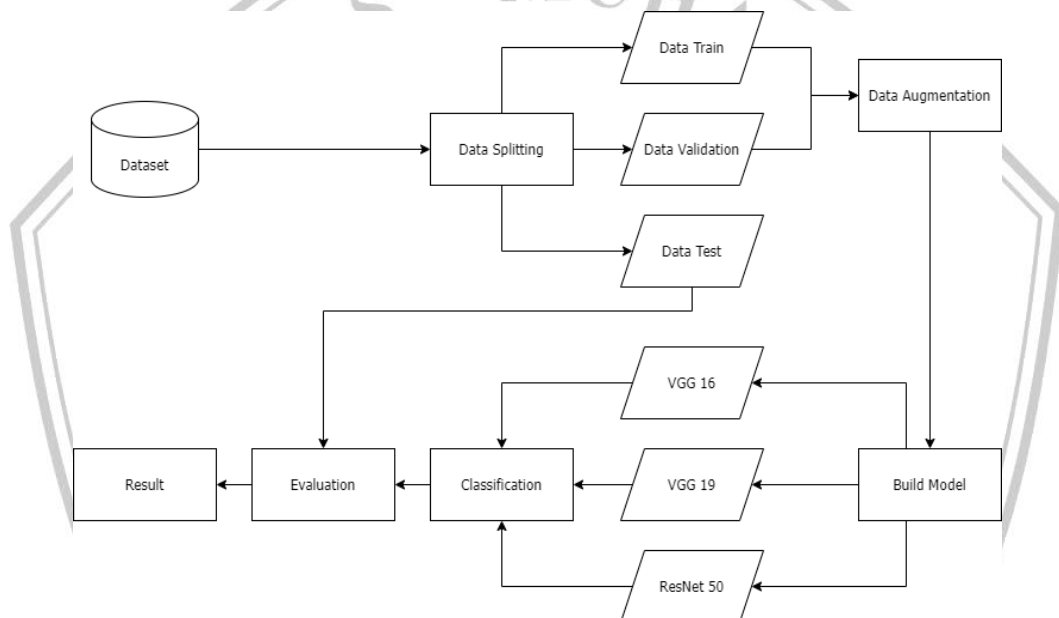


## BAB III METODOLOGI PENELITIAN

### 3.1. Tahapan Penelitian

Pada tahapan ini dilakukan pembahasan mengenai alur model penelitian yang diusulkan untuk mengklasifikasikan penyakit gigi dan lidah. Bagan tahapan model penelitian meliputi dataset, implementasi preprocessing, implementasi augmentasi, build model, proses klasifikasi / *training* model, evaluasi model, dan hasil klasifikasi dapat dilihat pada gambar 5.



**Gambar 5.** Tahapan Penelitian

### 3.2. Lingkungan Kerja

Lingkungan kerja pada penelitian ini meliputi perangkat keras serta perangkat lunak. Perangkat keras yang digunakan pada penelitian ini yaitu laptop ROG Strix G512LI\_G512LI dengan spesifikasi CPU Intel(R) Core(TM) i7-10750H CPU @ 2.60GHz (12 CPUs), ~2.6GHz, dan kartu grafis NVIDIA GeForce GTX 1650 Ti. Laptop yang digunakan dilengkapi dengan 16GB RAM dan menjalankan sistem operasi Windows. Selain itu, terdapat BIOS dengan versi G512LI.311 (type: UEFI).

Perangkat lunak yang digunakan pada penelitian ini berupa sistem operasi Windows 11 yang menjalankan aplikasi Microsoft Word untuk dokumentasi, serta

Google Collaboratory dan Google Drive untuk pengolahan dan penyimpanan data. Google Collaboratory digunakan untuk mendukung bahasa pemrograman Python versi 3.10.12 yang dipergunakan untuk melakukan pengolahan dan pelatihan data citra. Google Drive juga digunakan sebagai sarana akses dan penyimpanan dataset gambar gigi dan lidah untuk klasifikasi kesehatan gigi dan lidah.

### 3.3. Dataset

Dalam penelitian ini, digunakan dataset citra yang berasal dari situs Kaggle dan Roboflow. Terdapat 8.400 gambar yang terbagi dalam 12 kelas, dengan 6 kelas untuk gambar gigi, dan 6 kelas untuk gambar lidah. Contoh citra dari setiap kelas dapat dilihat pada gambar 6.



**Gambar 6.** Sampel Dataset

Terdapat 12 kelas dalam dataset yang digunakan pada penelitian ini. Nama untuk setiap kelas dalam dataset dapat dilihat pada tabel 1.

**Tabel 1.** Total Citra Setiap Kelas

Kelas		Total Citra
Gigi	Lidah	
<i>Caries</i>	<i>Fissure</i>	700
<i>Calculus</i>	<i>Geographic</i>	700
<i>Discoloration</i>	<i>Diabetic</i>	700
<i>Gingivitis</i>	<i>Coated</i>	700

<i>Hypodontia</i>	<i>Black Hairy</i>	700
<i>Normal Teeth</i>	<i>Normal Tongue</i>	700

### 3.4. Splitting Dataset

Pada tahapan berikutnya dilakukan pembagian dataset (*splitting dataset*). Dataset dibagi menggunakan *library splitfolders*, dengan proporsi pembagian 70% sebagai data train, 20% sebagai data validation, dan 10% sebagai data test. Data train dan validation digunakan untuk melatih model pada proses pelatihan, sedangkan data test digunakan untuk menguji model yang sudah dilatih.

### 3.5. Preprocessing Dataset

Pada penelitian ini dilakukan tahap *preprocessing* dengan mengubah ukuran gambar (*crop*) menjadi 224x224 pixel. Tahap ini bertujuan untuk menyamakan setiap data citra yang digunakan dalam melatih model. Ukuran ini umumnya digunakan untuk model yang diusulkan penulis VGG16, VGG19, dan ResNet50. *Preprocessing* memungkinkan model bekerja secara efisien dengan mempertahankan informasi yang cukup untuk menghasilkan representasi akurat dari citra.

### 3.6. Augmentasi Data

Pada tahapan ini dilakukan proses augmentasi data pada citra penyakit gigi dan lidah. Beberapa jenis proses augmentasi diintegrasikan pada dataset yang telah dilakukan preprocessing sebelumnya. Proses augmentasi dapat dilakukan untuk mengatasi ketidakseimbangan data pada kelas yang ada dalam dataset atau mengurangi terjadinya *overfitting*. Detail jenis proses augmentasi dan deskripsi penjelasannya dapat dilihat pada tabel 2.

**Tabel 2.** Detail Jenis Proses Augmentasi Citra

No	Augmentasi	Deskripsi Proses	Nilai
1	<i>Rescale</i>	Mengubah nilai piksel dengan rentang [0 - 1] dengan membagi setiap piksel dengan 255	1./255
2	<i>Shear Range</i>	Memindahkan titik dalam gambar secara acak dalam rentang [0 - 2]	0.2
3	<i>Zoom Range</i>	Melakukan zoom in dan out pada gambar dalam rentang [0 - 2]	0.2
4	<i>Rotation Range</i>	Merotasi gambar dalam rentang tertentu	45°
5	<i>Horizontal &amp; Vertical Flip</i>	Melakukan perputaran gambar secara horizontal dan vertical dengan acak	<i>True</i>
6	<i>Fill Mode</i>	Mengisi area baru yang terbentuk oleh transformasi dengan nilai piksel terdekat	<i>Nearest</i>

### 3.7. Rancangan Model Arsitektur

Pada penelitian ini penulis mengusulkan 3 model arsitektur berbeda, VGG16, VGG19, dan ResNet50 untuk melakukan klasifikasi penyakit gigi dan lidah. Setiap model yang digunakan disesuaikan dengan kebutuhan penelitian untuk mendapatkan kinerja model terbaik di setiap pengujian. Berikut ini adalah parameter-parameter yang digunakan dan model summary dari setiap model.

**Tabel 3.** Parameter Dalam Penerapan Model

Loss	Optimizer	Learning rate	Momentum	Steps per epoch	Verbose
Categorical Crossentropy	SGD	0.0001	0.9	100	1

**Tabel 4.** Model Summary VGG16

Layer (Type)	Output Shape	Param
vgg16 (Functional)	(None, 7, 7, 512)	14,714,688
conv2d (Conv2D)	(None, 7, 7, 64)	294,976
max_pooling2d (MaxPooling2D)	(None, 3, 3, 64)	0
flatten (Flatten)	(None, 576)	0
dense (Dense)	(None, 512)	295,424
dense_1 (Dense)	(None, 12)	6,156

**Tabel 5.** Model Summary VGG19

Layer (Type)	Output Shape	Param
vgg19 (Functional)	(None, 7, 7, 512)	20,024,384
conv2d (Conv2D)	(None, 7, 7, 64)	294,976
max_pooling2d (MaxPooling2D)	(None, 3, 3, 64)	0
flatten (Flatten)	(None, 576)	0
dense (Dense)	(None, 512)	295,424
dense_1 (Dense)	(None, 12)	6,156

**Tabel 6.** Model Summary ResNet50

Layer (Type)	Output Shape	Param
resnet50 (Functional)	(None, 7, 7, 2048)	23,587,712
conv2d (Conv2D)	(None, 7, 7, 64)	1,179,712
max_pooling2d (MaxPooling2D)	(None, 3, 3, 64)	0
flatten (Flatten)	(None, 576)	0
dense (Dense)	(None, 512)	295,424
dense_1 (Dense)	(None, 12)	6,156

### 3.8. Skenario Pengujian

Pada tahapan ini, dilakukan beberapa skenario pengujian. Semua skenario yang dilakukan dengan batch size sebesar 32, learning rate sebesar 0.0001, dan

momentum sebesar 0.9. Berikut merupakan detail skenario pengujian yang dilakukan:

- a. Skenario Pengujian pertama dilakukan pelatihan pada dataset original (tanpa proses augmentasi) dengan menggunakan beberapa model seperti VGG16, VGG19, dan ResNet50 dengan epoch pelatihan sebesar 25. Tujuan dari skenario ini adalah untuk mengevaluasi baseline performance dari masing-masing model tanpa adanya modifikasi data.
- b. Skenario pengujian kedua dilakukan dengan menerapkan proses augmentasi data terhadap dataset yang digunakan untuk pelatihan menggunakan model seperti VGG16, VGG19, dan ResNet50 dengan jumlah epoch 25. Augmentasi diterapkan untuk mengetahui pengaruhnya terhadap pengujian yang dilakukan.
- c. Skenario pengujian ketiga dilakukan menggunakan proses augmentasi, dengan jumlah epoch yang ditingkatkan secara bertahap menjadi 25, 50, dan 75. Tujuan dari skenario ini adalah untuk mengevaluasi dampak dari peningkatan jumlah epoch terhadap kinerja model dan untuk menentukan jumlah epoch optimal untuk pelatihan yang lebih efektif.