

## BAB III

### METODOLOGI PENELITIAN

#### 3.1 Subjek dan Objek Penelitian

Subjek dalam penelitian ini merupakan pengguna aplikasi cari *cafe*, yang terdiri dari laki-laki dan perempuan dengan rentang usia antara 18 hingga 29 tahun yang akrab dengan teknologi, terutama aplikasi berbasis *web*. Objek penelitian ini adalah *cafe* yang dipilih pengguna dari hasil rekomendasi model *deep learning* pada aplikasi cari *cafe* menggunakan prediksi identitas dan foto selfie.

#### 3.2 Alat dan Bahan Penelitian

Dalam penelitian ini, digunakan berbagai alat dan bahan untuk mendukung dan memfasilitasi pembuatan model *deep learning* tiga fungsi yaitu deteksi e-KTP, prediksi identitas dari foto selfie pada aplikasi cari *cafe*. Berikut adalah alat dan bahan yang digunakan dalam penelitian ini.

##### a. Alat

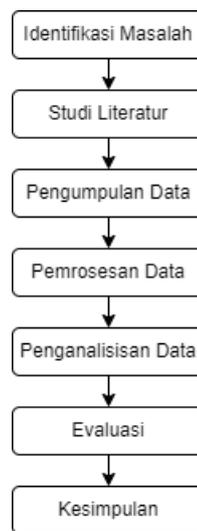
- Laptop HP RAM 8GB SSD 240
- *Windows 11*
- *Google Colaboratory*
- *Jupyter Notebook*
- *Google Form*
- *Google Chrome*

##### b. Bahan

Bahan yang digunakan adalah hasil tanggapan dari kuesioner *Google Form* yang disebarluaskan.

#### 3.3 Diagram Alir Penelitian / Proses Penelitian

Untuk memahami berbagai tahapan dalam penelitian, dilakukan penggambaran diagram alir penelitian. Gambar 3.1 menunjukkan diagram alir penelitian ini:



Gambar 3.1 Diagram Alir Penelitian

### 3.3.1 Identifikasi Masalah

Masalah penelitian dan tema penelitian mempengaruhi kualitas penelitian. Pada tahap awal penelitian, dilakukan identifikasi masalah yang akan menjadi fokus utama dalam penelitian. Dalam tahap identifikasi masalah, juga dilakukan riset yang mendalam. Tahap ini sangat penting, karena berperan sebagai landasan untuk mencapai tujuan penelitian. Masalah pada penelitian ini adalah belum adanya model *deep learning* untuk tiga fungsi dalam aplikasi cari *cafe* yaitu deteksi e-KTP, prediksi identitas dari foto selfie dan verifikasi identitas pengguna.

### 3.3.2 Studi Literatur

Setelah tahap identifikasi masalah, dilanjutkan dengan tahap studi literatur, di mana peneliti harus mengumpulkan atau mencari sejumlah informasi, buku, ataupun penelitian sebelumnya terkait dengan masalah dan tujuan penelitian. Studi literatur dilakukan untuk mendapatkan sumber acuan atau referensi yang berkaitan untuk menyelesaikan masalah atau topik penelitian. Studi literatur penelitian ini telah dirangkum pada kajian pustaka dan dasar teori.

### 3.3.3 Pengumpulan Data

Pada tahap pengumpulan data, dilakukan kegiatan untuk menemukan dan mengumpulkan data yang diperlukan guna mencapai tujuan penelitian serta memecahkan masalah yang ada. Dikarenakan penelitian ini memberikan tiga fungsi

maka data yang dikumpulkan terbagi menjadi tiga. *Dataset* ini dapat dikumpulkan dengan observasi maupun wawancara. Tabel 3.1 menunjukkan *dataset* yang digunakan dalam penelitian ini:

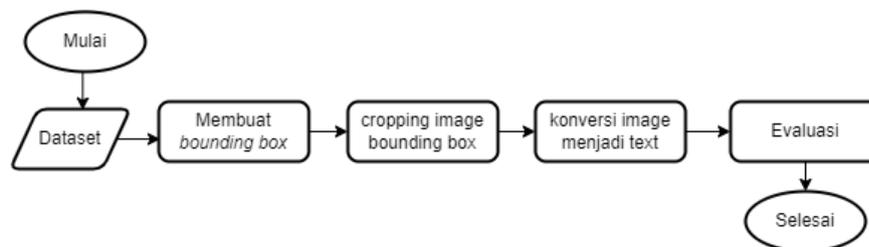
Tabel 3.1 *Dataset*

<b>Fungsi</b>	<b>Detail Fungsi</b>	<b>Dataset</b>	<b>Jumlah</b>
Fungsi 1	Membaca gambar e-KTP secara otomatis.	Gambar e-KTP.	70 gambar e-KTP.
Fungsi 2	Prediksi identitas dari foto selfie pengguna dan merekomendasikan <i>cafe</i> .	Foto selfie pengguna.	70 foto selfie pengguna.
		11 Komponen kuesioner identitas.	70 Jawaban 11 komponen kuesioner identitas.
		Info <i>cafe</i> .	25 Info <i>cafe</i> .
Fungsi 3	Memverifikasi identitas pengguna dari foto pada gambar e-KTP dengan foto selfie pengguna.	a = Foto selfie pengguna.	70 foto selfie pengguna.
		b = Gambar e-KTP.	70 gambar e-KTP.

### 3.3.4 Pemrosesan Data

Setelah data terkumpul maka dilakukanlah tahap pemrosesan data. Pada tahap pemrosesan data, data mentah yang sebelumnya dikumpulkan akan diolah untuk memberikan informasi yang dapat digunakan untuk penelitian. Pemrosesan data juga terbagi menjadi tiga:

- Pada pemrosesan data Fungsi 1 digunakan teknologi *Tesseract* OCR. Teknologi ini mampu mengkonversi gambar yang mengandung teks atau karakter menjadi karakter ASCII yang dapat dipahami oleh komputer. Gambar 3.2 menunjukkan diagram alir pemrosesan data Fungsi 1 menggunakan *Tesseract* OCR:



Gambar 3.2 Pemrosesan data Fungsi 1

Pemrosesan data Fungsi 1 dimulai dengan memasukkan *dataset* Fungsi 1. *Dataset* tersebut selanjutnya dibuat *bounding box* menggunakan YOLOv5. Setelah *bounding box* terbentuk, dilakukan proses pemangkasan gambar dengan memotong gambar sesuai dengan *bounding box* yang telah dibuat. Selanjutnya gambar-gambar dikonversi menjadi teks menggunakan *Tesseract* OCR. Selama proses ini, juga dilakukan pembersihan data di mana data yang tidak relevan dapat dihapus atau diganti. Terakhir dilakukan evaluasi dengan menggunakan WER.

- b. Pada pemrosesan Fungsi 2 digunakan algoritma CNN. Algoritma ini umumnya digunakan untuk analisis citra atau gambar. Proses pelatihan model dilakukan dengan menggunakan dua jenis fungsi aktivasi, yaitu *softmax* dan *sigmoid*. Aktivasi *softmax* umumnya digunakan dalam masalah klasifikasi multi kelas, sedangkan aktivasi *sigmoid* digunakan dalam masalah biner. Setiap jenis fungsi aktivasi diimplementasikan menggunakan tiga model berbeda, yaitu model CNN yang secara khusus dirancang untuk menangani masalah yang sedang diteliti, model VGG16, dan model ResNet50. Implementasi CNN terdiri dari empat *layer* konvolusi, tiga *layer Max Pooling2D* dan tiga *layer fully connected*. Tabel 3.2 menunjukkan arsitektur model CNN dengan fungsi aktivasi *softmax*:

Tabel 3.2 Arsitektur model CNN *softmax*

<b>Layer</b>	<b>Output Shape</b>
<i>Input</i>	(100, 100, 3)
<i>Conv2D</i>	(98, 98, 32)
<i>MaxPooling2D</i>	(49, 49, 32)
<i>Conv2D</i>	(47, 47, 64)
<i>MaxPooling2D</i>	(23, 23, 64)
<i>Conv2D</i>	(21, 21, 128)
<i>MaxPooling2D</i>	(10, 10, 128)
<i>Conv2D</i>	(8, 8, 256)
<i>MaxPooling2D</i>	(4, 4, 256)
<i>Flatten</i>	(4096)
<i>Dense</i>	(64)
<i>Dropout</i>	(64)
<i>Dense</i>	(11)

Model VGG16 yang digunakan terdapat sedikit modifikasi dengan menambahkan *layer dropout*. Tabel 3.3 menunjukkan arsitektur *transfer learning* model VGG16 termodifikasi dengan fungsi aktivasi *softmax*:

Tabel 3.3 Arsitektur model VGG16 *softmax*

<b>Layer</b>	<b>Output Shape</b>
<i>Input</i>	(100, 100, 3)
VGG16 ( <i>functional</i> )	(3, 3, 512)
<i>Flatten</i>	(4608)
<i>Dense</i>	(64)
<i>Dropout</i>	(64)
<i>Dense</i>	11

Model ResNet50 yang digunakan terdapat sedikit modifikasi dengan menambahkan *layer flatten*, *layer dense*, dan *layer dropout*. Tabel 3.4 menunjukkan arsitektur *transfer learning* model ResNet50 termodifikasi dengan fungsi aktivasi *softmax*:

Tabel 3.4 Arsitektur model ResNet50 *softmax*

<b>Layer</b>	<b>Output Shape</b>
<i>Input</i>	(100, 100, 3)
ResNet50 ( <i>functional</i> )	(4, 4, 2048)
<i>Flatten</i>	(32768)
<i>Dense</i>	(64)
<i>Dropout</i>	(64)
<i>Dense</i>	(11)

Pada aktivasi *sigmoid*, model diimplementasikan pada setiap folder. Setiap folder ini merupakan kelas yang digunakan dalam pemrosesan data dengan fungsi aktivasi *softmax* dan didalam setiap folder tersebut terdapat dua folder didalamnya. Setiap folder dilatih menggunakan arsitektur model yang sama dengan model yang menggunakan fungsi aktivasi *softmax*, kecuali perbedaannya terletak pada fungsi aktivasi yang digunakan. Tabel 3.5, Tabel 3.6 dan Tabel 3.7 menunjukkan arsitektur model CNN, model *transfer learning* VGG16, dan model *transfer learning* ResNet50 dengan fungsi aktivasi *sigmoid*:

Tabel 3.5 Arsitektur model CNN *sigmoid*

<b>Layer</b>	<b>Output Shape</b>
<i>Input</i>	(100, 100, 3)
<i>Conv2D</i>	(98, 98, 32)
<i>MaxPooling2D</i>	(49, 49, 32)
<i>Conv2D</i>	(47, 47, 64)
<i>MaxPooling2D</i>	(23, 23, 64)
<i>Conv2D</i>	(21, 21, 128)
<i>MaxPooling2D</i>	(10, 10, 128)
<i>Conv2D</i>	(8, 8, 256)
<i>MaxPooling2D</i>	(4, 4, 256)
<i>Flatten</i>	(4096)
<i>Dense</i>	(64)
<i>Dropout</i>	(64)
<i>Dense</i>	(1)

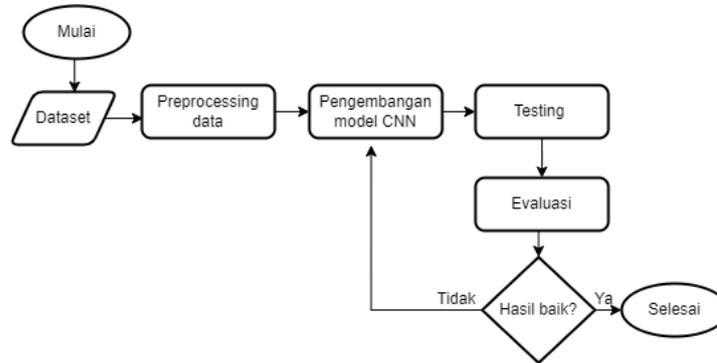
Tabel 3.6 Arsitektur model VGG16 *sigmoid*

<b>Layer</b>	<b>Output shape</b>
<i>Input</i>	(100, 100, 3)
<i>VGG16 (functional)</i>	(3, 3, 512)
<i>Flatten</i>	(4608)
<i>Dense</i>	(64)
<i>Dropout</i>	(64)
<i>dense</i>	(1)

Tabel 3. 7 Arsitektur model ResNet50 *sigmoid*

<b>Layer</b>	<b>Output Shape</b>
<i>Input</i>	(100, 100, 3)
<i>ResNet50</i>	(4, 4, 2048)
<i>Flatten</i>	(32768)
<i>Dense</i>	(64)
<i>Dropout</i>	(64)
<i>Dense</i>	(1)

Gambar 3.3 menunjukkan diagram alir pemrosesan data Fungsi 2 menggunakan CNN:



Gambar 3.3 Pemrosesan data Fungsi 2

Pemrosesan data Fungsi 2 dimulai dengan memasukkan *dataset* Fungsi 2 yang selanjutnya *dataset* tersebut melakukan tahap *preprocessing* data. Selanjutnya *dataset* masuk ke dalam proses pengembangan model CNN. Kemudian, model CNN tersebut dilakukan pengujian. Setelah pengujian, dilakukan evaluasi menggunakan *Confusion matrix* Apabila evaluasi menghasilkan performa yang memuaskan atau baik, model CNN tersebut akan dipertahankan. Namun, jika evaluasi menunjukkan hasil yang kurang memuaskan atau kurang baik, model CNN akan dikembangkan kembali dengan kembali ke tahap pengembangan model CNN untuk melakukan perbaikan atau peningkatan.

- c. Pada pemrosesan Fungsi 3 digunakan algoritma SNN. Algoritma ini bertujuan untuk membandingkan kesamaan atau mencocokkan dua *input* guna menghasilkan satu *output* dengan nilai yang sesuai dengan kedua *input* tersebut. Sebelum membentuk jaringan SNN, terlebih dahulu dibuat *base network* yang akan menjadi komponen utama dalam arsitektur SNN. *Base network* akan menerima *input* gambar dan menghasilkan vektor representasi yang mewakili fitur-fitur dari gambar tersebut. Tabel 3.8 menunjukkan arsitektur *base network*:

Tabel 3.8 Arsitektur *base network*

<b>Layer</b>	<b>Output Shape</b>
Input	(56, 56, 3)
Conv2D	(54, 54, 32)

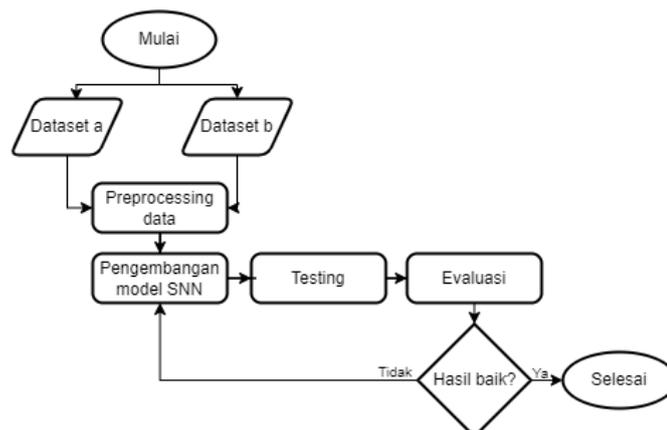
Layer	Output Shape
MaxPooling2D	(27, 27, 32)
Conv2D	(25, 25, 64)
MaxPooling2D	(12, 12, 64)
Flatten	(9216)
Dense ( <i>first</i> )	(128)
Dense ( <i>second</i> )	(128)
Dense ( <i>third</i> )	(128)

Selanjutnya merupakan pembuatan arsitektur SNN dengan menggunakan *base network* yang telah dibuat sebelumnya. SNN menghubungkan dua *input* (*input\_a* dan *input\_b*) ke *base network* dan mengukur skor kesamaan atau penilaian seberapa mirip kedua input tersebut. Tabel 3.9 menunjukkan arsitektur SNN:

Tabel 3.9 Arsitektur SNN

Layer	Output Shape
<i>Input (left)</i>	(56, 56, 3)
<i>Input (right)</i>	(56, 56, 3)
<i>Base network</i>	(128)
<i>Lambda</i>	(1)
<i>Output</i>	(1)

Gambar 3.4 menunjukkan diagram alir pemrosesan data Fungsi 3 menggunakan SNN:



Gambar 3.4 Pemrosesan data Fungsi 3

Pemrosesan data Fungsi 3 dimulai dengan memasukkan *dataset* Fungsi 3 yang kemudian dilanjutkan dengan tahap *preprocessing* data. Setelah itu, dilakukan

pengembangan model SNN dan dilakukan pengujian terhadap model SNN yang telah dibuat. Terakhir, dilakukan evaluasi menggunakan *Confusion matrix*. Jika hasil evaluasi baik, proses pemrosesan selesai. Namun, jika hasil evaluasi tidak baik, proses kembali ke tahap pengembangan model SNN untuk dilakukan perbaikan atau penyempurnaan.

Dengan tahap pemrosesan data ini, data yang telah terkumpul dan diolah akan menghasilkan informasi yang akan digunakan untuk analisis lebih lanjut.

### 3.3.5 Penganalisan Data

Setelah pemrosesan data selesai, langkah selanjutnya adalah menganalisis informasi yang dihasilkan, yang mencakup perbandingan dengan hipotesis awal yang telah disusun sebelumnya.

- a. Analisis Data Fungsi 1. Setelah data gambar e-KTP diproses dengan menggunakan teknologi *Tesseract* OCR untuk mengubahnya menjadi teks, langkah selanjutnya adalah tahap analisis. Analisis akan dilakukan untuk mengukur sejauh mana teknologi *Tesseract* OCR mampu menghasilkan teks yang akurat dan relevan dengan data gambar e-KTP.
- b. Analisis Data Fungsi 2. Setelah data foto selfie diproses dengan menggunakan *Convolutional Neural Network* (CNN) untuk mengklasifikasikan gambar. Setelah melalui pelatihan model dan pengujian, hasil analisis akan membandingkan performa masing-masing model (model khusus CNN, model *transfer learning* VGG16, dan model *transfer learning* ResNet50). Analisis juga akan melihat seberapa baik fungsi aktivasi (*softmax* dan *sigmoid*) digunakan dalam konteks ini untuk menghasilkan prediksi yang akurat.
- c. Analisis Data Fungsi 3. Setelah data foto pada gambar e-KTP dan foto selfie diproses dengan menggunakan *Siamese Neural Network* (SNN) untuk membandingkan kesamaan antar input. Hasil pemrosesan ini akan dianalisis dengan mengukur sejauh mana SNN mampu mengenali dan mencocokkan kesamaan antara dua input (foto pada gambar e-KTP dan foto selfie).

Analisis data pada setiap fungsi memberikan pemahaman mengenai sejauh mana efektivitas teknik dan model yang diterapkan dalam konteks tugas yang bersangkutan.

### 3.3.6 Evaluasi

Setelah mengimplementasikan ketiga fungsi, dilakukan evaluasi untuk mengukur performanya. Dari hasil evaluasi yang kuat, dapat diidentifikasi area-area yang berkinerja baik dan aspek-aspek yang mungkin perlu perbaikan.

- a. Evaluasi Fungsi 1. Untuk mengevaluasi fungsi 1 digunakan *Word Error Rate* (WER) yang mengukur perbedaan antara teks hasil OCR dan teks asli pada gambar. Semakin rendah nilai WER, semakin akurat dan relevan teks hasil OCR dibandingkan dengan teks asli.
- b. Evaluasi Fungsi 2. Untuk mengevaluasi fungsi 2 digunakan *Confusion Matrix*. *Confusion matrix* merupakan tabel yang menunjukkan seberapa baik model mampu mengklasifikasikan data ke dalam kelas yang benar atau salah. Dari *confusion matrix*, dapat dihitung berbagai metrik evaluasi seperti akurasi, presisi, recall, dan F1-score untuk setiap model. Metrik-metrik ini memberikan pemahaman yang lebih dalam tentang kinerja model dalam mengenali foto selfie.
- c. Evaluasi Fungsi 3. Untuk mengevaluasi fungsi 3 digunakan *Confusion matrix*. *Confusion matrix* ini memberikan informasi tentang akurasi dan kemampuan SNN dalam mengukur kesamaan foto pada gambar e-KTP dan foto selfie. *Confusion matrix* membantu dalam memahami sejauh mana SNN mampu membandingkan kedua jenis gambar tersebut.

### 3.3.7 Penulisan Kesimpulan

Penulisan kesimpulan adalah tahap akhir pada penelitian ini. Kesimpulan adalah pernyataan singkat dan jelas yang biasanya berisi mengenai inti dari suatu penelitian atau hasil dari suatu analisis.