi untuk normalisasi data citra. Tujuan dari tahap ini adalah untuk memperbaiki kualitas dan menyeragamkan *dataset* agar memaksimalkan proses *data training*. *Preprocessing* dilakukan dengan memanfaatkan ImageDataGenerator untuk memuat, mempersiapkan, hingga mengubah data citra sebelum dilakukan pemodelan. Augmentasi data dengan ImageDataGenerator berfungsi untuk transformasi gambar seperti memperbesar atau memperkecil, merotasi, memotong, mencerahkan, dan lain-lain. Proses ini membantu untuk mencegah terjadinya *overfitting* serta meningkatkan kemampuan model dalam mengenali variasi data.

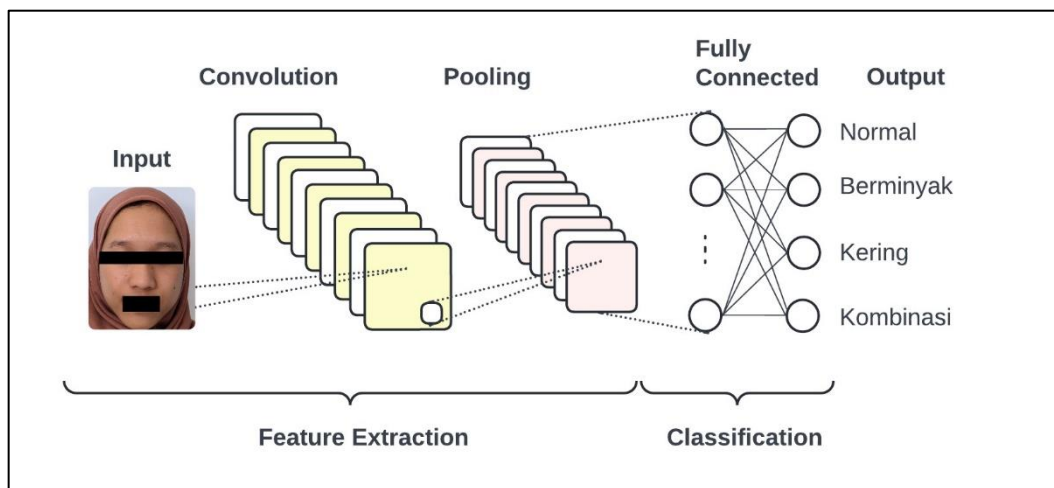
- a. Normalisasi Citra: Proses ini berfungsi untuk normalisasi data citra seperti merubah nilai piksel atau resolusi gambar.
- b. *Batching Data*: Data citra dapat dibagi menjadi kelompok-kelompok kecil dan memuat gambar secara otomatis. Hal ini berfungsi untuk mengurangi penggunaan memori agar dapat mengolah *dataset* yang besar.
- c. Penyesuaian Kelas: Ketika kondisi *dataset* yang dimiliki tidak seimbang, data citra pada setiap kelas dapat diseimbangkan untuk menangani distribusi kelas dengan cara membuat variasi citra dari kelas minoritas.

## 3.4 Training Model

*Data training* merupakan kumpulan dari data yang digunakan untuk melatih model dengan suatu tujuan tertentu dan selanjutnya digunakan untuk mengoptimalkan pengembangan model (Farnham et al., 2019). Proses *training* pada suatu model dilakukan untuk mempelajari ciri-ciri dari setiap data yang disajikan untuk menemukan pola dari setiap

kelompok data. Selain itu, tahap *training* juga dilakukan untuk memberikan *output* yang diharapkan serta mengetahui kemampuan suatu model dalam memberikan hasil prediksi dengan benar. Hasil dari *training data* selanjutnya digunakan untuk melakukan prediksi terhadap data *input* yang baru (Muttaqin et al., 2023).

Alur model CNN untuk penentuan tipe kulit wajah ditunjukkan pada Gambar 3.6. Gambar wajah secara keseluruhan ataupun potongan kulit wajah dijadikan sebagai *input*. Dari proses CNN, akan menghasilkan *output* berupa salah satu tipe kulit wajah di antara normal, berminyak, kering, atau kombinasi.



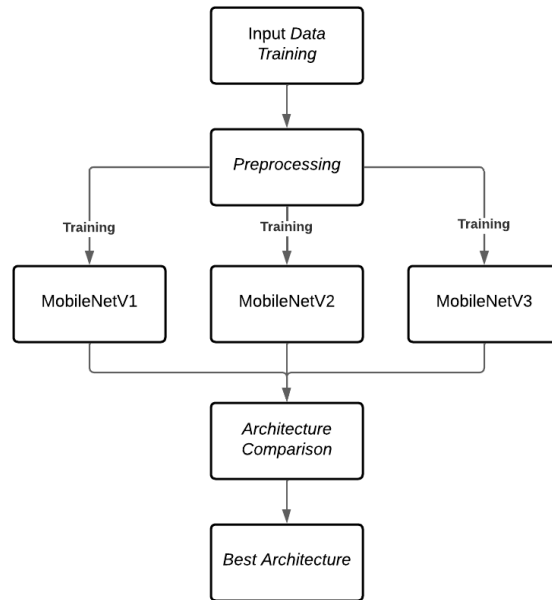
Gambar 3.6 Alur *Convolutional Neural Network* (CNN)

Proses *data training* merupakan tahap untuk mengembangkan model melalui pelatihan data. *Dataset* yang digunakan sebagai *data training* akan dilatih dengan memanfaatkan metode CNN. Tujuan *training data* adalah untuk memperoleh akurasi yang tinggi dan meminimalkan *error* sehingga model yang dikembangkan sesuai dengan tujuan (Suartika E.P et al., 2016). Secara umum, alur *training data* berjalan ditunjukkan pada Gambar 3.7.



Gambar 3.7 Alur *Data Training*

Pada proses *data training*, penelitian ini mencoba menggunakan tiga arsitektur CNN antara lain MobilNetV1, MobilNetV2, dan MobilNetV3. *Input data training* menggunakan *dataset* yang sama untuk setiap arsitekturnya sehingga hasilnya dapat dibandingkan. Gambar 3.8 menunjukkan alur *training data* yang diterapkan pada setiap arsitektur CNN yang akan diimplementasikan dalam penelitian ini.



Gambar 3.8 Alur *Data Training*

Alur *data training* dimulai dari *input data* berupa gambar. Selanjutnya, gambar akan melalui tahap *preprocessing* yang bertujuan untuk mempersiapkan citra sesuai dengan kebutuhan model yang dikembangkan. Beberapa hal yang dilakukan pada tahap *preprocessing* antara lain normalisasi, *resize*, augmentasi data hingga penghapusan *noise*. Selanjutnya, diteruskan dengan proses *data training* yang dilakukan untuk pelatihan terhadap model dengan menggunakan data yang telah disiapkan untuk *training* model. Berbeda dengan alur pemodelan CNN pada umumnya, pada penelitian ini tahap *data training* mencoba satu persatu arsitektur antara lain MobileNetV1, MobileNetV2, dan MobileNetV3. Setelah proses *training*, dilanjutkan dengan pengujian menggunakan *data testing* yang tidak termasuk dalam *data training*. Tujuan dari *training* model adalah untuk mendapatkan pemodelan dengan performa terbaik berdasarkan data yang diberikan. Selanjutnya, hasil dari masing-masing arsitektur dibandingkan nilainya, sehingga didapatkan arsitektur terbaik dan paling sesuai untuk pemodelan pada penelitian ini.

### 3.5 Model Evaluation

*Model evaluation* merupakan tahap penilaian kinerja dari model yang dikembangkan dengan tujuan untuk mengetahui performa model. Performa model dapat dilihat dari beberapa faktor seperti nilai akurasi, presisi, *recall*, hingga *loss function*. Berdasarkan evaluasi model, dapat dilihat arsitektur terbaik dalam pengembangan model klasifikasi tipe kulit wajah. Besaran akurasi yang didapatkan pada pemodelan didapatkan berdasarkan *preprocessing* dan pemodelan yang telah dilakukan. Pemodelan dapat memberikan hasil yang berbeda karena

adanya perubahan dalam *preprocessing*, bahkan perubahan kecil sekalipun, Setiap perubahan data input akan berkontribusi pada hasil pemodelan yang diberikan (Goodfellow et al., 2017). Identifikasi kesalahan model secara spesifik dapat dilihat melalui hasil dari *confusion matrix* seperti pada kondisi apa model tersebut cenderung melakukan kesalahan dalam klasifikasi kelas seperti cenderung mengabaikan kasus positif atau mengelompokkan kelas negatif sebagai positif (Wardani & Leonardi, 2023).

## BAB 4

### Hasil dan Pembahasan

Hasil dan pembahasan berisi pemaparan dari penulis mengenai gambaran proses pemodelan yang dijabarkan secara rinci mulai dari pemrosesan *dataset*, *preprocessing*, data *training*, hingga pengujian model. Metode yang diimplementasikan dalam pemodelan ini adalah *Convolutional Neural Network* (CNN) dengan memberikan perbandingan hasil antara MobileNetV1, MobileNetV2, dan MobileNetV3Large.

#### 4.1 Dataset

*Dataset* yang digunakan dalam penelitian ini telah melalui proses *labelling* sebelumnya. Dikarenakan minimnya *dataset*, pembagian citra berdasarkan jenis kulit telah dilakukan di luar model, sehingga hanya perlu membagi keseluruhan data citra ke dalam folder data *training* dan data *testing*. *Source code* pembagian folder data *training* dan data *testing* ditampilkan pada Gambar 4.1.

```
[ ] import os
import shutil
import random

# Path folder data citra
data_folder = '/content/Data_Citra'

# Persentase data testing (20%)
test_percentage = 20

# Create folder training dan testing
train_folder = 'train_folder'
test_folder = 'test_folder'
os.makedirs(train_folder, exist_ok=True)
os.makedirs(test_folder, exist_ok=True)

# List all the subfolders in the data folder (assuming each subfolder represents a category)
categories = [f.name for f in os.scandir(data_folder) if f.is_dir()]

# Loop through each category and split data into train and test sets
for category in categories:
    category_path = os.path.join(data_folder, category)
    train_category_path = os.path.join(train_folder, category)
    test_category_path = os.path.join(test_folder, category)
    os.makedirs(train_category_path, exist_ok=True)
    os.makedirs(test_category_path, exist_ok=True)

# Get a list of all the image files in the current category folder
image_files = [f.name for f in os.scandir(category_path) if f.is_file()]

# Calculate the number of samples to be used for testing
num_test_samples = int(len(image_files) * test_percentage / 100)

# Randomly shuffle the image files
random.shuffle(image_files)

# Move the first num_test_samples to the test folder, and the rest to the train folder
for i, image_file in enumerate(image_files):
    src_path = os.path.join(category_path, image_file)
    if i < num_test_samples:
        dst_path = os.path.join(test_category_path, image_file)
    else:
        dst_path = os.path.join(train_category_path, image_file)
    shutil.copy(src_path, dst_path)

print("Data splitting into train and test folders is complete.")
```

Gambar 4.1 *Source code* Pembagian Data

Selanjutnya, *dataset* yang telah dibagi ke dalam folder *data training* dengan persentase 80% menghasilkan 472 data citra serta untuk *data testing* dengan persentase 20% menghasilkan 116 data citra. Perincian pembagian data citra menjadi *data training* dan *data testing* telah ditunjukkan pada Tabel 3.3.

## 4.2 Preprocessing

Langkah awal pada proses *data training* adalah melakukan *preprocessing*. Setelah sebelumnya sudah dilakukan *labelling* dan *prepared data* dilanjutkan dengan mengatur nilai-nilai transformasi pada `ImageDataGenerator` untuk proses augmentasi data. `ImageDataGenerator` merupakan alat untuk *preprocessing* gambar dalam pembelajaran mesin yang berfungsi untuk menghasilkan variasi gambar dari satu set gambar dengan tujuan untuk meningkatkan keragaman data dan menghindari terjadinya *overfitting*. `ImageDataGenerator` bertujuan untuk meningkatkan kemampuan model dalam mengenali gambar dalam berbagai posisi, jarak, dan pandangan lain serta meningkatkan variasi gambar. *Source code* penggunaan `ImageDataGenerator` dalam *preprocessing* pemodelan ini ditampilkan pada Gambar 4.2.

```
[ ] train_datagen = ImageDataGenerator(  
    rescale=1./255,  
    rotation_range=40,  
    height_shift_range=0.2,  
    width_shift_range=0.2,  
    horizontal_flip=True,  
    vertical_flip=True,  
    zoom_range=0.2,  
    shear_range=0.2,  
    fill_mode='nearest')  
  
validation_datagen = ImageDataGenerator(  
    rescale=1./255)
```

Gambar 4.2 *Source code* Augmentasi Data 1

Beberapa transformasi beserta dengan parameter yang diterapkan dalam pemodelan ini antara lain:

- a. *rescale* merupakan skala perubahan yang diterapkan pada piksel gambar yang berfungsi untuk mengubah skala intensitas piksel agar lebih mudah diproses pada pemodelan. Dalam pemodelan ini, parameter yang digunakan sebesar 1./255 yang artinya setiap piksel dalam gambar akan dibagi 255 yang membuat rentang piksel dalam gambar berada dalam rentang 0 sampai 1.

- b. *rotation\_range* berfungsi untuk menentukan sudut rotasi gambar. Pada pemodelan ini, setiap gambar akan mengalami rotasi sebesar 40 derajat searah jarum jam dan berlawanan dengan jarum jam.
- c. *height\_shift\_range* dan *width\_shift\_range* merupakan transformasi untuk mengontrol jarak pergeseran gambar secara vertikal maupun horizontal. Dalam pemodelan ini, jarak pergeseran vertikal dan horizontal sebesar 0,2 atau 20% dari gambar asli.
- d. *horizontal\_flip* dan *vertical\_flip* untuk menentukan pemantulan gambar secara horizontal atau vertikal dengan parameter boolean dengan probabilitas sebesar 50%.
- e. *zoom\_range* merupakan transformasi untuk penyusutan dan perbesaran gambar. Dalam pemodelan ini, parameter yang digunakan sebesar 0,2 atau 20% untuk penyusutan dan perbesaran dari gambar asli.
- f. *shear\_range* berfungsi untuk mengontrol tingkat pergeseran atau distorsi gambar.
- g. *fill\_mode* berfungsi untuk mengisi bagian-bagian kosong dalam gambar setelah proses transformasi gambar dengan parameter '*nearest*' yang berarti mengisi bagian kosong tersebut dengan piksel terdekatnya.

Selanjutnya, menyiapkan augmentasi data yang digunakan untuk melatih data citra dan validasi dengan memanfaatkan kelas ImageDataGenerator. Fungsi dari generator ini adalah untuk augmentasi data citra yang digunakan selama pelatihan serta proses validasi pada model yang dikembangkan. *Source code* augmentasi pada generator data latih dan generator validasi dalam pemodelan ini ditampilkan pada Gambar 4.3.

```

train_generator = train_datagen.flow_from_directory(
    train_folder,
    target_size=(224,224),
    class_mode='categorical',
    batch_size=32,
    shuffle=True)

validation_generator = validation_datagen.flow_from_directory(
    test_folder,
    target_size=(224,224),
    batch_size=32,
    class_mode='categorical',
    shuffle=True)

```

Gambar 4.3 *Source code* Augmentasi Data 2

Pada generator data latih yang digunakan dalam augmentasi data, berikut detail penjelasan baris kode di atas:

- a. *train\_datagen* merupakan salah satu *instance* dari *ImageDataGenerator* yang berfungsi untuk augmentasi data. Sebelumnya, telah dilakukan transformasi yang bertujuan untuk transformasi citra dengan cara *resize*, *zoom range*, *shear range*, *horizontal flipping*, dan sebagainya.
- b. *Flow\_from\_directory* merupakan *instance* yang mampu menghasilkan bagian-bagian dari data yang telah melewati proses augmentasi.
- c. *Train\_folder* merupakan *path* untuk menuju direktori maupun subdirektori yang berisi data citra yang digunakan dalam pelatihan.
- d. *Target\_size* merupakan sebesar apa ukuran dari data citra akan dilakukan *resize*.
- e. *Class\_mode* merupakan parameter yang digunakan dalam menentukan jenis label yang digunakan. Dalam model ini menggunakan mode *categorical* untuk klasifikasi lebih dari dua kelas.
- f. *Shuffle* merupakan parameter untuk menentukan apakah sebelum dibentuk menjadi bagian tertentu, data dalam direktori akan diacak ataupun tidak. Dalam model ini, parameter *shuffle* berisi nilai *true* yang berarti data citra akan diacak setiap generator mencapai akhir atau setiap mulai *epoch*.

### 4.3 Membangun Model

Setelah model melalui proses augmentasi data, selanjutnya dilakukan proses untuk membangun model. Model CNN yang dimanfaatkan dalam pemodelan ini menggunakan tipe *sequential* yang artinya, setiap lapisan ditambahkan satu per satu. *Source code* pembangunan model ditampilkan pada Gambar 4.4.

```

model = tf.keras.models.Sequential([
    tf.keras.layers.Conv2D(32, (3,3), activation='relu', input_shape=(200, 200, 3)),
    tf.keras.layers.MaxPooling2D(2, 2),
    tf.keras.layers.Conv2D(32, (3,3), activation='relu'),
    tf.keras.layers.MaxPooling2D(2,2),
    tf.keras.layers.Conv2D(64, (3,3), activation='relu'),
    tf.keras.layers.Flatten(),
    tf.keras.layers.Dropout(0.5),,
    tf.keras.layers.Dense(512, activation='relu'),
    tf.keras.layers.Dense(1, activation='sigmoid')
])
model.summary()

```

Gambar 4.4 *Source code* Model

Berikut penjelasan setiap lapisan pada proses membangun model CNN yang dilakukan:

- a. *Conv2D Layer* untuk lapisan konvolusi pertama dengan menggunakan 32 filter ukuran kernel (3,3) serta fungsi aktivasi ReLU. Nilai *input\_shape* menunjukkan

bahwa data citra memiliki ukuran dimensi 200x200 piksel dan 3 saluran warna yaitu RGB.

- b. *MaxPooling2D* Layer merupakan lapisan *pooling* pertama yang menggunakan ukuran (2,2) pada operasi *max pooling* untuk mengurangi dimensi spasial dari citra.
- c. *Conv2D Layer* untuk lapisan konvolusi kedua menggunakan 32 filter, ukuran kernel (3,3) dan menggunakan fungsi aktivasi ReLU.
- d. *MaxPooling2D Layer* untuk lapisan kedua dengan ukuran (2,2).
- e. *Flatten Layer* digunakan untuk mengubah *output* menjadi vektor satu dimensi dari lapisan konvolusi.
- f. *Conv2D Layer* untuk lapisan konvolusi ketiga menggunakan 64 filter dan ukuran kernel (3,3) serta fungsi aktivasi ReLU.
- g. *Flatten Layer* berfungsi untuk mengubah lapisan konvolusi menjadi vektor satu dimensi.
- h. *Dropout Layer* berfungsi menerapkan teknik modifikasi untuk membantu mencegah terjadinya *overfitting* secara acak dengan membuang beberapa bagian unit selama pelatihan.
- i. *Dense layer* pada lapisan *fully connected* dengan fungsi aktivasi ReLU dan 512 neuron.
- j. *Dense layer* pada lapisan *output* menggunakan fungsi aktivasi *sigmoid*. Fungsi dari lapisan ini adalah untuk klasifikasi biner, dengan nilai *output* mendekati 0 untuk kelas negatif dan mendekati 1 untuk kelas positif.

*Output* yang dihasilkan dari Gambar 4.4 pada *source code* “model.summary” ditunjukkan pada Gambar 4.5. Hasil tersebut menunjukkan informasi mengenai kompleksitas model, struktur, serta jumlah parameter selama pelatihan. Hal ini membantu verifikasi arsitektur model sebelum melalui tahap selanjutnya.

```

Model: "sequential"
-----
Layer (type)                Output Shape                Param #
-----
conv2d (Conv2D)             (None, 198, 198, 32)       896
max_pooling2d (MaxPooling2D) (None, 99, 99, 32)         0
conv2d_1 (Conv2D)           (None, 97, 97, 32)         9248
max_pooling2d_1 (MaxPooling2D) (None, 48, 48, 32)         0
conv2d_2 (Conv2D)           (None, 46, 46, 64)         18496
flatten (Flatten)           (None, 135424)             0
dropout (Dropout)           (None, 135424)             0
dense (Dense)                (None, 512)                69337600
dense_1 (Dense)              (None, 1)                  513
-----
Total params: 69366753 (264.61 MB)
Trainable params: 69366753 (264.61 MB)
Non-trainable params: 0 (0.00 Byte)

```

Gambar 4.5 Hasil Model

#### 4.4 Training Model

Setelah model dibangun, selanjutnya pelatihan pada model dilakukan dengan menggunakan data yang sebelumnya telah dibagi untuk data training. Pada training model, menggunakan tiga arsitektur yaitu MobileNetV1, MobileNetV2, dan MobileNetV3Large.

Pengembangan menggunakan TensorFlow dan Keras dengan arsitektur model training MobileNet yang telah dilatih. Parameter `weights='imagenet'` menunjukkan bahwa model menggunakan berat yang telah dilatih sebelumnya pada dataset ImageNet. Parameter `include_top` dengan nilai `true` menunjukkan bahwa lapisan output atau lapisan terakhir dari model akan diikutsertakan. Source code pada tahap training model menggunakan MobileNetV1 ditunjukkan pada

Gambar 4.6, MobileNetV2 pada Gambar 4.7 serta MobileNetV3Large pada Gambar 4.8.

- a. MobileNetV1

```

import tensorflow as tf
from tensorflow.keras.applications import MobileNet

# Set seed untuk numpy
np.random.seed(42)

# Set seed untuk tensorflow
tf.random.set_seed(42)

# Load MobileNetV1 with pre-trained weights on ImageNet
model = MobileNet(weights='imagenet', include_top=True)

# Mengompilasi model
model.compile(optimizer='adam', loss='sparse_categorical_crossentropy',
              metrics=['accuracy'])

# Proses pelatihan
history = model.fit(
    train_generator,
    steps_per_epoch=2,
    epochs=10,
    validation_data=validation_generator,
    validation_steps=1,
    verbose=2
)

```

Gambar 4.6 *Training Model MobileNetV1*

b. MobileNetV2

*Source code* pada tahap *training* model menggunakan MobileNetV2 ditunjukkan pada Gambar 4.7.

```

import tensorflow as tf
from tensorflow.keras.applications import MobileNetV2

# Set seed untuk numpy
np.random.seed(42)

# Set seed untuk tensorflow
tf.random.set_seed(42)

# Membuat objek model MobileNetV2
model = MobileNetV2(weights='imagenet', include_top=True)

# Mengompilasi model
model.compile(optimizer='adam', loss='sparse_categorical_crossentropy',
              metrics=['accuracy'])

# Proses pelatihan
history = model.fit(
    train_generator,
    steps_per_epoch=2,
    epochs=10,
    validation_data=validation_generator,
    validation_steps=1,
    verbose=2
)

```

Gambar 4.7 *Training Model MobileNetV2*

c. MobileNetV3

*Source code* pada tahap *training* model menggunakan MobileNetV3Large ditunjukkan pada Gambar 4.8.

```

import tensorflow as tf
from tensorflow.keras.applications import MobileNetV3Large

# Set seed untuk numpy
np.random.seed(42)

# Set seed untuk tensorflow
tf.random.set_seed(42)

# Membuat objek model MobileNetV3
model = MobileNetV3Large(weights='imagenet', include_top=True)

# Mengompilasi model
model.compile(optimizer='adam', loss='sparse_categorical_crossentropy',
              metrics=['accuracy'])

# Proses pelatihan
history = model.fit(
    train_generator,
    steps_per_epoch=2,
    epochs=10,
    validation_data=validation_generator,
    validation_steps=1,
    verbose=2
)

```

Gambar 4.8 *Training Model MobileNetV3Large*

Berikut penjelasan *source code* pada tahap *training* model menggunakan model arsitektur MobileNetV1, MobileNetV2, serta MobileNetV3:

- 1) *Set seed* digunakan untuk mengatur seed acak pada NumPy dan TensorFlow. Tujuan pengaturan nilai *set seed* adalah agar hasil model tetap sama pada setiap iterasi atau konsisten.
- 2) MobileNet/MobileNetV2/MobileNetV3Large digunakan untuk memuat model arsitektur MobileNet dengan menggunakan bobot pada ImageNet yang sebelumnya telah dilatih melalui baris kode *weights='imagenet'*. Sedangkan *include\_top=True* memuat keseluruhan arsitektur pada MobileNet termasuk lapisan paling atas.
- 3) Compile model berfungsi untuk mengompilasi model untuk memperbarui bobot dan cara kinerja model akan diukur selama proses pelatihan dengan memanfaatkan *optimizer='adam'*, *loss='sparse\_categorical\_crossentropy'*, dan *metrics=['accuracy']*.
- 4) *Model.fit* digunakan untuk melatih model menggunakan metode fit. Beberapa parameter yang digunakan antara lain:
  - *train\_generator* yang menghasilkan *batch data* dari direktori secara dinamis selama pelatihan berlangsung.

- *steps\_per\_epoch* digunakan untuk menentukan jumlah *batch* yang akan dilakukan model pada setiap *epoch* pada saat *training*. Nilai yang diberikan pada parameter ini adalah 2 yang berarti model akan mengambil 2 *batch* dari generator pada setiap *epoch*.
- parameter *epoch* diberikan nilai 10 yang berarti jumlah *epoch* atau banyaknya model melatih seluruh *dataset* adalah 10 kali.
- *validation\_data* digunakan selama pelatihan untuk mengevaluasi kinerja dari model pada setiap *epoch*.
- *validation\_steps* merupakan parameter yang berfungsi untuk menentukan jumlah *batch* pada model untuk setiap *epoch* saat evaluasi data validasi. Pada model ini diberikan nilai 1 yang berarti model akan melangkah sebanyak 1 kali pada evaluasi setiap *epoch*.
- parameter *verbose* digunakan untuk mengontrol banyaknya *output* yang akan ditampilkan selama *training model*. Nilai 2 pada parameter *verbose* berarti *output* yang akan ditampilkan menunjukkan kemajuan pelatihan setiap *epoch* secara ringkas dan detail dengan informasi *loss* dan *metrics*.

## 4.5 Model Evaluation

Tahap evaluasi model bertujuan untuk mengukur hasil kinerja dari model yang telah dikembangkan. Model divalidasi setelah selesai dilatih dengan menggunakan *data testing* yang telah disiapkan. Kinerja model dapat dilihat dari hasil nilai *accuracy* dan *loss* dari model pada setiap arsitektur.

### 4.5.1 Plotting Akurasi dan Loss

*Plotting* akurasi dan *loss* bertujuan untuk melihat apakah model yang dikembangkan memiliki kinerja melalui visualisasinya. *Source code* pada tahap evaluasi model untuk mendapatkan nilai akurasi dan *loss* model dari setiap arsitektur ditunjukkan pada Gambar 4.9. Berikut penjelasan setiap baris kode pada *source code plotting* akurasi dan *loss*:

- parameter *acc*, *val\_acc*, *loss*, dan *val\_loss* berfungsi untuk mengambil nilai akurasi, akurasi validasi, *loss*, dan *loss* validasi dari riwayat pelatihan model pada setiap *epoch*.
- *epoch* berisi nilai yang merepresentasikan jumlah *epoch* yang digunakan pada model dan digunakan sebagai visualisasi sumbu x pada plot.

- `plt.plot` digunakan untuk membuat plot akurasi maupun *loss*. *Epoch* menggambarkan sumbu x sedangkan `acc` menggambarkan sumbu y berdasarkan nilai akurasi atau *loss*.
- `plt.title` digunakan untuk membuat judul pada plot yang ditampilkan.
- `plt.show` digunakan untuk menampilkan plot akurasi maupun *loss* yang telah dibuat.

```

acc = history.history['accuracy']
val_acc = history.history['val_accuracy']
loss = history.history['loss']
val_loss = history.history['val_loss']

epochs = range(len(acc))

#Plot Akurasi
plt.plot(epochs, acc, 'orange', label = 'Akurasi Training')
plt.plot(epochs, val_acc, 'royalblue', label = 'Akurasi Validasi')
plt.title('Akurasi Training dan Validasi')
plt.legend(loc = 0)
plt.figure()
plt.show()

# Plot Loss
plt.plot(epochs, loss, 'orange', label = 'Loss Training')
plt.plot(epochs, val_loss, 'royalblue', label = 'Loss Validasi')
plt.title('Loss Training dan Validasi')
plt.legend(loc = 0)
plt.figure()
plt.show()

```

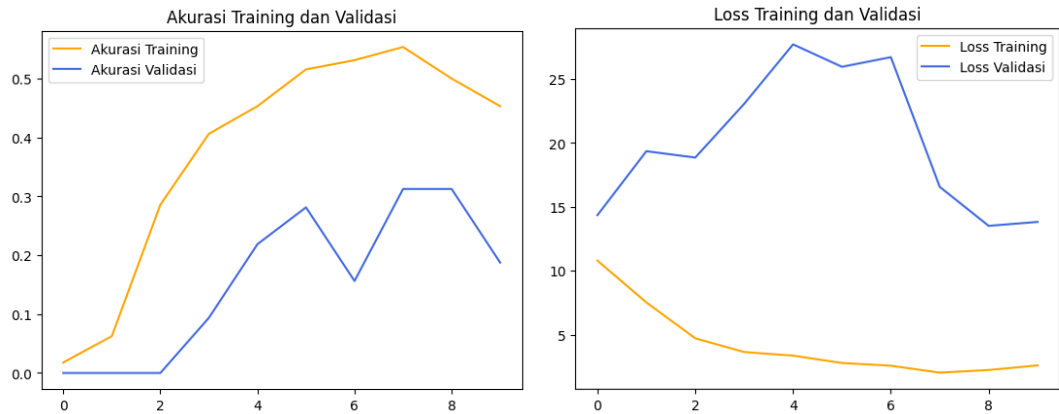
Gambar 4.9 *Plotting* Akurasi dan *Loss*

Hasil *potting* akurasi dan *loss* untuk masing-masing arsitektur model ditunjukkan pada Gambar 4.10 untuk arsitektur MobileNetV1, Gambar 4.11 untuk MobileNetV2, dan Gambar 4.12 untuk MobileNetV3.

a. MobileNet V1

*Training model* menggunakan arsitektur MobileNetV1 memberikan hasil akurasi dan *loss* yang divisualisasikan menggunakan plot yang ditunjukkan pada Gambar 4.10. Plot akurasi *training* dan validasi dengan garis berwarna biru menunjukkan nilai *validation accuracy* sebesar 18% dan garis berwarna jingga menunjukkan hasil nilai *training accuracy* sebesar 45%. Sedangkan plot *loss training* dan validasi dengan *validation loss* sebesar 13.8281 yang digambarkan dengan garis biru, sedangkan garis jingga menunjukkan nilai *training loss* sebesar 2.6019. Nilai yang dihasilkan pada pengembangan model menggunakan arsitektur MobileNetV1 cukup rendah. Kesimpulannya, *training model* dengan dengan arsitektur MobilNetV1 dapat dikatakan kurang baik karena masih banyak

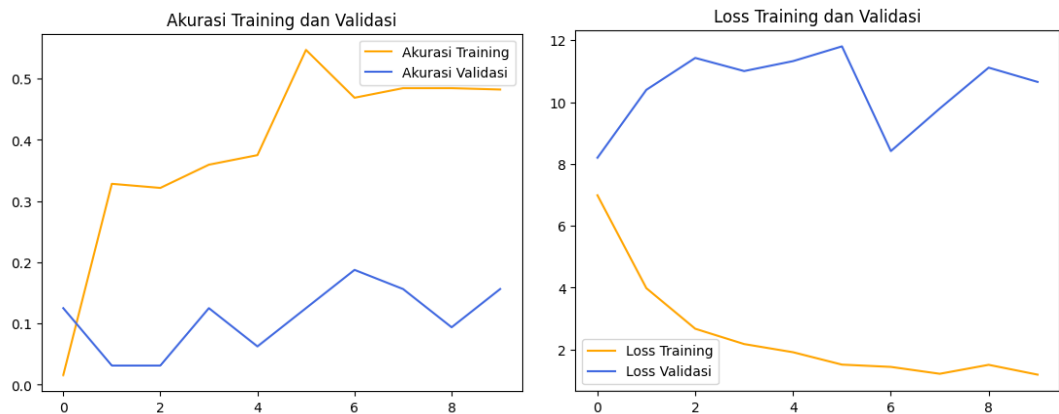
kekurangan salah satunya dari sisi akurasi yang masih rendah dan nilai *loss* yang tinggi.



Gambar 4.10 Hasil Akurasi dan *Loss* MobileNetV1

b. MobileNet V2

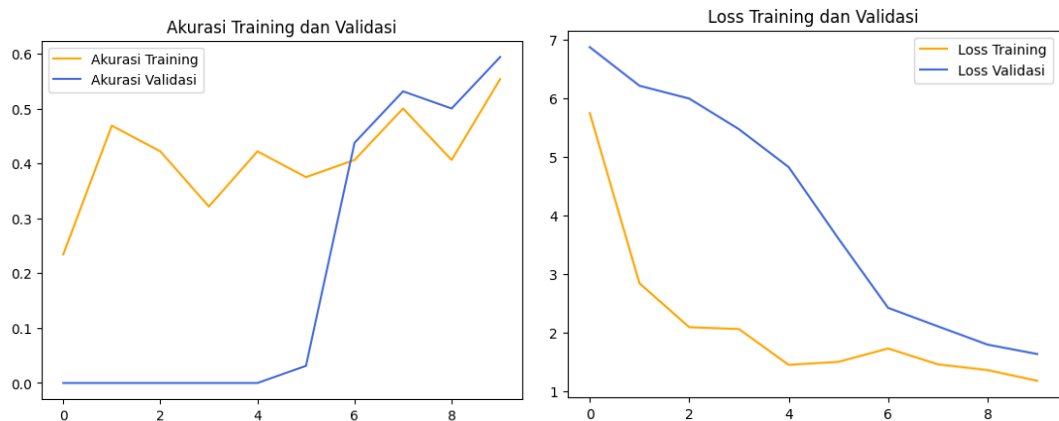
Berdasarkan hasil *training model* menggunakan arsitektur MobileNetV2 memberikan hasil akurasi dan *loss* yang divisualisasikan menggunakan plot yang ditunjukkan pada Gambar 4.11. Plot akurasi *training* dan validasi dengan garis berwarna biru menunjukkan nilai *validation accuracy* sebesar 15% dan garis berwarna jingga menunjukkan hasil nilai *training accuracy* sebesar 48%. Sedangkan plot *loss training* dan validasi dengan *validation loss* sebesar 10.6518 yang digambarkan dengan garis biru, sedangkan garis jingga menunjukkan nilai *training loss* sebesar 1.1890. Nilai yang dihasilkan pada pengembangan model menggunakan arsitektur MobileNetV2 masih rendah. Kesimpulannya, *training model* dengan arsitektur MobilNetV2 lebih baik dari MobileNetV1 tetapi masih dalam kategori kurang baik karena dari sisi akurasi masih rendah dan *loss* masih cukup tinggi.



Gambar 4.11 Hasil Akurasi dan *Loss* MobileNetV2

c. MobileNet V3

*Training model* menggunakan arsitektur MobileNetV3 memberikan hasil akurasi dan *loss* yang divisualisasikan menggunakan plot yang ditunjukkan pada Gambar 4.12. Plot akurasi *training* dan validasi dengan garis berwarna biru menunjukkan nilai *validation accuracy* sebesar 46% dan garis berwarna jingga menunjukkan hasil nilai *training accuracy* sebesar 40%. Sedangkan plot *loss training* dan validasi dengan *validation loss* sebesar 2.2750 yang digambarkan dengan garis biru, sedangkan garis jingga menunjukkan nilai *training loss* sebesar 1.3513. Nilai yang dihasilkan pada pengembangan model menggunakan arsitektur MobileNetV2 masih rendah. Kesimpulannya, *training model* dengan dengan arsitektur MobilNetV3Large lebih baik daripada kedua arsitektur sebelumnya karena akurasi *training* dan *validation* cukup seimbang dan nilai *loss* cukup rendah walaupun secara keseluruhan masih kurang baik sebagai model.



Gambar 4.12 Hasil Akurasi dan *Loss* MobileNetV3Large

#### 4.5.2 Perbandingan *Accuracy* dan *Loss*

Perbandingan nilai akurasi dan *loss* pada setiap arsitektur yang dicoba pada model yang dikembangkan ditunjukkan pada Tabel 4.1.

Tabel 4.1 Perbandingan Validasi *Accuracy* dan *Loss*

<i>Architecture</i>	<i>Batch</i>	<i>Validation</i>	
		<i>Accuracy</i>	<i>Loss</i>
MobilNetV1	Size 116	18%	13.8281
MobilNetV2	Size 116	15%	10.651
MobilNetV3	Size 116	46%	2.2750

#### 4.5.3 *Confusion Matrix*

*Confusion matrix* bertujuan untuk menunjukkan seberapa baik sebuah model dalam memprediksi data serta mengidentifikasi kesalahan. Tahap membuat dan menampilkan

*confusion matrix* dari model dengan menggunakan *source code* yang ditunjukkan pada Gambar 4.13. Penjelasan *source code* yang digunakan antara lain:

- Dimulai dengan impor fungsi *confusion\_matrix* dari *sklearn.metrics* yang digunakan untuk mengukur kinerja model.
- Fungsi *y\_test = validation\_generator.classes* untuk menyimpan label aktual kelas dari dataset validasi.
- Fungsi *y\_pred = model.predict(validation\_generator)* digunakan untuk menyimpan hasil prediksi dengan menggunakan model yang sebelumnya telah dilatih untuk memprediksi kelas dari dataset validasi.
- *y\_pred\_class = np.argmax(y\_pred, axis=1)* berfungsi untuk mengubah hasil prediksi yang sebelumnya berupa probabilitas menjadi label kelas dari setiap baris hasil prediksi diambil nilai tertingginya.
- *cm = confusion\_matrix(y\_test, y\_pred\_class)* digunakan untuk membuat *confusion matrix* dengan *y\_test* sebagai label kelas aktual dan *y\_pred\_class* sebagai label kelas hasil prediksi.
- Fungsi *f, ax = plt.subplots(figsize=(6,5))* untuk membuat plot *heatmap* dengan figur dan *axis* sesuai dengan ukuran yang ditentukan.
- *sns.heatmap(cm, annot=True, linewidths=0.01, cmap="Blues", linecolor="gray", fmt='.0f', ax=ax)* digunakan untuk membuat *heatmap* dengan parameter nilai *linewidths=001* untuk menetapkan ukuran lebar garis sel. Selanjutnya, *cmap="blues"* untuk menentukan warna *heatmap* dan *linecolor="gray"* untuk warna garis sel. Parameter *fmt='.0f'* untuk menentukan format nilai dan *ax=ax* untuk menentukan *subplot* yang digunakan.
- *plt.xlabel("Label Prediksi")* untuk memberikan label pada sumbu x.
- *plt.ylabel("Label Aktual")* untuk memberikan label pada sumbu y.
- *plt.title("Confusion Matrix")* untuk memberikan judul pada *heatmap*.
- *plt.show()* untuk menampilkan hasil dari *heatmap confusion matrix*.

```

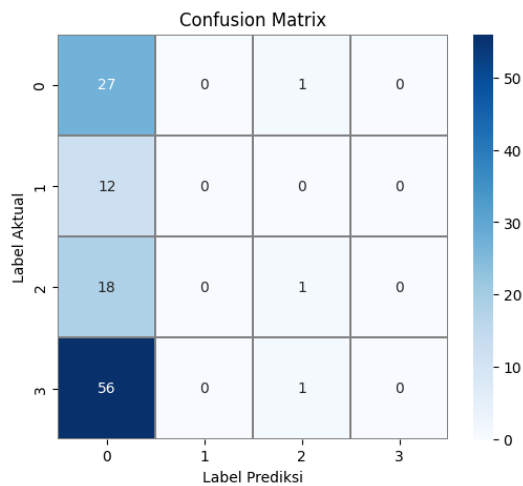
from sklearn.metrics import confusion_matrix
y_test = validation_generator.classes
y_pred = model.predict(validation_generator)
y_pred_class = np.argmax(y_pred, axis=1)
cm = confusion_matrix(y_test, y_pred_class)

f, ax = plt.subplots(figsize=(6, 5))
sns.heatmap(cm,
            annot=True,
            linewidths=0.01,
            cmap="Blues",
            linecolor="gray",
            fmt='.0f',
            ax=ax)
plt.xlabel("Label Prediksi")
plt.ylabel("Label Aktual")
plt.title("Confusion Matrix")
plt.show()

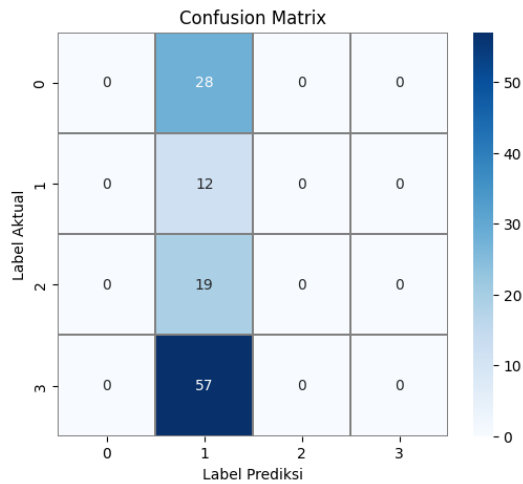
```

Gambar 4.13 *Source Code Confusion Matrix*

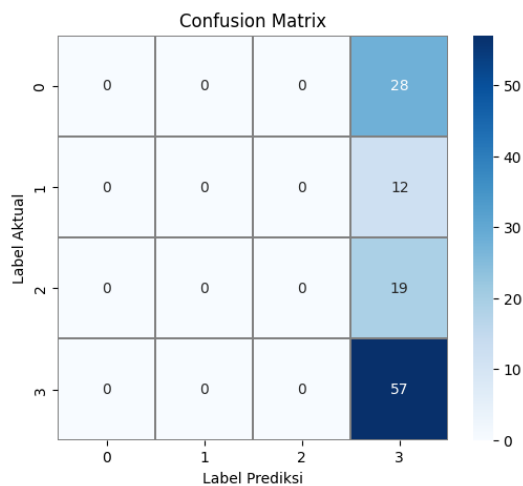
Hasil *heatmap confusion matrix* untuk masing-masing arsitektur model ditunjukkan pada Gambar 4.14 untuk arsitektur MobileNetV1, Gambar 4.15 untuk MobileNetV2, dan Gambar 4.16 untuk MobileNetV3.



Gambar 4.14 *Confusion Matrix MobileNetV1*



Gambar 4.15 *Confusion Matrix* MobileNetV2



Gambar 4.16 *Confusion Matrix* MobileNetV3Large

Hasil *confusion matrix* pada setiap arsitektur menunjukkan adanya kekurangan dalam pemodelan yang telah dilakukan. Kinerja model baik dari sisi *accuracy*, *precision*, dan *recall* menunjukkan nilai yang kurang memuaskan sehingga *confusion matrix* menampilkan hasil yang kurang baik.

#### 4.5.4 *Classification Report*

*Classification report* menunjukkan rangkuman hasil klasifikasi yang dilakukan. Laporan klasifikasi mencakup nilai presisi, recall, dan nilai F1. *Source code* ditunjukkan pada Gambar 4.17. Pada saat fungsi *classification\_report* dipanggil Bersama dengan label *actual y\_test* dan label prediksi *y\_pred\_class*, *classification report* akan menghitung metrik evaluasi pada setiap kelas.

```
from sklearn.metrics import classification_report
report = classification_report(y_test, y_pred_class)
print(report)
```

Gambar 4.17 *Source Code* Hasil Model

Hasil dari *classification report* yang dicoba pada setiap arsitektur model ditunjukkan pada Tabel 4.2. Nilai *precision*, *recall*, dan nilai F1 yang dihasilkan dari setiap arsitektur dibandingkan dalam sebuah tabel. Berdasarkan hasil tersebut, dapat dilihat bahwa nilai *precision* yang menunjukkan besaran presisi dan akurasi dari model dalam prediksi terbaik ditunjukkan pada arsitektur MobileNetV3Large pada kelas ke 3, sedangkan kelas lainnya *precision* bernilai 0. Begitu juga dengan arsitektur MobileNetV1 maupun MobileNetV2, nilai *precision* menunjukkan hasil yang kurang merata dikarenakan tidak seimbangannya antar kelas pada setiap tipe kulit wajah. Seperti yang telah dijelaskan sebelumnya bahwa kelas yang memiliki data citra lebih minim menjadi salah satu sumber penurunan kinerja pada saat penerapan model (De Angeli et al., 2022). Nilai *recall* yang menunjukkan *class* yang ditangkap oleh model arsitektur memberikan hasil yang kurang memuaskan karena tidak semua kelas pada setiap arsitektur mampu mengingat kembali sampel data yang digunakan, hanya beberapa kelas pada setiap arsitektur yang mampu memberikan nilai yang baik. Nilai *accuracy* yang dihasilkan pada setiap arsitektur menunjukkan seberapa baik model dalam melakukan prediksi citra menggunakan model. *Accuracy* terbaik ditunjukkan pada arsitektur MobileNetV3Large dengan nilai sebesar 0.49 atau dapat melakukan prediksi klasifikasi secara benar hampir setengah sampel yang digunakan sebagai data uji.

Tabel 4.2 Perbandingan *Precision*, *Recall*, dan *F1-Score*

<b>Architecture</b>	<b>Batch</b>	<b>Class</b>	<b>Precision</b>	<b>Recall</b>	<b>F1-Score</b>
MobileNetV1	Size 116	0	0.24	0.96	0.38
	Size 116	1	0.00	0.00	0.00
	Size 116	2	0.34	0.05	0.09
	Size 116	3	0.00	0.00	0.00
		<i>accuracy</i>			0.24
		<i>macro avg</i>	0.14	0.25	0.12
		<i>weighted avg</i>	0.11	0.24	0.11
MobileNetV2	Size 116	0	0.00	0.00	0.00
	Size 116	1	0.10	1.00	0.19
	Size 116	2	0.00	0.00	0.00
	Size 116	3	0.00	0.00	0.00
		<i>accuracy</i>			0.10
		<i>macro avg</i>	0.03	0.25	0.05
		<i>weighted avg</i>	0.01	0.10	0.02

MobileNetV3Large	Size 116	0	0.00	0.00	0.00
	Size 116	1	0.00	0.00	0.00
	Size 116	2	0.00	0.00	0.00
	Size 116	3	0.49	1.00	0.66
		<i>accuracy</i>			0.49
		<i>macro avg</i>	0.12	0.25	0.16
		<i>weighted avg</i>	0.24	0.49	0.32

#### 4.5.5 Hasil Evaluasi

Evaluasi model menunjukkan seberapa baik suatu model dalam melakukan klasifikasi sesuai dengan tujuan pengembangan model. Berdasarkan hasil evaluasi yang telah dilakukan, beberapa poin penting yang didapatkan antara lain:

- Jumlah citra pada setiap tipe kulit wajah dinilai tidak seimbang antar kelasnya, hal ini berakibat pada hasil *accuracy*, *precision* dan *recall* yang kurang maksimal untuk setiap kelasnya.
- Terdapat citra yang memiliki kualitas citra yang kurang baik terutama untuk citra yang diambil per *patch* dikarenakan *patch* tersebut diambil secara langsung memotong citra penuh dari wajah lalu dilakukan pemotongan gambar sesuai dengan titik yang dibutuhkan.
- Ciri-ciri atau kemiripan antar tipe kulit wajah kurang bisa terlihat, hal ini dikarenakan minimnya data citra yang digunakan untuk pemodelan sehingga masih memerlukan banyak perbandingan untuk setiap kelasnya.
- Terdapat citra yang memiliki pencahayaan kurang merata atau tidak seimbang antar gambar, walaupun sudah dilakukan pengambilan dengan kamera, sudut, maupun lokasi yang sama. Hal ini disebabkan oleh adanya perbedaan pencahayaan dikarenakan waktu pengambilan yang berbeda (pagi, siang, sore, malam) sehingga memengaruhi hasil citra yang diambil. Untuk mengetahui detail pencahayaan pada setiap gambar dengan waktu pengambilan yang berbeda, perlu dilakukan plot pencahayaan untuk setiap kelompok waktu.

#### 4.6 Plot Pencahayaan

Berdasarkan hasil evaluasi model poin mengenai data citra memiliki cahaya kurang merata, perlu dilakukan plot pencahayaan menggunakan nilai RGB. Tujuan plot pencahayaan ini

adalah untuk mengetahui nilai RGB citra pada setiap kelompok waktu. Kelompok waktu dibagi menjadi empat, yaitu pagi, siang, sore, dan malam. Detail pembagian waktu dapat dilihat pada Tabel 4.3.

Tabel 4.3 Pembagian Waktu Plot Pencahayaan

No	Masa dalam hari	Waktu
1	Pagi	09.00 – 11.00 WIB
2	Siang	11.01 – 15.00 WIB
3	Sore	15.01 – 17.00 WIB
4	Malam	17.01 – 19.00 WIB

*Source code* plot pencahayaan ditampilkan pada Gambar 4.18, Gambar 4.19, dan Gambar 4.20. Dalam *source code*, proses plot pencahayaan dimulai dengan menghitung rata-rata warna RGB. Selanjutnya, *file* gambar pada setiap folder waktu (pagi, siang, sore, malam) didapatkan. Setelah itu, plot pencahayaan dilakukan dengan memvisualisasikan rata-rata warna dari setiap gambar serta folder waktu. Terakhir, dari plot pencahayaan ditampilkan nilai rata-rata warnanya.

```
# Menghitung rata-rata warna untuk setiap kanal
def calculate_average_color(image):
    red_channel_mean = np.mean(image[:, :, 0])
    green_channel_mean = np.mean(image[:, :, 1])
    blue_channel_mean = np.mean(image[:, :, 2])
    return red_channel_mean, green_channel_mean, blue_channel_mean

# Mendapatkan daftar file gambar dalam folder
def process_images_in_folder(folder_path):
    image_files = glob.glob(os.path.join(folder_path, '*.jpg'))
    data = []

    # Load dan hitung rata-rata warna untuk setiap gambar dalam folder
    for image_file in image_files:
        image = imread(image_file)
        average_color = calculate_average_color(image)
        data.append(average_color)

    return data
```

Gambar 4.18 *Source Code* Plot Pencahayaan 1

```
# Mendapatkan daftar folder waktu ('pagi', 'siang', 'sore', 'malam')
time_folders = ['Pagi', 'Siang', 'Sore', 'Malam']

# Membuat DataFrame kosong untuk menyimpan hasil
df = pd.DataFrame(columns=['Red', 'Green', 'Blue'])

# Load gambar di setiap folder waktu
for time_folder in time_folders:
    time_folder_path = os.path.join('/content/Data_Citra_mod_crop_waktu3', time_folder)
    data = process_images_in_folder(time_folder_path)
    df_temp = pd.DataFrame(data, columns=['Red', 'Green', 'Blue'])
    df_temp['Folder'] = time_folder
    df = pd.concat([df, df_temp], ignore_index=True)
```

Gambar 4.19 *Source Code* Plot Pencahayaan 2

```

# Plot rata-rata cahaya warna per gambar pada masing-masing folder waktu untuk setiap warna (merah, hijau, biru)
plt.figure(figsize=(15, 10))

for i, time_folder in enumerate(time_folders, 1):
    df_time_folder = df[df['Folder'] == time_folder]

    plt.subplot(2, 2, i)
    plt.plot(df_time_folder.index, df_time_folder['Red'], label=f'{time_folder} - Red', color='red')
    plt.plot(df_time_folder.index, df_time_folder['Green'], label=f'{time_folder} - Green', color='green')
    plt.plot(df_time_folder.index, df_time_folder['Blue'], label=f'{time_folder} - Blue', color='blue')
    plt.xlabel('Image Index')
    plt.ylabel('Average Color Value')
    plt.title(f'Average Color Value per Image by Time of Day - {time_folder}')
    plt.legend()

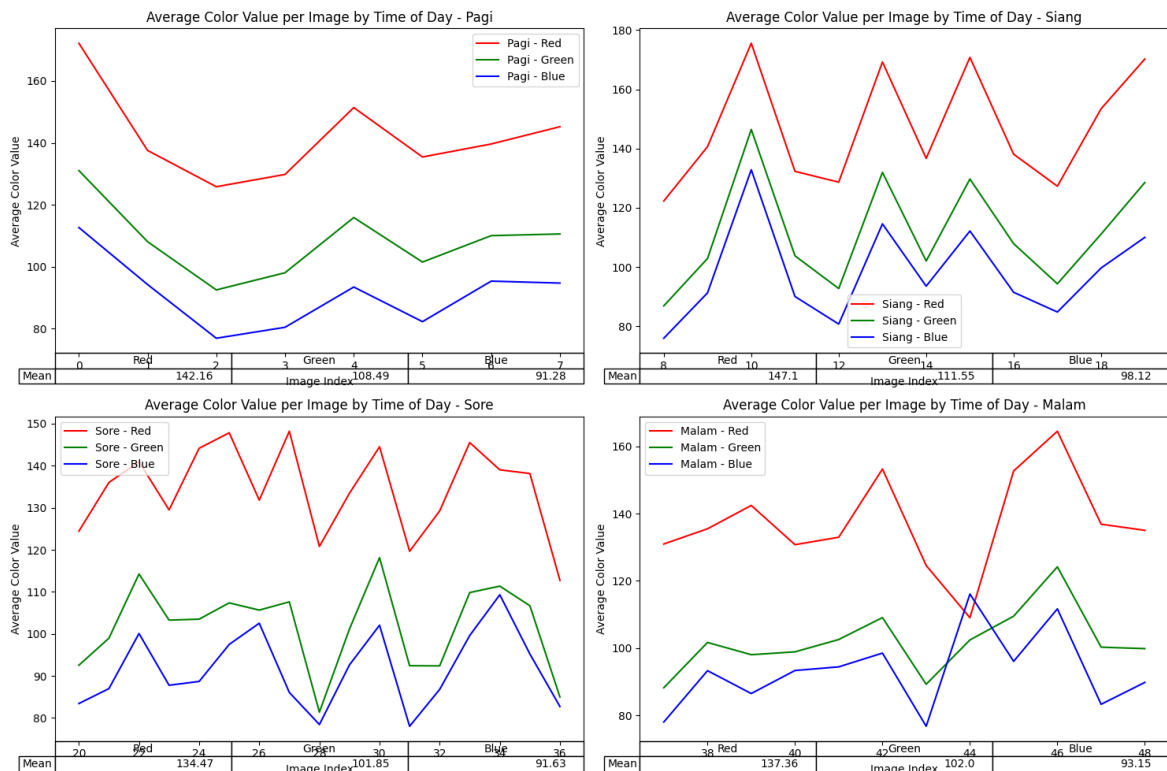
    # Menampilkan tabel numerik
    ax = plt.gca()
    ax.table(cellText=df_time_folder.describe().loc[['mean']].values.round(2),
            rowLabels=['Mean'],
            colLabels=df_time_folder.describe().columns,
            loc='bottom')
    ax.axis('tight')

plt.tight_layout()
plt.show()

```

Gambar 4.20 Source Code Plot Pencahayaan 3

Hasil dari plot pencahayaan sesuai dengan *source code* sebelumnya ditunjukkan pada Gambar 4.21. Rata-rata pencahayaan untuk setiap warna RGB dan folder waktu ditunjukkan pada Tabel 4.4.



Gambar 4.21 Hasil Plot Pencahayaan

Selanjutnya, hasil dari plot pencahayaan dilakukan perhitungan menggunakan standar deviasi. Standar deviasi bertujuan untuk mengukur secara statistik mengenai sebaran

data berdasarkan nilai rata-rata yang dapat diukur menggunakan rumus (4.1). Pada perhitungan plot pencahayaan, nilai standar deviasi dianggap sesuai untuk menilai apakah warna pada setiap waktu dapat dinilai merata atau tidak. Semakin tinggi nilai standar deviasi maka semakin bervariasi warna pada kelompok waktu tersebut dan semakin rendah nilai standar deviasi maka semakin minim variasi warna pada kelompok waktu tersebut.

$$\sigma = \sqrt{\frac{\sum(x_i - \mu)^2}{N}} \quad (4.1)$$

Berdasarkan hasil rata-rata warna pada plot pencahayaan yang ditunjukkan pada Gambar 4.21, nilai tersebut selanjutnya dihitung dengan standar deviasi menggunakan *source code* yang ditunjukkan pada Gambar 4.22. Perhitungan standar deviasi berdasarkan nilai rata-rata setiap waktu pada setiap warna.

```
# Data
data = {
  'Pagi': [142.16, 108.49, 91.28],
  'Siang': [147.10, 111.55, 98.12],
  'Sore': [134.47, 101.85, 91.63],
  'Malam': [137.36, 102.00, 93.15]
}

index = ['Red', 'Green', 'Blue']
df_std = pd.DataFrame(data, index=index)

# Menghitung standar deviasi untuk setiap warna
df_std['Std'] = df_std.std(axis=1)

# Menghitung standar deviasi untuk setiap waktu
std_time = df_std.std()

print("Standar Deviasi Berdasarkan Waktu:")
print(std_time)

print("\nDataFrame Standar Deviasi:")
print(df_std)
```

Gambar 4.22 *Source Code* Standar Deviasi

Hasil dari perhitungan dari plot pencahayaan dan standar deviasi ditunjukkan pada Tabel 4.4. Dari hasil tersebut menunjukkan variasi nilai warna data citra pada setiap kelompok waktu (pagi, siang, sore, malam).

Tabel 4.4 Hasil Plot Pencahayaan

Warna	Mean				Nilai Standar Deviasi
	Pagi	Siang	Sore	Malam	
Red	142.16	147.10	134.47	137.36	5.547644
Green	108.49	111.55	101.85	102.00	4.838115
Blue	91.28	98.12	91.63	93.15	3.156163

<b>Nilai Standar</b>	25.879939	25.308786	22.374757	23.392222	-
<b>Deviasi</b>					

Berdasarkan hasil yang ditunjukkan, berikut analisis hasil nilai standar deviasi untuk setiap warna:

- a. Berdasarkan warna (*red, green, blue*)
  - Nilai standar deviasi paling tinggi berada pada warna *red* (5.547644), artinya berdasarkan data citra yang digunakan menunjukkan bahwa variasi tertinggi terdapat pada warna merah. Warna yang paling dominan pada data citra merupakan warna merah.
  - Nilai standar deviasi paling rendah berada pada warna *blue* (3.156163), artinya berdasarkan data citra yang digunakan menunjukkan minimnya variasi pada warna biru. Distribusi pencahayaan untuk warna biru lebih seragam atau merata.
- b. Berdasarkan kelompok waktu (pagi, siang, sore, malam)
  - Hasil standar deviasi tertinggi dalam kelompok waktu ditunjukkan pada waktu pagi (25.879939), diikuti siang (25.308786), malam (23.392222), lalu sore (22.374757). Hal ini menunjukkan adanya variasi besar yang dipengaruhi oleh cahaya karena perbedaan waktu pengambilan data.

Berdasarkan hasil dari standar deviasi berdasarkan warna maupun waktu, dapat disimpulkan bahwa adanya variasi dapat disebabkan oleh perbedaan pencahayaan yang diterima objek ataupun kondisi objek yang difoto. Selain faktor waktu pengambilan citra yang mengakibatkan adanya perbedaan cahaya yang diterima, faktor lingkungan ataupun posisi matahari juga memengaruhi citra yang dihasilkan, mengingat pengambilan citra dilakukan di luar ruangan.

#### 4.7 Hubungan Antara Kelembaban dan Tipe Kulit Wajah

Pada tahap pengumpulan data citra, peneliti melakukan pengecekan kadar kelembaban kulit wajah menggunakan alat SK-IV Digital Moisture. Pengecekan dilakukan pada lima titik yaitu dahi, hidung, pipi kanan, pipi kiri, dan dagu. Rata-rata hasil kelembaban setiap tipe kulit wajah ditunjukkan pada Tabel 4.5.

Tabel 4.5 Rerata Kelembaban Kulit

<b>Tipe Kulit</b>	<b>Moisture Dahi</b>	<b>Moisture Hidung</b>	<b>Moisture Pipi Kanan</b>	<b>Moisture Pipi Kiri</b>	<b>Moisture Daggu</b>	<b>Rerata</b>
<b>Berminyak</b>	43,85	48,04	33,49	34,90	37,95	39,65
<b>Kering</b>	53,18	46,13	33,02	35,06	36,14	40,71

Tipe Kulit	Moisture Dahi	Moisture Hidung	Moisture Pipi Kanan	Moisture Pipi Kiri	Moisture Dagu	Rerata
Kombinasi	53,94	43,18	36,26	38,54	34,10	41,20
Normal	53,60	44,31	36,21	37,16	41,05	42,47

Berdasarkan nilai kelembaban yang disajikan pada Tabel 4.5 untuk setiap tipe kulit, rerata nilai kelembaban tertinggi ditunjukkan pada tipe kulit normal. Namun, jika dilihat nilai kelembaban pada setiap titik tidak selalu tertinggi. Selain itu, banyaknya sampel pada setiap tipe kulit tidak berimbang satu sama lain. Untuk itu, pada penelitian ini, nilai kelembaban dinilai tidak berpengaruh pada klasifikasi tipe kulit wajah.

## 4.8 Pemodelan Setelah Perbaikan Citra

Setelah melihat hasil plot pencahayaan citra yang sudah dijelaskan pada 4.6, menunjukkan adanya kekurangan dalam data citra yang digunakan dan perlu dilakukan perbaikan. Untuk itu, perlu dilakukan pemodelan ulang dengan menggunakan citra yang diperbaiki.

### 4.8.1 Perbaikan Citra

*Source code* pada Gambar 4.23 digunakan untuk memproses gambar berdasarkan citra *input* dengan menerapkan koreksi gamma, penyesuaian warna dengan LAB, serta normalisasi histogram. Beberapa metode perbaikan citra ini digunakan secara bersamaan dengan tujuan mampu memperbaiki citra yang akan digunakan untuk pemodelan agar dapat memberikan hasil yang lebih baik dari sebelumnya.

```
def adjust_gamma(image, gamma=1.0):
    invGamma = 1.0 / gamma
    table = np.array([(i / 255.0) ** invGamma] * 255 for i in np.arange(0, 256)).astype("uint8")
    return cv2.LUT(image, table)

def adjust_lab(image):
    lab = cv2.cvtColor(image, cv2.COLOR_BGR2LAB)
    l, a, b = cv2.split(lab)
    l = cv2.equalizeHist(l)
    lab = cv2.merge((l, a, b))
    corrected_image = cv2.cvtColor(lab, cv2.COLOR_LAB2BGR)
    return corrected_image

def normalize_histogram(image):
    image_yuv = cv2.cvtColor(image, cv2.COLOR_BGR2YUV)
    image_yuv[:, :, 0] = cv2.equalizeHist(image_yuv[:, :, 0])
    normalized_image = cv2.cvtColor(image_yuv, cv2.COLOR_YUV2BGR)
    return normalized_image
```

Gambar 4.23 *Source Code* Perbaikan Citra 1

Koreksi gamma dengan menggunakan *adjust\_gamma* berfungsi untuk menerapkan perbaikan gamma pada citra. Koreksi gamma berfungsi untuk mengubah intensitas piksel sesuai dengan nilai gamma yang diberikan yang bertujuan untuk menyesuaikan kecerahan gambar.

Selanjutnya, citra diperbaiki dengan *adjust\_lab* yang berfungsi untuk mengubah gambar dari ruang warna BGR ke LAB, dan meningkatkan kontras dengan cara menyesuaikan saluran L (*lightness*) dengan *histogram equalization*, kemudian mengubahnya kembali ke ruang warna BGR. Untuk perbaikan citra menggunakan *normalize\_histogram* berfungsi untuk mengubah citra dari ruang warna BGR ke YUV, menyesuaikan saluran Y (*brightness*) dengan *histogram equalization*, dan kemudian mengubahnya kembali ke ruang warna BGR.

```
def process_images(input_folder, output_folder, gamma=1.0):
    if not os.path.exists(output_folder):
        os.makedirs(output_folder)

    for filename in os.listdir(input_folder):
        if filename.endswith(".jpg") or filename.endswith(".png") or filename.endswith(".jpeg"):
            img_path = os.path.join(input_folder, filename)
            image = cv2.imread(img_path)

            if image is not None:
                # Apply gamma correction
                image = adjust_gamma(image, gamma)
                # Apply LAB adjustment
                image = adjust_lab(image)
                # Apply histogram normalization
                image = normalize_histogram(image)
                # Save the processed image
                output_path = os.path.join(output_folder, filename)
                cv2.imwrite(output_path, image)
                print(f"Processed and saved: {filename}")
            else:
                print(f"Failed to read: {filename}")

def process_multiple_folders(input_folders, output_folders, gamma=1.0):
    for input_folder, output_folder in zip(input_folders, output_folders):
        print(f"Processing folder: {input_folder}")
        process_images(input_folder, output_folder, gamma)
```

Gambar 4.24 *Source Code* Perbaikan Citra 2

Selanjutnya, pada *source code* yang ditunjukkan dalam Gambar 4.24, fungsi *process\_images* bertujuan untuk memproses semua citra dari folder *input*. Untuk setiap citra, diterapkan perbaikan citra dengan koreksi gamma, penyesuaian LAB, dan normalisasi histogram lalu menyimpan citra yang telah diproses tersebut ke folder *output*. Setelah itu, fungsi *process\_multiple\_folders* menjalankan beberapa folder *input* secara berurutan dengan memanggil *process\_images* untuk setiap pasangan folder *input* dan *output*.

```

# Daftar folder input dan output
input_folders = [
    '/content/Data_Citra_mod_crop/Berminyak',
    '/content/Data_Citra_mod_crop/Kering',
    '/content/Data_Citra_mod_crop/Kombinasi',
    '/content/Data_Citra_mod_crop/Normal'
]
output_folders = [
    '/content/Citra_Perbaikan/Berminyak',
    '/content/Citra_Perbaikan/Kering',
    '/content/Citra_Perbaikan/Kombinasi',
    '/content/Citra_Perbaikan/Normal'
]

gamma_value = 2.2 # Nilai gamma untuk koreksi

# Memproses gambar dalam beberapa folder
process_multiple_folders(input_folders, output_folders, gamma=gamma_value)

```

Gambar 4.25 *Source Code* Perbaikan Citra 3

*Source code* pada Gambar 4.25 mendefinisikan daftar folder *input* dengan menggunakan *input\_folders*, mendefinisikan folder *output* melalui *output\_folders*, serta memberikan nilai *gamma* untuk koreksi *gamma*. Setelah itu, fungsi *process\_multiple\_folders* dipanggil untuk memproses citra dalam semua folder *input* yang telah ditentukan.

#### 4.8.2 *Dataset*

Data citra yang sudah dilakukan perbaikan selanjutnya digunakan sebagai *dataset*. Sama dengan proses yang dilakukan pada poin 4.1, *dataset* dibagi ke dalam folder yang digunakan untuk *data training* dan *data testing* dengan persentase 80% untuk *data training* menghasilkan sebanyak 472 data citra serta untuk *data testing* dengan persentase 20% menghasilkan sebanyak 116 data citra.

#### 4.8.3 *Preprocessing*

Tahap *preprocessing* dengan augmentasi data ditunjukkan pada Gambar 4.26 menggunakan kelas *ImageDataGenerator* dari pustaka *Keras*. Kelas ini berfungsi untuk melakukan augmentasi data citra selama proses pelatihan menggunakan model *deep learning*. Augmentasi data bertujuan untuk meningkatkan variasi data citra pelatihan dengan membuat versi yang sedikit berbeda dari gambar asli. Variasi data citra yang baru akan membantu memperluas *dataset* pelatihan secara efektif. Augmentasi data dapat meningkatkan kemampuan model untuk mengenali fitur atau sudut pandang yang berbeda, sehingga akan membantu model untuk generalisasi yang lebih baik serta dapat menghindari *overfitting*.

```

from keras.preprocessing.image import ImageDataGenerator

# Menggunakan ImageDataGenerator untuk augmentasi data
datagen = ImageDataGenerator(
    rotation_range=20,
    width_shift_range=0.2,
    height_shift_range=0.2,
    shear_range=0.2,
    zoom_range=0.2,
    horizontal_flip=True,
    fill_mode='nearest'
)

# Memuat data pelatihan dan validasi dengan augmentasi
train_generator = datagen.flow_from_directory(
    train_folder,
    target_size=(150, 150),
    batch_size=32,
    class_mode='categorical'
)

validation_generator = datagen.flow_from_directory(
    test_folder,
    target_size=(150, 150),
    batch_size=32,
    class_mode='categorical'
)

```

Gambar 4.26 *Source Code* Augmentasi Data

#### 4.8.4 *Training Model*

Pelatihan model dengan memanfaatkan arsitektur MobileNetV3Large. Berdasarkan hasil evaluasi model yang telah dilakukan sebelumnya pada 4.5, arsitektur yang dapat memberikan hasil terbaik dalam pemodelan untuk menentukan tipe kulit wajah berdasarkan citra foto adalah MobileNetV3Large, sehingga pada pemodelan setelah perbaikan citra memanfaatkan arsitektur tersebut.

```

# Load MobileNetV3Large
base_model = MobileNetV3Large(input_shape=(224, 224, 3), include_top=False, weights='imagenet')
base_model.trainable = False # Freeze the base model

# Membangun model
model = Sequential([
    Conv2D(32, (3, 3), activation='relu', input_shape=(150, 150, 3)),
    MaxPooling2D((2, 2)),
    Conv2D(64, (3, 3), activation='relu'),
    MaxPooling2D((2, 2)),
    Conv2D(128, (3, 3), activation='relu'),
    MaxPooling2D((2, 2)),
    Flatten(),
    Dense(512, activation='relu'),
    Dropout(0.5),
    Dense(4, activation='softmax')
])

# Compile model
model.compile(optimizer=Adam(learning_rate=0.0001), loss='categorical_crossentropy', metrics=['accuracy'])

# Early stopping
early_stopping = EarlyStopping(monitor='val_loss', patience=10, restore_best_weights=True)

# Train model
history = model.fit(
    train_generator,
    epochs=50,
    validation_data=validation_generator,
    callbacks=[early_stopping]
)

```

Gambar 4.27 *Source Code* Training Model

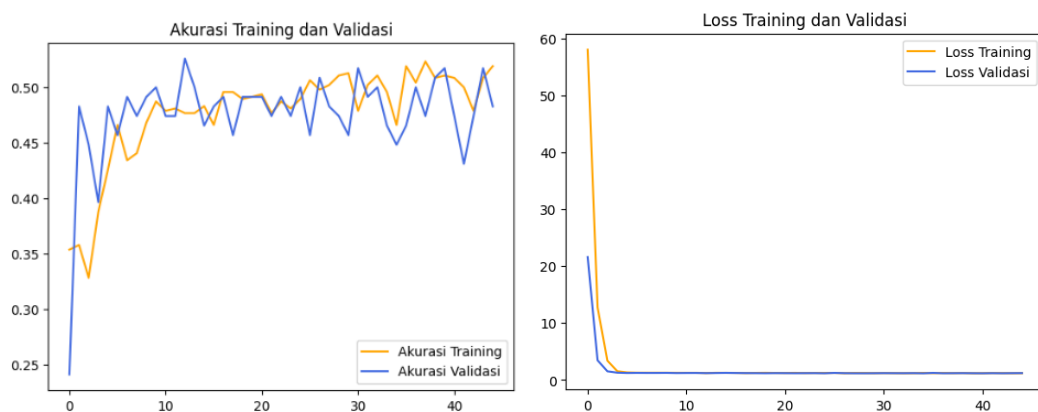
Isi *source code* pada Gambar 4.27 telah dijelaskan sebelumnya pada poin 4.3. Secara keseluruhan, *source code* tersebut digunakan membangun model konvolusi mulai dari mengompilasinya dengan *optimizer* Adam dan fungsi *loss categorical\_crossentropy*, dan melatihnya menggunakan augmentasi data dari *ImageDataGenerator*. Pada model ini diterapkan *callback early stopping* untuk mencegah terjadinya *overfitting* serta mengembalikan bobot terbaik.

#### 4.8.5 Hasil

Pemodelan dengan menggunakan citra setelah perbaikan memberikan *output* berupa *plotting* akurasi dan *loss*, *confusion matrix*, serta *classification report*. Beberapa *output* dari hasil pemodelan ini antara lain:

##### a. *Plotting* Akurasi dan *Loss*

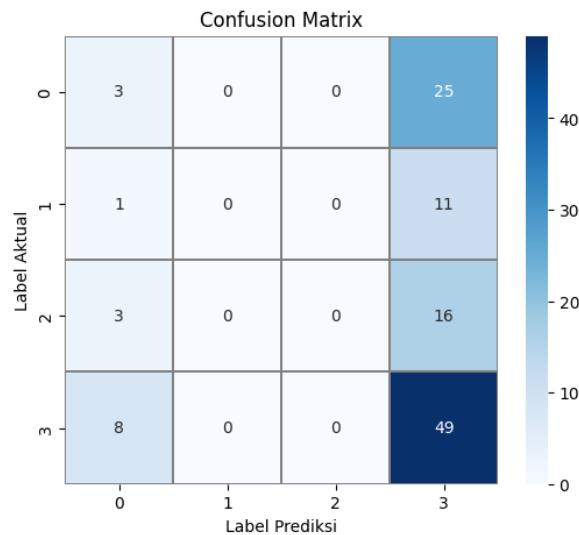
*Training* model menggunakan arsitektur MobileNetV3 dengan citra setelah perbaikan memberikan hasil akurasi dan *loss* yang divisualisasikan menggunakan plot yang ditunjukkan pada Gambar 4.28. Plot akurasi *training* dan validasi dengan garis berwarna biru menunjukkan nilai *validation accuracy* sebesar 48% dan garis berwarna jingga menunjukkan hasil nilai *training accuracy* sebesar 51%. Sedangkan plot *loss training* dan validasi dengan *validation loss* sebesar 1.2150 yang digambarkan dengan garis biru, sedangkan garis jingga menunjukkan nilai *training loss* sebesar 1.2000. Dari hasil *plotting training* model dengan menggunakan citra setelah perbaikan, memberikan hasil yang lebih baik daripada citra sebelum perbaikan. Akurasi *training* dan *validation* cukup seimbang dan nilai *loss* cukup rendah walaupun secara keseluruhan sebagai model masih bisa ditingkatkan.



Gambar 4.28 Hasil Akurasi dan *Loss*

##### b. *Confusion Matrix*

*Confusion matrix* memberikan gambaran tentang hasil kinerja model klasifikasi dengan menampilkan jumlah prediksi yang benar dan yang salah untuk setiap kelas.



Gambar 4.29 Hasil *Confusion Matrix*

Pada *confusion matrix*, baris mewakili kelas sebenarnya (label aktual) dan kolom mewakili kelas yang diprediksi oleh model (label prediksi). Label kelas pada *confusion matrix* adalah sebagai berikut:

- 0: Kelas Berminyak
- 1: Kelas Kering
- 2: Kelas Kombinasi
- 3: Kelas Normal

Tabel 4.6 Hasil *Confusion Matrix*

	Prediksi: 0	Prediksi: 1	Prediksi: 2	Prediksi: 3
Aktual: 0	3	0	0	25
Aktual: 1	1	0	0	11
Aktual: 2	3	0	0	16
Aktual: 3	8	0	0	49

Berdasarkan hasil *confusion matrix* yang ditunjukkan pada Gambar 4.29 dan Tabel 4.6, detail hasil adalah sebagai berikut:

- 1) Baris pertama (Label Aktual 0):
  - 3 contoh diklasifikasikan dengan benar sebagai kelas 0 (*True Positives*)
  - 25 contoh salah diklasifikasikan sebagai kelas 3 (*False Negatives*)
- 2) Baris kedua (Label Aktual 1):
  - 1 contoh salah diklasifikasikan sebagai kelas 0 (*False Negatives*)
  - 11 contoh salah diklasifikasikan sebagai kelas 3 (*False Negatives*)
- 3) Baris ketiga (Label Aktual 2):

- 3 contoh salah diklasifikasikan sebagai kelas 0 (*False Negatives*)
  - 16 contoh salah diklasifikasikan sebagai kelas 3 (*False Negatives*)
- 4) Baris keempat (Label Aktual 3):
- 8 contoh salah diklasifikasikan sebagai kelas 0 (*False Negatives*)
  - 49 contoh diklasifikasikan dengan benar sebagai kelas 3 (*True Positives*)

5) Kesimpulan Hasil dari *Confusion Matrix*

- *True Positives* (Prediksi Benar):
  - Kelas 0 (berminyak) : 3 contoh
  - Kelas 1 (kering) : 0 contoh
  - Kelas 2 (kombinasi) : 0 contoh
  - Kelas 3 (normal) : 49 contoh
- *False Positives* (Prediksi Salah sebagai Kelas):
 

Banyak contoh dari kelas 0, 1, dan 2 diklasifikasikan sebagai kelas 3, yang menunjukkan model sering salah prediksi ke kelas 3.

Selanjutnya, berdasarkan hasil *confusion matrix* dilakukan perhitungan TP, TN, FP, FN, dan IP. Perhitungan dilakukan untuk setiap kelas (0, 1, 2, 3) sebagai berikut:

1) Kelas 0 (berminyak):

- TP : 3
- FP :  $1 + 3 + 8 = 12$
- FN : 25
- TN : Total - (TP + FP + FN) =  $116 - (3 + 12 + 25) = 76$

2) Kelas 1 (kering):

- TP : 0
- FP :  $0 + 0 + 0 = 0$
- FN :  $1 + 11 + 0 = 12$
- TN : Total - (TP + FP + FN) =  $116 - (0 + 0 + 12) = 104$

3) Kelas 2 (kombinasi):

- TP : 0
- FP :  $0 + 0 + 0 = 0$
- FN :  $3 + 0 + 16 = 19$
- TN : Total - (TP + FP + FN) =  $116 - (0 + 0 + 19) = 97$

4) Kelas 3 (normal):

- TP : 49

- FP :  $25 + 11 + 16 = 52$
- FN : 8
- TN : Total - (TP + FP + FN) =  $116 - (49 + 52 + 8) = 7$

Dari keseluruhan hasil perhitungan tersebut, rincian jumlah data ditunjukkan pada Tabel 4.7. Penjelasan enam penilaian yang telah dihitung di atas adalah sebagai berikut:

- *True Positives* (TP): Jumlah prediksi yang benar untuk kelas tertentu
- *True Negatives* (TN): Jumlah prediksi yang benar untuk kelas lain
- *False Positives* (FP): Jumlah prediksi yang salah yang diklasifikasikan sebagai kelas tertentu
- *False Negatives* (FN): Jumlah prediksi yang salah untuk kelas tertentu yang diklasifikasikan sebagai kelas lain
- *Incorrect Predictions* (IP): Jumlah prediksi yang salah (FP + FN)

Tabel 4.7 Hasil TP, TN, FP, FN, dan IP

Kelas	TP	TN	FP	FN	IP
Berminyak	3	76	12	25	37
Kering	0	104	0	12	12
Kombinasi	0	97	0	19	19
Normal	49	7	52	8	60

Berdasarkan nilai *confusion matrix* dan perhitungan TP, TN, FP, FN, dan IP, analisis hasil kinerja model sebagai berikut:

- Model memiliki performa yang sangat baik dalam mengklasifikasikan kelas normal dengan 49 prediksi benar dari total 57 contoh kelas normal (86% akurasi pada kelas normal).
- Model ini memiliki kesulitan besar dalam mengklasifikasikan kelas berminyak, kering, dan kombinasi, dengan sebagian besar contoh salah diklasifikasikan sebagai kelas normal.
- Walaupun kelas berminyak memiliki performa terbaik, namun memiliki jumlah prediksi salah tertinggi atau banyak contoh yang salah diklasifikasikan ke kelas normal, hal ini menunjukkan bias dalam model atau ketidakseimbangan kelas dalam data pelatihan.
- Kelas lain (berminyak, kering, kombinasi) memiliki jumlah prediksi benar yang jauh lebih rendah daripada kelas normal, menunjukkan bahwa model kesulitan dalam mengklasifikasikan contoh-contoh dari kelas tersebut.

c. *Classification Report*

Tabel 4.8 Nilai *Precision*, *Recall*, dan *F1-Score* Data *Training*

<b>Class</b>	<b>Precision</b>	<b>Recall</b>	<b>F1-Score</b>	<b>Support</b>
0 (berminyak)	0.18	0.04	0.07	116
1 (kering)	0.00	0.00	0.00	48
2 (kombinasi)	0.00	0.00	0.00	77
3 (normal)	0.48	0.92	0.63	231
<hr/>				
<i>accuracy</i>			0.46	472
<i>macro avg</i>	0.16	0.24	0.17	472
<i>weighted avg</i>	0.28	0.46	0.32	472

Hasil klasifikasi pada *data training* untuk nilai *precision*, *recall*, dan *F1-Score* ditunjukkan pada Tabel 4.8. Hasil akurasi menggunakan data training menunjukkan nilai sebesar 0.46 atau sebanyak 46% dari data telah berhasil diklasifikasikan dengan benar oleh model. Berikutnya, nilai *precision*, *recall*, dan *F1-Score* dengan menggunakan data *testing* ditunjukkan pada Tabel 4.9.

Tabel 4.9 Nilai *Precision*, *Recall*, dan *F1-Score* Data *Testing*

<b>Class</b>	<b>Precision</b>	<b>Recall</b>	<b>F1-Score</b>	<b>Support</b>
0 (berminyak)	0.40	0.14	0.21	14
1 (kering)	0.00	0.00	0.00	6
2 (kombinasi)	0.00	0.00	0.00	9
3 (normal)	0.52	0.96	0.67	28
<hr/>				
<i>accuracy</i>			0.51	57
<i>macro avg</i>	0.23	0.28	0.22	57
<i>weighted avg</i>	0.35	0.51	0.38	57

Secara keseluruhan, dari semua gambar yang telah digunakan dalam pemodelan sebagai data *training* maupun data *testing*, hasil akurasi sebesar 0.51 yang artinya 51% berhasil diprediksi dengan benar oleh model. Tanpa mempertimbangkan jumlah citra untuk setiap kelas, nilai presisi rata-rata untuk semua kelas adalah 23%, nilai rata-rata *recall* sebesar 28%, dan *F1-score* sebesar 22%. Disisi lain, jika melihat hasil dengan mempertimbangkan jumlah citra pada setiap kelas, nilai *precision* yang didapatkan sebesar 35%, nilai *recall* 51%, dan nilai *F1-score* 38%.

Berdasarkan *classification report*, dapat dilihat bahwa kelas berminyak, kering, dan kombinasi mendapatkan hasil *precision*, *recall*, dan *F1-score* yang rendah. Hal ini menunjukkan bahwa model mengalami kesulitan untuk mengklasifikasikan citra pada setiap kelas ini. Selain itu, nilai *precision* 0 pada kelas kering dan kombinasi menunjukkan bahwa model tidak dapat memprediksi dengan benar untuk kelas-kelas tersebut. Pada kelas normal, nilai *precision* dan *recall* yang cukup tinggi menunjukkan bahwa model dapat melakukan prediksi dengan benar lebih dari setengah gambar.

d. Hasil Evaluasi

Hasil evaluasi dari pemodelan setelah perbaikan citra menjelaskan seberapa baik pemodelan yang telah dilakukan untuk mencapai tujuan dari pengembangan model terutama setelah dilakukan perbaikan citra. Berdasarkan hasil evaluasi yang telah dilakukan, beberapa poin penting yang didapatkan antara lain:

- Hasil akurasi dengan menggunakan data citra setelah perbaikan citra mengalami peningkatan dibandingkan sebelumnya.
- Walaupun sudah dilakukan perbaikan citra, jumlah citra pada setiap tipe kulit wajah tetap sama sehingga pemodelan untuk setiap kelas memberikan hasil yang tidak seimbang, hal ini berakibat pada hasil accuracy, precision dan recall yang kurang maksimal untuk masing-masing kelas.
- Setelah dilakukan perbaikan citra, masih terdapat citra yang memiliki pencahayaan kurang baik dan tidak seimbang antar gambar. Seperti pada evaluasi sebelumnya, hal ini disebabkan oleh adanya perbedaan pencahayaan dikarenakan waktu pengambilan yang berbeda (pagi, siang, sore, malam) sehingga memengaruhi hasil citra yang diambil.
- Perbedaan waktu pengambilan citra memberikan efek kualitas citra yang dihasilkan menjadi kurang baik dan tidak seragam sehingga menyebabkan akurasi yang kurang baik atau kurang seimbang.

## BAB 5

### Kesimpulan dan Saran

#### 5.1 Kesimpulan

Model yang telah dikembangkan dengan berbagai arsitektur pada metode klasifikasi *Convolutional Neural Network* (CNN) menggunakan sebanyak 588 data citra, sebesar 80% dari data digunakan sebagai *data training* dan 20% digunakan sebagai *data testing*. Berdasarkan pengujian dan analisis hasil, dapat ditarik kesimpulan sebagai berikut:

- a. Pengambilan data citra sudah sesuai dengan protokol pengambilan citra dengan *setting* tempat *outdoor* dan menggunakan alat pengambilan gambar yang sama.
- b. Model yang dikembangkan mampu melakukan klasifikasi tipe kulit wajah dengan menggunakan citra foto walaupun hasil klasifikasi masih belum memberikan hasil yang maksimal.
- c. Model yang dikembangkan dengan CNN menggunakan arsitektur MobileNetV3Large dipilih sebagai arsitektur terbaik berdasarkan hasil yang ditunjukkan dengan hasil *validation accuracy* sebesar 46% serta *accuracy* pengujian untuk klasifikasi dengan menggunakan *data testing* sebesar 49%.
- d. Data citra yang digunakan dalam pemodelan memiliki beberapa permasalahan seperti jumlah yang tidak seimbang untuk setiap kelasnya, pencahayaan yang kurang merata pada citra, kualitas citra yang kurang baik khususnya untuk citra per *patch*, serta ciri-ciri yang kurang terlihat untuk setiap tipe kulit.
- e. Berdasarkan nilai kelembaban pada kulit wajah yang diambil pada saat pengumpulan data citra, disimpulkan bahwa kelembaban pada setiap titik pada wajah dinilai tidak berpengaruh pada klasifikasi tipe kulit wajah.
- f. Setelah dilakukan perbaikan citra dengan koreksi gamma, penyesuaian warna dengan LAB, serta normalisasi histogram, dilakukan pemodelan ulang dengan menggunakan arsitektur terbaik yang telah dipilih sebelumnya yaitu MobileNetV3. Berdasarkan pemodelan, nilai *validation accuracy* sebesar 48% dan nilai *training accuracy* sebesar 51%, hasil tersebut menunjukkan adanya peningkatan setelah data citra diperbaiki.
- g. Perbedaan hasil antara *macro average* dan *weighted average* menunjukkan bahwa distribusi kelas pada model tidak seimbang. Kelas normal mendominasi jumlah citra yang diprediksi dengan benar, sementara kelas lainnya mengalami kesulitan.

## 5.2 Saran

Adanya keterbatasan data citra membuat pemodelan yang dilakukan memberikan hasil yang kurang maksimal. Ketidakseimbangan pada kinerja model di antara kelas-kelas menunjukkan kebutuhan untuk penyesuaian lebih lanjut. Oleh karena itu, diharapkan penelitian kedepannya dapat melakukan pemodelan dengan menggunakan metode maupun data yang lebih baik lagi. Kuantitas dan kualitas citra lebih ditingkatkan dengan memastikan keseimbangan jumlah serta kualitas data antar kelas yang digunakan baik sebagai *data training* maupun *data testing*. Adanya keseimbangan kuantitas dan kualitas data citra untuk setiap kelasnya akan memberikan hasil pemodelan yang lebih stabil.

## Daftar Pustaka

- Alaska, K. P., & Hidayat, H. T. (2021a). Klasifikasi Citra Kulit Wajah Menggunakan Metode Naïve Bayes Berbasis Android. *JAISE : Journal of Artificial Intelligence and Software Engineering*, 1(1), 16–22.
- Alaska, K. P., & Hidayat, H. T. (2021b). Klasifikasi Citra Kulit Wajah Menggunakan Metode Naïve Bayes Berbasis Android. *JAISE : Journal of Artificial Intelligence and Software Engineering*, 1(1), 16–22.
- Amelia, R. D., Tritoasmoro, I. I., & Ibrahim, N. (2019a). Klasifikasi Jenis Kulit Wajah Menggunakan Metode Discrete Wavelet Transform dan Backpropagation. *E-Proceeding of Engineering*, 6(2), 4147–4153.
- Amelia, R. D., Tritoasmoro, I. I., & Ibrahim, N. (2019b). Klasifikasi Jenis Kulit Wajah Menggunakan Metode Discrete Wavelet Transform dan Backpropagation. *E-Proceeding of Engineering*, 6(2), 4147–4153.
- Badan Pusat Statistik. (2021). Berita Resmi Statistik. *Sensus Penduduk 2020*, 27, 1–52.
- Baumann, L. (2008). Understanding and Treating Various Skin Types: The Baumann Skin Type Indicator. *Dermatologic Clinics*, 26(3), 359–373.  
<https://doi.org/10.1016/j.det.2008.03.007>
- Baumann, L. (2010). *The Skin Type Solution: a Revolutionary Guide to Your Best Skin Ever*. Random House Publishing Group.
- Baumann, L. (2015). *Approach to Understanding Skin Type*. December.
- Cho, C. (2015). *The Little Book of Skin Care*. Imprint of HarperCollins Publishers.
- De Angeli, K., Gao, S., Danciu, I., Durbin, E. B., Wu, X. C., Stroup, A., Doherty, J., Schwartz, S., Wiggins, C., Damesyn, M., Coyle, L., Penberthy, L., Tourassi, G. D., & Yoon, H. J. (2022). Class imbalance in out-of-distribution datasets: Improving the robustness of the TextCNN for the classification of rare cancer types. *Journal of Biomedical Informatics*, 125. <https://doi.org/10.1016/j.jbi.2021.103957>
- EU-Indonesia Business Network. (2019). EIBN Sector Reports: Cosmetics. *Indonesian French Chamber of Commerce and Industry*, 1–93.
- Farage, M. A. (2019). The Prevalence of Sensitive Skin. *Frontiers in Medicine*, 6(May), 1–13. <https://doi.org/10.3389/fmed.2019.00098>

- Farhan, M., Rafi, Widodo, A., Wahyu, Rahman, M., & Arif. (2019a). Ekstraksi Ciri Pada Klasifikasi Tipe Kulit Wajah Menggunakan Metode Haar Wavelet. *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer*, 3(3), 2903–2909.
- Farhan, M., Rafi, Widodo, A., Wahyu, Rahman, M., & Arif. (2019b). Ekstraksi Ciri Pada Klasifikasi Tipe Kulit Wajah Menggunakan Metode Haar Wavelet. *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer*, 3(3), 2903–2909.
- Farnham, B., Tokyo, S., Boston, B., Sebastopol, F., & Beijing, T. (2019). *Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow Concepts, Tools, and Techniques to Build Intelligent Systems* (2nd ed.). O'Reilly Media, Inc.
- Firaz, T., Nusantara, B., Atmaja, R. D., Elektro, F. T., & Telkom, U. (2018a). Klasifikasi Jenis Kulit Wajah Pria Berdasarkan Tekstur Menggunakan Metode Gray Level Co-Occurrence Matrix (GLCM) dan Support Vector Machine (SVM). *EProceedings of Engineering*, 5(2), 2130–2137.
- Firaz, T., Nusantara, B., Atmaja, R. D., Elektro, F. T., & Telkom, U. (2018b). Klasifikasi Jenis Kulit Wajah Pria Berdasarkan Tekstur Menggunakan Metode Gray Level Co-Occurrence Matrix (GLCM) dan Support Vector Machine (SVM). *EProceedings of Engineering*, 5(2), 2130–2137.
- Goodfellow, I., Bengio, Y., & Courville, A. (2017). *Deep Learning*. MIT.
- Hanjaya, R. (2023). *Analisis Kualitas Dataset Terhadap Akurasi Model Deteksi dan Lokalisasi Objek*.
- Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., & Adam, H. (2017). *MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications*. <http://arxiv.org/abs/1704.04861>
- Indriyani, & Made Sudarma, I. (2020a). Classification of Facial Skin Type Using Discrete Wavelet Transform, Contrast, Local Binary Pattern and Support Vector Machine. *Journal of Theoretical and Applied Information Technology*, 98(5), 768–779.
- Indriyani, & Made Sudarma, I. (2020b). Classification of Facial Skin Type Using Discrete Wavelet Transform, Contrast, Local Binary Pattern and Support Vector Machine. *Journal of Theoretical and Applied Information Technology*, 98(5), 768–779.
- Krejcie, R. V., & Morgan, D. W. (1970). Determining sample Size for Research Activities, Educational and Psychological Measurement. *International Journal of Employment Studies*, 18(1), 89–123.

- Kusumaningrum, S. D., Muhimmah, I., & Korespondensi, P. (2023). *Analisis Faktor dan Metode untuk Menentukan Tipe Kulit Wajah: Tinjauan Literatur*. 10(4).  
<https://doi.org/10.25126/jtiik.2023106955>
- Molina, D. A., Causa, L., & Tapia, J. (2020). Reduction of Bias for Gender and Ethnicity from Face Images using Automated Skin Tone Classification. *BIOSIG 2020 - Proceedings of the 19th International Conference of the Biometrics Special Interest Group*.
- Muhaba, K. A., Dese, K., Aga, T. M., Zewdu, F. T., & Simegn, G. L. (2022). Automatic Skin Disease Diagnosis Using Deep Learning from Clinical Image and Patient Information. *Skin Health and Disease*, 2(1). <https://doi.org/10.1002/ski2.81>
- Muttaqin, Arafah, M., Jaya, A. K., Suryawan, M. A., Gustiana, Z., Banjarnahor, A. R., Bukidz, D. P., Simanjuntak, H. M., Saputra, N., & Fajrillah. (2023). *Implementasi Artificial Intelligence (AI) dalam Kehidupan* (1st ed.). Yayasan Kita Menulis.
- Nissa, N. F., Janiati, A., Cahya, N., Anton, & Astuti, P. (2021). Application of Deep Learning Using Convolutional Neural Network (CNN) Method For Women's Skin Classification. *Scientific Journal of Informatics*, 8(1), 144–153.  
<https://doi.org/10.15294/sji.v8i1.26888>
- Nurkhasanah, & Murinto. (2021). Klasifikasi Penyakit Kulit Wajah Menggunakan Metode Convolutional Neural Network. *Sainteks*, 18(2), 183.  
<https://doi.org/10.30595/sainteks.v18i2.13188>
- Putro, A. D., & Hermawan, A. (2021). Pengaruh Cahaya dan Kualitas Citra dalam Klasifikasi Kematangan Pisang Cavendish Berdasarkan Ciri Warna Menggunakan Artificial Neural Network. *MATRIK : Jurnal Manajemen, Teknik Informatika Dan Rekayasa Komputer*, 21(1), 215–228. <https://doi.org/10.30812/matrik.v21i1.1396>
- Reddy, S. P., & Manohar, A. B. S. R. (2019). Studies on Different CNN Algorithms for Face Skin Disease Classification Based on Clinical Images. *IEEE Access*, 7, 66505–66511. <https://doi.org/10.1109/ACCESS.2019.2918221>
- Resa Arif Yudianto, M., Sukmasetya, P., Abul Hasani, R., & Sasongko, D. (2022). Pengaruh Data Preprocessing terhadap Imbalanced Dataset pada Klasifikasi Citra Sampah menggunakan Algoritma Convolutional Neural Network. *Building of Informatics, Technology and Science (BITS)*, 4(3).  
<https://doi.org/10.47065/bits.v4i3.2575>
- Roslan, R., Razly, I. N. M., Sabri, N., & Ibrahim, Z. (2020). Evaluation of Psoriasis Skin Disease Classification Using Convolutional Neural Network. *IAES International*

- Journal of Artificial Intelligence*, 9(2), 349–355.  
<https://doi.org/10.11591/ijai.v9.i2.pp349-355>
- Rostamailis. (2005). *Perawatan Badan, Kulit dan Rambut*. Rineka Cipta.
- Safira, N. P., Magdalena, R., & Saidah, S. (2020). Klasifikasi Jenis Kulit Manusia Menggunakan Metode Gabor Wavelet Berbasis Android. *E-Proceeding of Engineering*, 7(2), 3693–3702.
- Saidah, S., Fuadah, Y. N., Alia, F., Ibrahim, N., Magdalena, R., & Rizal, S. (2021a). Facial Skin Type Classification Based on Microscopic Images Using Convolutional Neural Network (CNN). *Proceedings of the 1st International Conference on Electronics, Biomedical Engineering, and Health Informatics*, 746, 75–83.  
[https://doi.org/10.1007/978-981-33-6926-9\\_7](https://doi.org/10.1007/978-981-33-6926-9_7)
- Saidah, S., Fuadah, Y. N., Alia, F., Ibrahim, N., Magdalena, R., & Rizal, S. (2021b). Facial Skin Type Classification Based on Microscopic Images Using Convolutional Neural Network (CNN). *Proceedings of the 1st International Conference on Electronics, Biomedical Engineering, and Health Informatics*, 746, 75–83.  
[https://doi.org/10.1007/978-981-33-6926-9\\_7](https://doi.org/10.1007/978-981-33-6926-9_7)
- Santi, I. H., & Andari, B. (2019). Sistem Pakar Untuk Mengidentifikasi Jenis Kulit Wajah dengan Metode Certainty Factor. *INTENSIF: Jurnal Ilmiah Penelitian Dan Penerapan Teknologi Sistem Informasi*, 3(2), 159.  
<https://doi.org/10.29407/intensif.v3i2.12792>
- Sartika, Tritasmoro, I. I., & Susatio, E. (2012a). *Klasifikasi Jenis Kulit Wajah Berdasarkan Analisis Tekstur dengan Metode K-Nearest Neighbor*. 0–6.
- Sartika, Tritasmoro, I. I., & Susatio, E. (2012b). *Klasifikasi Jenis Kulit Wajah Berdasarkan Analisis Tekstur dengan Metode K-Nearest Neighbor*. 0–6.
- Sinaulan, C. D., & Hantara, A. (2021). Model Klasifikasi Permasalahan Kulit Wajah Menggunakan Metode Support Vector Machine. *Jurnal Pendidikan Dan Kewirausahaan*, 9(1), 297–308. <https://doi.org/10.47668/pkwu.v9i1.246>
- Sofia Saidah, Suparta, I. P. Y. N., & Suhartono, E. (2022). Modifikasi Convolutional Neural Network Arsitektur GoogLeNet dengan Dull Razor Filtering untuk Klasifikasi Kanker Kulit. *Jurnal Nasional Teknik Elektro Dan Teknologi Informasi*, 11(2), 148–153. <https://doi.org/10.22146/jnteti.v11i2.2739>
- Statista. (2020). *Consumer Markets Beauty & Personal Care in Indonesia*.  
<https://www.statista.com/outlook/cmo/beauty-personal-care/indonesia#revenue>

- Suartika E.P, I. W., Wijaya, A. Y., & Soelaiman, R. (2016). Klasifikasi Citra Menggunakan Convolutional Neural Network (CNN) pada Caltech 101. *JURNAL TEKNIK ITS*, 5(1), 2337–3539.
- Suryaningsih, B. E. (2018). Characteristics of facial melasma on Javanese women in Yogyakarta, Indonesia. *Journal of Pakistan Association of Dermatologists*, 28(3), 306–310.
- Susanti, S. (2014). *500 Rahasia Cantik Alami Bebas Jerawat (I)*. Gramedia Widiasarana Indonesia.
- Swiss Business Hub Indonesia. (2018). *Fact Finding Mission to Indonesia*. Switzerland Global Enterprise. [https://www.sge.com/sites/default/files/article/downloads/ffm\\_indonesia\\_cosmetics\\_2018.pdf](https://www.sge.com/sites/default/files/article/downloads/ffm_indonesia_cosmetics_2018.pdf)
- Syahrul, M., & Putra, S. (n.d.). *Perbaikan dan Peningkatan Kualitas Citra Menggunakan CNN*.
- Taieb, C., Gayraud, F., Dinet, D., & Sayag, M. (2021). Interest in Micellar Solution to Reduce Unpleasant Skin Sensations. *Clinical, Cosmetic and Investigational Dermatology*, 14, 1017–1022. <https://doi.org/10.2147/CCID.S318802>
- Triyanto, E. (2010). Pengalaman Masa Pubertas Remaja Studi Fenomenologi di Purwokerto. *Jurnal Ners*, 5(2), 147–153.
- Velasco, J., Pascion, C., Alberio, J. W., Apuang, J., Cruz, J. S., Gomez, M. A., Molina, B., Tuala, L., Thio-Ac, A., & Jorda, R. (2019). A Smartphone-based Skin Disease Classification Using Mobilenet CNN. *International Journal of Advanced Trends in Computer Science and Engineering*, 8(5), 2632–2637. <https://doi.org/10.30534/ijatcse/2019/116852019>
- Wardani, K. R., & Leonardi, L. (2023). Klasifikasi Penyakit pada Daun Anggur menggunakan Metode Convolutional Neural Network. *Jurnal Tekno Insentif*, 17(2), 112–126. <https://doi.org/10.36787/jti.v17i2.1130>
- Wijanarko A.P, T. (2014). Pengaruh Intensitas Cahaya Terhadap Hasil Pengenalan Citra dengan Gray Level Co-Occurrence Matrix dan Probabilistic Neural Network. *Jurnal Teknologi Informasi Dan Komunikasi*.
- Wu, Z., Zhao, S., Peng, Y., He, X., Zhao, X., Huang, K., Wu, X., Fan, W., Li, F., Chen, M., Li, J., Huang, W., Chen, X., & Li, Y. (2019). Studies on Different CNN Algorithms for Face Skin Disease Classification Based on Clinical Images. *IEEE Access*, 7, 66505–66511. <https://doi.org/10.1109/ACCESS.2019.2918221>

- Wulandari, S. A. (2019). Pengelompokan Jenis Kulit Normal, Berminyak dan Kering Menggunakan 4-Connectivity dan 8-Connectivity Region Properties Berdasarkan Ciri Rerata Bound. *Jurnal Transformatika*, 17(1), 78.  
<https://doi.org/10.26623/transformatika.v17i1.1341>
- Zakiah, Patmasari, R., & Saidah, S. (2021). Klasifikasi Jenis Kulit Wajah Menggunakan Metode Convolutional Neural Network (CNN). *E-Proceeding of Engineering*, 8(6), 11610.
- ZAP Beauty Clinic & Markplus Inc. (2018). ZAP Beauty Index 2018. *Mark Plus Inc*, 1–33.
- ZAP Beauty Clinic & Markplus Inc. (2021). ZAP Beauty Index 2021. *Mark Plus Inc*.

# LAMPIRAN A

## Lembar Penjelasan Kepada Calon Subjek

Assalamualaikum wr. wb.

Yth. Bapak/Ibu/Sdr.

Saya, Izzati Muhimmah, Ph.D., dari Jurusan Informatika, akan melakukan penelitian berjudul “Purwarupa Sistem Analisis Kulit wajah Berdasarkan Wajah”. Penelitian ini bertujuan mengembangkan prototipe sistem yang mampu mendeteksi permasalahan kulit wajah seperti jerawat, keriput, kemerahan, bercak hitam, kulit kering, kulit kusam, kulit berminyak, dan kulit normal. Untuk dapat membangun prototipe yang dapat mendeteksi permasalahan kulit tersebut dengan benar, prototipe kami memerlukan data berupa citra atau foto wajah. Foto wajah dibutuhkan untuk menjadi acuan penelitian dalam memperlajati sifat-sifat permasalahan kulit.

Tim peneliti meminta kesediaan Bapak/Ibu/Sdr. untuk berpartisipasi dalam penelitian ini. Penelitian ini memerlukan lima foto wajah dari lima sudut pandang yang berbeda (lurus, sering kanan, sering kiri).

Beberapa hal terkait penelitian yang perlu saya sampaikan adalah sebagai berikut:

1. Kesekarelaan dalam penelitian

Bapak/Ibu/Sdr. Bebas memilih keikutsertakan dalam penelitian ini tanpa ada paksaan. Bila sudah memutuskan untuk ikut, Bapak/Ibu/Sdr. Juga bebas untuk mengundurkan diri/berubah pikiran setiap saat tanpa dikenai denda atau pun sanksi apapun.

2. Kerahasiaan penelitian

Semua data yang berkaitan dengan identitas dan informasi yang digunakan dalam penelitian ini terjamin kerahasiannya dan hanya akan diketahui oleh peneliti dan staf penelitian. Hasil penelitian akan dipublikasikan tanpa identitas pribadi. Dalam publikasi hasil penelitian, kemungkinan akan ada beberapa foto subjek yang ditampilkan, namun identitas pribadi tidak akan disebutkan dan tanda-tanda ciri khas subjek akan ditutup akan menjaga kerahasiaan.

3. Prosedur penelitian

Bapak/Ibu/Sdr. yang memenuhi syarat penelitian dan setuju untuk berpartisipasi dalam penelitian akan dilakukan pemotretan (pengambilan foto wajah) menggunakan kamera atau alat khusus yang ada di klinik kecantikan dari jarak kurang lebih 1 meter. Selanjutnya foto wajah akan dinilai kelayakannya dan dianotasi oleh dokter SpKK serta diolah lebih lanjut untuk kepentingan pengembangan sistem komputer.

4. Kewajiban subjek penelitian

Subjek penelitian wajib hadir di lokasi yang disepakati untuk keperluan pemeriksaan dan pengambilan foto/citra wajah.

5. Risiko/efek samping penelitian

Risiko yang muncul dalam penelitian ini adalah tersebarnya foto wajah dan diagnosis medis subjek sehingga muncul gangguan terhadap privasi. Untuk itu, peneliti mengambil Langkah-langkah sebagai berikut:

(1) Data dan foto di komputer hanya diperuntukkan bagi petugas yang terlibat dalam penelitian dan dapat diakses dengan menggunakan password dan akses pribadi. (2) Sebelum mengakses setiap informasi yang berkaitan dengan penelitian, semua petugas harus menandatangani formulir pernyataan persetujuan untuk melindungi keamanan dan kerahasiaan informasi subjek. (3) Sebelum turut serta dalam pelaksanaan penelitian, semua petugas harus menandatangani persetujuan untuk menjaga kerahasiaan informasi subjek. (4) Apabila mungkin, identifikasi subjek penelitian dihapus (anonim) dari informasi yang berhubungan dengan penelitian. Data penelitian akan disimpan oleh peneliti selama 3 (tiga) tahun.

6. Manfaat dan kompensasi bagi subjek penelitian

Tidak ada manfaat secara langsung yang didapatkan oleh subjek yang terlibat dalam penelitian ini. Akan tetapi, setelah penelitian ini selesai dilaksanakan, diharapkan hasilnya dapat memberikan manfaat berupa memfasilitasi masyarakat agar dapat melakukan *self-assessment* untuk kondisi wajahnya serta dapat menjadi *support system* untuk dokter dalam mendiagnosis masalah kulit wajah.

Peneliti akan memberikan kompensasi untuk mengganti waktu yang disediakan oleh subjek untuk ikut serta dalam penelitian ini berupa souvenir.

7. Informasi tambahan

Peneliti memberikan kesempatan kepada Bapak/Ibu/Sdr. untuk bertanya atau mendapatkan semua informasi yang masih diperlukan untuk melaporkan semua kejadian/kerugian selama berjalannya penelitian. Bapak/Ibu/Sdr. dapat menghubungi peneliti:

Nama : Izzati Muhimmah, Ph.D.

Nomor HP : +62 811-2535-151

Email : [izzati@uii.ac.id](mailto:izzati@uii.ac.id)

Alamat : Fakultas Teknologi Industri/ Prodi Teknik Informatika  
Universitas Islam Indonesia  
Jl. Kaliurang KM. 14.5, Sleman, Yogyakarta.

## LAMPIRAN B

### *INFORMED CONSENT*

#### (PERNYATAAN PERSETUJUAN IKUT PENELITIAN)

Yang bertanda tangan di bawah ini:

Nama : \_\_\_\_\_

Umur : \_\_\_\_\_

Jenis Kelamin : \_\_\_\_\_

Alamat : \_\_\_\_\_

Telah mendapat keterangan secara terinci dan jelas mengenai:

1. Penelitian yang berjudul “Purwarupa Sistem Analisis Kulit Wajah Berdasarkan Citra Wajah”.
2. Perlakuan yang akan didapatkan.
3. Manfaat dalam mengikuti penelitian.
4. Bahaya yang akan timbul.
5. Prosedur penelitian.
6. Mendapat kesempatan untuk mengajukan pertanyaan seputar penelitian tersebut.

Oleh karena itu saya bersedia / tidak bersedia\*) secara sukarela untuk berpartisipasi dalam penelitian ini dengan penuh kesadaran dan tanpa paksaan serta saya mengizinkan / tidak mengizinkan\*) foto saya dicetak untuk publikasi dengan identitas yang dirahasiakan.

Yogyakarta, \_\_\_\_\_

Peneliti,

Subjek,

Izzati Muhimmah, Ph.D

\_\_\_\_\_

Saksi,

\_\_\_\_\_

# LAMPIRAN C

## Dokumentasi Pengambilan Data Citra

