

**PENERAPAN *ARTIFICIAL INTELLIGENCE* DALAM APLIKASI
PEMBELAJARAN SKETSA DENGAN *AUGMENTED
REALITY* MENGGUNAKAN ALGORITMA
*CONVOLUTIONAL NEURAL NETWORK***

SKRIPSI

Diajukan Sebagai Salah satu Syarat
Untuk Memperoleh Gelar Sarjana Jenjang Strata Satu (S1)
Pada Program Studi Informatika Fakultas Teknik
Universitas Muhammadiyah Ponoogo



M.B. GIGIH BASKORO ASHARI

21533482

**PROGRAM STUDI INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS MUHAMMADIYAH PONOROGO**

2025

HALAMAN PENGESAHAN

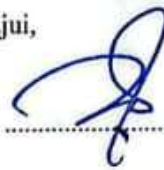
Nama : M.B. Gigih Baskoro Ashari
NIM : 21533482
Program Studi : Teknik Informatika
Fakultas : Teknik
Judul Skripsi : Penerapan *Artificial Intelligence* Dalam Aplikasi Pembelajaran Sketsa Dengan *Augmented Reality* Menggunakan Algoritma *Convolutional Neural Network*

Isi dan formatnya telah disetujui dan dinyatakan memenuhi syarat
Untuk mengikuti ujian skripsi
Pada Program Studi Teknik Informatika Fakultas Teknik Universitas
Muhammadiyah Ponorogo

Ponorogo, 14 Agustus 2025

Menyetujui,

Arin Yuli Astuti, S.Kom., M.Kom
NIK. 19890717 201309 13
(Dosen Pembimbing Utama)

: 

Ismail Abdurrozzaq Z., S.Kom, M.Kom
NIK. 19880728 201804 13
(Dosen Pembimbing Pendamping)

: 

Mengetahui,



Dekan Fakultas Teknik,

Edy Kurniawan, S.T, M.T
NIK. 19771026 200810 12

Ketua Program Studi Teknik Informatika,



Adi Fajaryanto Cobantoro, S.Kom, M.Kom
NIK. 19840924 201309 13

PERNYATAAN ORISINALITAS SKRIPSI

Yang bertanda tangan dibawah ini :

Nama : M.B. Gigih Baskoro Ashari

NIM : 21533482

Program Studi : Teknik Informatika

Dengan ini menyatakan bahwa Skripsi saya dengan judul : “Penerapan *Artificial Intelligence* Dalam Aplikasi Pembelajaran Sketsa Dengan *Augmented Reality* Menggunakan Algoritma *Convolutional Neural Network*” bahwa berdasarkan hasil penelusuran berbagai karya ilmiah, gagasan dan masalah ilmiah yang saya rancang/ teliti di dalam Naskah Skripsi ini adalah asli dari pemikiran saya. Tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis dikutip dalam naskah ini dan disebutkan dalam sumber kutipan dan daftar pustaka.

Apabila ternyata di dalam Naskah Skripsi ini dapat dibuktikan terdapat unsur – unsur plagiatisme, saya bersedia Ijazah saya dibatalkan, serta diproses sesuai dengan peraturan perundang-undang yang berlaku.

Demikian pernyataan ini dibuat dengan sesungguhnya dan dengan sebenar-benarnya.

Ponorogo, 14 Agustus 2025

Mahasiswa,



M.B. Gigih Baskoro Ashari

NIM 21533482

HALAMAN BERITA ACARA UJIAN

Nama : M.B. Gigih Baskoro Ashari
NIM : 21533482
Program Studi : Teknik Informatika
Fakultas : Teknik
Judul Skripsi : Penerapan *Artificial Intelligence* Dalam Aplikasi Pembelajaran Sketsa Dengan *Augmented Reality* Menggunakan Algoritma *Convolutional Neural Network*

Telah diuji dan dipertahankan dihadapan

Dosen penguji tugas akhir jenjang Strata Satu (S1) pada :

Hari : Senin
Tanggal : 28 Juli 2025

Dosen Penguji

Arin Yuli Astuti, S.Kom., M.Kom
NIK. 19890717 201309 13
(Ketua Penguji)

: 

Yovi Litanianda, S.Pd, M.Kom
NIK. 19810221 201309 13
(Anggota Penguji I)

: 

Ghulam Asrofi Buntoro, S.T., M.Eng
NIK. 19870723 202109 12
(Anggota Penguji II)

: 

Mengetahui,



Dekan Fakultas Teknik,

Edy Kurniawan, S.T, M.T
NIK. 19771026 20081012

Ketua Program Studi Teknik Informatika,

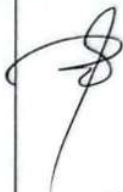
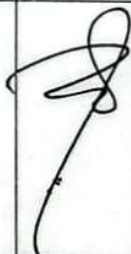


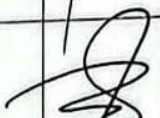
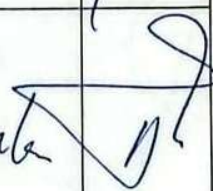
Adi Fajaryanto Cobantoro, S.Kom, M.Kom
NIK. 19840924 201309 13

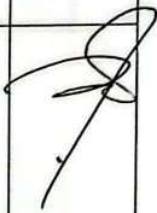
**BERITA ACARA
BIMBINGAN SKRIPSI**

Nama : M.B. GIGIH BASKORD ASHARI
 NIM : 21533482
 Judul Skripsi : Penerapan Artificial Intelligence Dalam Aplikasi Pembelajaran Sketsa Dengan ~~Algoritma~~ Augmented Reality Menggunakan Algoritma Convolutional Neural Network
 Dosen Pembimbing Utama : ARIN YULI ASTUTI, S.Kom., M.Kom.

PROSES PEMBIMBINGAN

No	Tanggal	Materi Yang Dikonsultasikan	Saran Pembimbing / Hasil	Tanda Tangan
1	16/24/09	Bab I	Isi lebih dalam sesuai	
2	31/10/24	Bab I	Perbaiki topik masalah dan pemilihan obyek.	
3	18/11/24	Bab I, II	Revisi	
4	9/12/24	Bab I, II	Revisi rumusan, Tugan bukan - Teori ditambahkan.	

No	Tanggal	Materi Yang Dikonsultasikan	Saran Pembimbing / Hasil	Tanda Tangan
5	13/24 12	Bab II	Revisi bab II	
6	20/24 12	Bab II	Acc bab II	
7	7/24 11	Mab III	Revisi bab III	
8	13/1 2015	III	- X Nama Hake	
9	12/1 2015	IV	Jabul final	
10	14/1 2015	I - IV	ACC sampul	

No	Tanggal	Materi Yang Dikonsultasikan	Saran Pembimbing / Hasil	Tanda Tangan
11	14/25/03	Bab III	Revisi bab III, IV	
12	05/25/05	Bab IV	Revisi bab IV	
13	13/25/06	Bab IV	Revisi algoritma	
14	09/25/07	Bab IV	Revisi Algoritma, Sistem, Rumus Pengujian disekuiler	
15	14/25/07	Bab I ✓ 1	Revisi bab I, IV	
16	18/25/07	Bab ✓	Simpulan + Abstrak	

No	Tanggal	Materi Yang Dikonsultasikan	Saran Pembimbing / Hasil	Tanda Tangan
17	18/25/07	All Sidang	All Sidang	
18				
19				
20				
21				
22				

BERITA ACARA BIMBINGAN SKRIPSI

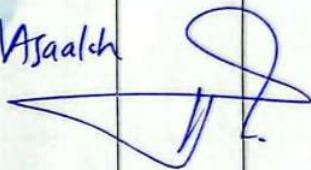
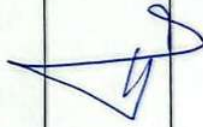
Nama : M.B. GIGIH BASKORO ASHARI

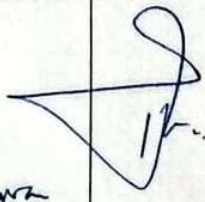
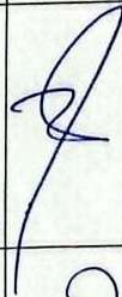
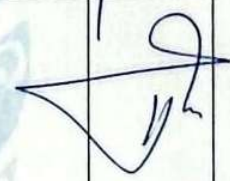
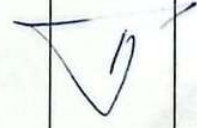
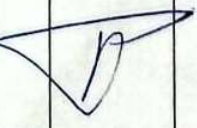
NIM : 21533482

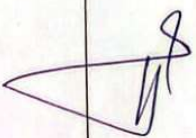
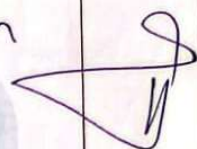
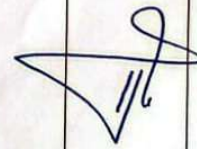
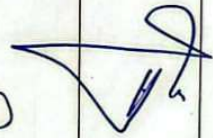
Judul Skripsi : Penerapan Artificial Intelligence Dalam Aplikasi pembelajaran sketsa dengan Augmented Reality menggunakan Algoritma Convolutional Neural Network

Dosen Pembimbing Pendamping : ISMAIL ABDURRAZZAQ Z, S.Kom, M.Kom

PROSES PEMBIMBINGAN

No	Tanggal	Materi Yang Dikonsultasikan	Saran Pembimbing / Hasil	Tanda Tangan
1	1/11 2024	Pengantar Tesis	Isi Abstrak	
2	5/11 2024	Bab I	- Cover - R-Masalah - Halaman - Layar	
3	12/12 2024	Bab I	- Layar Bab 2	
4	16/12 2024	Bab I	- Tabel - Paragraf - Sub Bab - Rumus	

No	Tanggal	Materi Yang Dikonsultasikan	Saran Pembimbing / Hasil	Tanda Tangan
5	9/1 2025	Bab <u>II</u>	- bab <u>II</u> Gln - revisi terlampir - cek revisi di diagram	
6	13/25 01	Acc Simpd	Acc simpd	
7	5/5 25	Bab <u>V</u>	Lengkap	
8	7/7 2025	Bab <u>IV</u>	- Plancher sesuai - Perbaikan	
9	7/7 2025	Bab <u>IV</u>	- Revisi Implanasi - Perbaikan	
10	11/7 2025	Bab <u>IV</u>	- Revisi Basor sesuai - Perbaikan	

No	Tanggal	Materi Yang Dikonsultasikan	Saran Pembimbing / Hasil	Tanda Tangan
11	14/7 2020	Bab <u>IV</u> , <u>V</u>	- Revisi Pengantar - Pengantar Keampuhan	
12	18/7 2020	Bab <u>V</u>	- Revisi Bab <u>V</u> , Keampuhan Farah	
13	20/7 2020	Demo Aplikasi	App	
14	24/7 2020	Solusi Berkas	Artikel	.
15	25/7 2020	Plagiasi Skripsi	Cek plagiasi	
16	25/7 2020	Bab <u>I</u> - <u>V</u>	ALL ready	

HASIL PLAGIASI SKRIPSI



UNIVERSITAS MUHAMMADIYAH PONOROGO
LEMBAGA LAYANAN PERPUSTAKAAN

Jalan Budi Utomo No. 10 Ponorogo 63471 Jawa Timur Indonesia
Telp. (0352) 481124, Fax (0352) 461796, e-mail : lib@umpo.ac.id
website : www.library.umpo.ac.id
TERAKREDITASI A
(SK Nomor 000137/ LAP.PT/ III.2020)
NPP. 3502102D2014337

SURAT KETERANGAN HASIL *SIMILARITY CHECK* KARYA ILMIAH MAHASISWA UNIVERSITAS MUHAMMADIYAH PONOROGO

Dengan ini kami nyatakan bahwa karya ilmiah ilmiah dengan rincian sebagai berikut :

Nama : M.B. Gigih Baskoro Ashari
NIM : 21533482
Judul : Penerapan Artificial Intelligence Dalam Aplikasi Pembelajaran Sketsa dengan Augmented Reality menggunakan Algoritma Convolutional Neural Network
Fakultas / Prodi : Teknik Informatika

Dosen pembimbing :

1. Arin Yuli Astuti S.Kom.,M.Kom
2. Ismail Abdurrazzaq Z., S.Kom.,M.Kom

Telah dilakukan check plagiasi berupa **Skripsi** di Lembaga Layanan Perpustakaan Universitas Muhammadiyah Ponorogo dengan prosentase kesamaan sebesar **14 %**

Demikian surat keterangan dibuat untuk digunakan sebagaimana mestinya.

Ponorogo, 21-08-2025
Kepala Lembaga Layanan Perpustakaan



Yolan Priatna, S.IIP., M.A
NIK. 1992052820220921

NB: Dosen pembimbing dimohon untuk melakukan verifikasi ulang terhadap kelengkapan dan keaslian karya beserta hasil cek Turnitin yang telah dilakukan

HASIL PLAGIASI JURNAL



UNIVERSITAS MUHAMMADIYAH PONOROGO
LEMBAGA LAYANAN PERPUSTAKAAN
Jalan Budi Utomo No. 10 Ponorogo 63471 Jawa Timur Indonesia
Telp. (0352) 481124, Fax (0352) 461796, e-mail : lib@umpo.ac.id
website : www.library.umpo.ac.id
TERAKREDITASI A
(SK Nomor 000137/ LAP.PT/ III.2020)
NPP. 3502102D2014337

SURAT KETERANGAN HASIL SIMILARITY CHECK KARYA ILMIAH MAHASISWA UNIVERSITAS MUHAMMADIYAH PONOROGO

Dengan ini kami nyatakan bahwa karya ilmiah ilmiah dengan rincian sebagai berikut :

Nama : M.B. Gigih Baskoro Ashari
NIM : 21533482
Judul : Application of Artificial Intelligence in Sketch Learning with Augmented Reality Using Convolutional Neural Network Algorithm
Fakultas / Prodi : Teknik Informatika

Dosen pembimbing :

1. Arin Yuli Astuti S.Kom.,M.Kom
2. Ismail Abdurrazzaq Z., S.Kom.,M.Kom

Telah dilakukan check plagiasi berupa **Jurnal** di Lembaga Layanan Perpustakaan Universitas Muhammadiyah Ponorogo dengan prosentase kesamaan sebesar **16 %**

Demikian surat keterangan dibuat untuk digunakan sebagaimana mestinya.

Ponorogo, 21-08-2025
Kepala Lembaga Layanan Perpustakaan



Yolan Priatna, S.IIP., M.A
NIK. 1992052820220921

NB: Dosen pembimbing dimohon untuk melakukan verifikasi ulang terhadap kelengkapan dan keaslian karya beserta hasil cek Turnitin yang telah dilakukan

MOTTO

“Kemandirian menumbuhkan keteguhan, pengkhianatan mengajarkan kewaspadaan, dan ilmu membuka jalan menuju kebebasan.”

~ Bintoro



HALAMAN PERSEMBAHAN

Segala puji syukur penulis panjatkan ke hadirat Allah Subhanahu wa Ta'ala atas rahmat, taufik, dan hidayah-Nya sehingga penulisan skripsi ini dapat terselesaikan dengan baik. Shalawat serta salam semoga senantiasa tercurah kepada junjungan kita Nabi Muhammad Shallallahu 'Alaihi Wasallam, suri teladan sepanjang masa yang menjadi penerang jalan dalam menuntut ilmu dan menjalani kehidupan. Dengan penuh rasa hormat dan kerendahan hati, karya sederhana ini penulis persembahkan kepada:

1. Ayahanda dan Ibunda tercinta, atas segala doa, kasih sayang, pengorbanan, serta dukungan moral maupun material yang tiada henti diberikan.
2. Seluruh keluarga besar, yang senantiasa menjadi sumber motivasi dan penguat dalam menempuh perjalanan akademik ini.
3. Ibu Arin Yuli Astuti, S.Kom., M.Kom. dan Bapak Ismail Abdurrozaq Z., S.Kom., M.Kom., selaku dosen pembimbing, atas bimbingan, arahan, serta kesabaran dalam mendampingi penulis hingga penelitian ini terselesaikan.
4. Amanda Zulfi Kurnia Tsani, atas dukungan, doa, dan semangat yang diberikan selama proses penulisan karya ilmiah ini.
5. Diri penulis sendiri, M.B. Gigih Baskoro Ashari, sebagai bentuk penghargaan atas usaha, ketekunan, dan kerja keras dalam menyelesaikan tanggung jawab akademik ini.

Penulis menyadari bahwa skripsi ini masih jauh dari kata sempurna. Oleh karena itu, kritik dan saran yang membangun sangat diharapkan demi penyempurnaan karya ini. Semoga skripsi ini dapat memberikan manfaat, khususnya dalam pengembangan ilmu di bidang Teknik Informatika, serta menjadi amal jariyah bagi penulis dan pembacanya.

**PENERAPAN *ARTIFICIAL INTELLIGENCE* DALAM APLIKASI
PEMBELAJARAN SKETSA DENGAN *AUGMENTED
REALITY* MENGGUNAKAN ALGORITMA
*CONVOLUTIONAL NEURAL NETWORK***

M.B.Gigih Baskoro Ashari, Arin Yuli Astuti, Ismail Abdurrozzaq Zulkarnain
Program Studi Teknik Infomatika, Fakultas Teknik, Universitas Muhammadiyah

Ponorogo

e-mail: mbgigihbaskoroashari@gmail.com

Abstrak

Pembelajaran Seni sketsa seringkali memberikan tantangan bagi pemula, karena keterbatasan dalam memahami bentuk, dan proporsionalitas sebuah gambar. Metode konvensional, seperti buku panduan gambar dan petunjuk langsung, seringkali tidak memberikan umpan balik yang efektif. Penelitian ini mengembangkan aplikasi IMAJI, sebuah sistem yang dirancang menggunakan metode Prototype dan dikembangkan bersama algoritma *Convolutional Neural Network* (CNN) arsitektur *MobileNetV2* yang diimplementasikan melalui Tensorflow Lite. Aplikasi memanfaatkan teknologi *ARCore* untuk menampilkan *overlay* pola sketsa di atas media gambar nyata, serta menghitung nilai *Mean Squared Error* (MSE) untuk mengevaluasi tingkat kesesuaian antara hasil gambar pengguna dan *template*. Metode *White Box* digunakan untuk menguji sistem. Hasil implementasi menunjukkan bahwa sistem mampu mengklasifikasikan gambar sketsa dengan sangat akurat ke dalam dua belas label objek dan tahapan gambar. Umpan balik secara langsung yang ditampilkan berdasarkan nilai MSE berhasil membantu pengguna dalam mengetahui kesalahan gambar. Integrasi AR dan CNN dalam aplikasi ini terbukti mampu meningkatkan efektivitas serta interaktivitas dalam proses pembelajaran sketsa.

Kata Kunci : *Sketsa, Artificial Intelligence, Augmented Reality, Convolutional Neural Network, MobileNetV2.*

KATA PENGANTAR

Assalamu'alaikum Warahmatullahi Wabarakaatuh.

Segala puji dan syukur penulis panjatkan ke hadirat Allah SWT atas rahmat, taufik, dan hidayah-Nya, sehingga skripsi dengan judul “Penerapan *Artificial Intelligence* Dalam Aplikasi Pembelajaran Sketsa Dengan *Augmented Reality* Menggunakan Algoritma *Convolutional Neural Network*”.

Tersusunnya skripsi ini tidak terlepas dari dukungan, bimbingan, dan bantuan berbagai pihak. Oleh karena itu, penulis menyampaikan terima kasih yang sebesar-besarnya kepada:

1. Bapak Edy Kurniawan, S.T., M.T. selaku Dekan Fakultas Teknik Universitas Muhammadiyah Ponorogo.
2. Bapak Adi Fajaryanto Cobantoro, S.Kom., M.Kom., selaku Kaprodi Teknik Informatika Universitas Muhammadiyah Ponorogo..
3. Ibu Arin Yuli Astuti, S.Kom., M.Kom. selaku pembimbing 1, dan Bapak Ismail Abdurrozzaq Z., S.Kom., M.Kom. selaku pembimbing 2, atas bimbingan dan arahan selama penyusunan skripsi..
4. Bapak Yovi Litanianda, S.Pd., M.Kom. dan Bapak Ghulam Asrofi Buntoro, S.T., M.Kom. selaku penguji 1 dan 2, atas masukan dan saran berharga.
5. Kedua orang tua tercinta beserta keluarga besar, atas doa, kasih sayang, dukungan, dan semangat yang tiada henti dalam setiap langkah penulis hingga terselesaikannya skripsi ini.
6. Amanda Zulfi Kurnia Tsani yang senantiasa memberikan semangat, doa, serta dukungan penuh selama proses penulisan skripsi.
7. Teman-teman Teknik Informatika Angkatan 2021 khususnya kelas C, atas kebersamaan dan bantuan yang diberikan.
8. Cokro Sketsa dan Bapak Kukuh Purwo Abdi Laksono sebagai tempat penelitian dan narasumber.

Wassalamualaikum Warahmatullahi Wabarakaatuh.

Ponorogo, 19 Agustus 2025

Penulis,

M.B. Gigih Baskoro Ashari

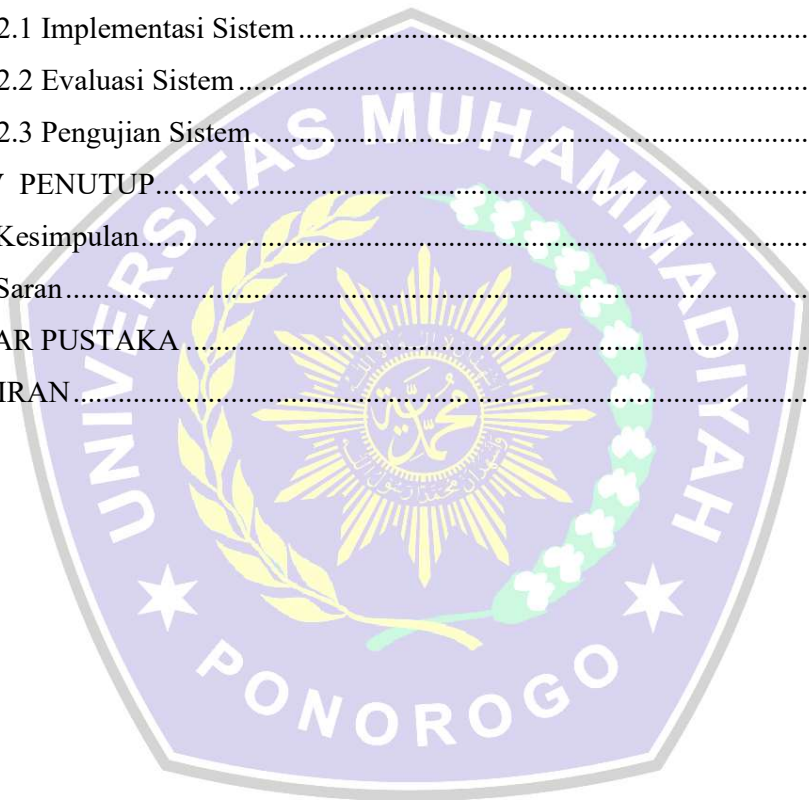
NIM. 21533482

DAFTAR ISI

HALAMAN PENGESAHAN.....	ii
PERNYATAAN ORISINALITAS SKRIPSI	iii
HALAMAN BERITA ACARA UJIAN	iv
BERITA ACARA BIMBINGAN SKRIPSI	v
HASIL PLAGIASI SKRIPSI.....	xii
HASIL PLAGIASI JURNAL	xiii
MOTTO	xiv
HALAMAN PERSEMBAHAN	xv
ABSTRAK	xvi
KATA PENGANTAR	xvii
DAFTAR ISI.....	xviii
DAFTAR GAMBAR	xxi
DAFTAR TABEL	xxiii
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	3
1.3 Batasan Masalah.....	3
1.4 Tujuan Penelitian.....	3
1.5 Manfaat Penelitian.....	4
BAB II TINJAUAN PUSTAKA.....	5
2.1 Penelitian Terdahulu.....	5
2.2 Landasan Teori	9
2.2.1 <i>Artificial Intelligence Feedback</i> (Umpan Balik AI)	9
2.2.2 <i>Convolutional Neural Network (CNN)</i>	11
2.2.3 <i>Augmented Reality (AR)</i>	14
2.2.4 Sketsa	15
2.2.5 <i>Python</i>	16
2.2.6 <i>Confusion Matrix</i>	16
2.2.7 <i>TensorFlow Lite</i>	17

2.2.8 Kotlin.....	17
2.2.9 AR Sceneform	18
2.2.10 Visual Studio Code	19
2.2.11 Android Studio	19
2.2.12 Database	19
2.2.13 Datasets.....	20
2.2.14 Data Flow Diagram (DFD)	20
2.2.15 Entity Relationship Diagram (ERD)	21
2.2.16 Use Case Diagram	22
2.2.17 Flowchart	24
2.2.18 Perancangan Sistem	25
2.2.19 Pengembangan Sistem	25
2.2.20 Pengujian Sistem.....	27
BAB III METODE PENELITIAN.....	29
3.1 Tahapan Penelitian	29
3.1.1 Identifikasi Masalah	30
3.1.2 Analisis Kebutuhan	30
3.1.3 Pengumpulan Data	31
3.1.4 Objek Penelitian	32
3.2 Perancangan Sistem.....	33
3.2.1 Flowchart Sistem.....	33
3.2.2 Diagram Use Case.....	34
3.2.3 Data Flow Diagram (DFD)	36
3.2.4 Entity Relationship Diagram (ERD).....	39
3.2.5 Perancangan Antarmuka	42
3.2.6 Perancangan Database.....	43
3.2.7 Pembagian Dataset.....	46
3.2.8 Implementasi Algoritma.....	47
3.2.9 Pengintegrasian Model CNN.....	49
3.2.10 Penerapan CNN untuk AI Feedback.....	50
3.3 Evaluasi Sistem.....	51

3.4 Pengujian Sistem	52
BAB IV ANALISA DATA DAN PEMBAHASAN	56
4.1 Analisis Data	56
4.1.1 Pengolahan <i>Dataset</i>	56
4.1.2 Arsitektur <i>model</i> CNN	58
4.1.3 Pelatihan <i>Model</i> CNN	60
4.1.4 Integrasi <i>Model</i> CNN	62
4.2 Pembahasan	63
4.2.1 Implementasi Sistem.....	64
4.2.2 Evaluasi Sistem.....	67
4.2.3 Pengujian Sistem.....	71
BAB V PENUTUP.....	87
5.1 Kesimpulan.....	87
5.2 Saran.....	87
DAFTAR PUSTAKA	89
LAMPIRAN.....	94



DAFTAR GAMBAR

Gambar 2.1 Cara kerja CNN <i>MobileNetV2</i>	12
Gambar 2.2 Confusion Matrix	16
Gambar 2.3 Alur Metode Prototype.....	25
Gambar 2.4 Pengujian WhiteBox	27
Gambar 3.1 Alur Tahapan Penelitian.....	29
Gambar 3.2 <i>Flowchart Sistem</i>	33
Gambar 3.3 <i>Use Case Diagram</i>	34
Gambar 3.4 DFD Level 0.....	36
Gambar 3.5 DFD level 1	37
Gambar 3.6 Perancangan ERD	39
Gambar 3.7 Antarmuka Sistem.....	42
Gambar 3. 8 Antarmuka Sistem (2)	43
Gambar 3.9 Alur Implementasi Algoritma	47
Gambar 3.10 Perancangan Arsitektur <i>Model CNN MobileNetV2</i>	48
Gambar 4.1 Contoh Gambar didalam Dataset	57
Gambar 4.2 Arsitektur Model CNN <i>MobilNetV2</i>	59
Gambar 4.3 Hasil Log dari epoch	61
Gambar 4.4 Grafik Akurasi Training Loss CNN <i>MobileNetV2</i>	61
Gambar 4.5 Pilih template latihan (Beranda).....	64
Gambar 4.6 Galeri progres pengguna	65
Gambar 4.7 Halaman Profil Pengguna.....	66
Gambar 4.8 Halaman Utama Admin.....	67
Gambar 4.9 <i>Confusion Matrix CNN MobileNetV2</i>	68
Gambar 4.10 Hasil Classification Report.....	69
Gambar 4.11 Pengujian AR-overlay	72
Gambar 4.12 Hasil Analisa gambar pengguna.....	76
Gambar 4.13 Perbandingan <i>Template</i> dengan Hasil.....	78
Gambar 4.14 Hasil fitur <i>AI feedback</i>	80

Gambar 4.15 Implementasi Mode Manual.....	82
Gambar 4.16 Hasil gambar dan upload gambar pengguna	83
Gambar 4.17 <i>Upload template</i> baru (<i>Admin</i>).....	85



DAFTAR TABEL

Tabel 2.1 Penelitian Terdahulu	5
Tabel 2.2 Atribut DFD	20
Tabel 2.3 Atribut ERD	21
Tabel 2.4 Atribut <i>Use Case Diagram</i>	23
Tabel 2.5 Atribut <i>Flowchart</i>	24
Tabel 3.1 Struktur tabel <i>user</i>	44
Tabel 3.2 Struktur tabel <i>admin</i>	44
Tabel 3.3 Struktur tabel <i>template_gambar</i>	44
Tabel 3.4 Struktur tabel <i>hasil_gambar</i>	45
Tabel 3.5 Pembagian <i>Dataset</i>	46
Tabel 3.6 Skenario Pengujian Sistem	53
Tabel 4.1 Parameter Pelatihan	60
Tabel 4.2 Pengujian fitur <i>Overlay</i>	72
Tabel 4.3 Hasil contoh Prediksi 3 kelas	75
Tabel 4.4 Hasil pengujian Inferensi	75
Tabel 4.5 Contoh nilai piksel	77
Tabel 4.6 Hasil pengujian MSE	78
Tabel 4.7 Hasil Pengujian <i>AI Feedback</i>	79
Tabel 4.8 Hasil Pengujian Mode Manual	81
Tabel 4.9 Hasil Pengujian fitur <i>Upload Hasil</i>	83
Tabel 4.10 Hasil Pengujian <i>Upload Template</i>	84
Tabel 4.11 Hasil pengujian Sistem	85

BAB I

PENDAHULUAN

1.1 Latar Belakang

Pembelajaran seni sketsa seringkali menjadi tantangan bagi pemula karena membutuhkan keterampilan teknis dan visual yang kuat untuk menghasilkan gambar yang akurat. Dalam lingkungan pendidikan tradisional, instruksi sering terbatas pada buku atau demonstrasi langsung, yang tidak selalu efektif dalam membantu siswa memahami konsep proporsi, perspektif atau bayangan secara mendalam. Oleh karena itu, diperlukan pendekatan yang lebih interaktif dan dinamis untuk meningkatkan kualitas pembelajaran seni sketsa. Hal tersebut memungkinkan siswa belajar dengan cara yang lebih praktis dan langsung.[1]

Salah satu bentuk pembelajaran alternatif yang sudah berkembang di masyarakat adalah komunitas seni, seperti Cokro Sketsa Ponorogo. Komunitas ini menjadi wadah bagi pegiat seni sketsa untuk belajar bersama dan mengasah keterampilan mereka. Namun, pendekatan komunitas ini masih memiliki keterbatasan dalam memberikan *Feedback* objektif dan terstruktur bagi anggotanya, terutama bagi pemula yang ingin belajar secara mandiri tanpa selalu mengandalkan mentor. Komunitas ini didirikan oleh Kukuh Purwo Abdi Laksono bersama beberapa seniman lainnya yang sebagian besar merupakan lulusan dari Institut Seni Indonesia (ISI) Jogjakarta. Di tengah dominasi seni pertunjukan seperti tari dan teater di Ponorogo, Cokro Sketsa hadir untuk memperkenalkan dan mengembangkan seni rupa di kota Ponorogo. Saat ini, komunitas yang memiliki sekitar 15 anggota ini rutin berkumpul dan melakukan aktivitas menggambar setiap Selasa malam, dengan sketsa yang menggambarkan suasana kota, seperti tata letak, ornamen, hingga aktivitas warga di sepanjang jalan HOS Cokroaminoto. [2], [3]

Meskipun komunitas seni seperti Cokro Sketsa Ponorogo memberikan wadah belajar, pendekatan ini masih memiliki keterbatasan dalam memberikan *Feedback* yang terstruktur, cepat, dan objektif kepada para anggotanya. Oleh karena itu, diperlukan solusi aplikatif berbasis teknologi yang dapat membantu pemula dalam memahami kesalahan mereka secara langsung dan mendapatkan panduan

yang lebih sistematis. Dalam penelitian ini, dikembangkan Aplikasi Imaji, sebuah aplikasi pembelajaran seni sketsa berbasis mobile yang mengintegrasikan teknologi *Augmented Reality* (AR) dan *Artificial Intelligence* (AI) untuk memandu proses belajar. Dengan integrasi teknologi ini, pengguna dapat langsung mengetahui kesalahan dalam gambar mereka, dan mendapat saran perbaikan untuk meningkatkan keterampilan mereka secara lebih cepat dan akurat. Teknologi ini juga memungkinkan pengalaman belajar yang lebih personal dan langsung, mengatasi keterbatasan dari metode instruksi tradisional. Namun, untuk mengantisipasi ketidakstabilan pada sistem AR akibat perubahan pencahayaan atau pergerakan kamera, disediakan juga mode manual sebagai alternatif. Dalam mode ini, pengguna tetap dapat melihat pola gambar secara statis dan melanjutkan proses belajar tanpa hambatan teknis.[4]

Untuk menerapkan solusi ini, metode *Convolutional Neural Network* (CNN) digunakan sebagai algoritma utama dalam sistem *Artificial Intelligence* (AI) yang akan diterapkan pada pembelajaran sketsa. CNN memiliki kemampuan untuk mengenali pola dan detail dalam gambar dengan presisi yang tinggi, yang memungkinkan aplikasi ini untuk mendeteksi kesalahan dalam gambar sketsa dan memberikan umpan balik yang tepat. Selain itu, teknologi *Augmented Reality* (AR) akan digunakan untuk menampilkan pola gambar langsung di media gambar yang sedang dikerjakan oleh pengguna. Meskipun AR berperan sebagai perantara untuk menampilkan pola gambar, fokus utama dari penelitian ini adalah pada penerapan AI dengan CNN, yang akan mendeteksi dan memberikan umpan balik terhadap gambar yang dibuat oleh pengguna.[5]

Dengan demikian, tema yang diusulkan dalam penelitian ini adalah **Penerapan *Artificial Intelligence* dalam Aplikasi Pembelajaran Sketsa dengan *Augmented Reality* Menggunakan Algoritma *Convolutional Neural Network*.** Dengan pendekatan ini, diharapkan dapat mengatasi tantangan dalam pembelajaran seni sketsa, mempercepat proses belajar, dan meningkatkan keterampilan menggambar para pengguna secara lebih efektif.

1.2 Rumusan Masalah

Rumusan masalah dari penelitian ini adalah sebagai berikut:

1. Bagaimana teknologi *Artificial Intelligence* (AI) dapat membantu pengguna menggambar sesuai dengan pola sketsa?
2. Bagaimana *Augmented Reality* (AR) dapat berintegrasi dengan algoritma *Convolutional Neural Network* (CNN) untuk memberikan panduan visual dalam proses menggambar sketsa?

1.3 Batasan Masalah

Batasan masalah diperlukan untuk menentukan ruang lingkup penelitian dan agar penelitian dapat dilakukan secara terfokus. Batasan masalah untuk penelitian ini adalah

1. Aplikasi yang dikembangkan akan berfokus pada pembelajaran seni sketsa menggunakan teknologi *Artificial Intelligence* dengan *Augmented Reality* dan *Convolutional Neural Network* (CNN).
2. Pola menggambar yang tersedia sudah ditentukan dan disajikan dalam bentuk *template* yang dapat diikuti pengguna.
3. Deteksi kesalahan dan umpan balik otomatis akan terbatas pada aspek dasar, seperti kesesuaian sketsa dengan pola yang telah ditentukan.
4. Penerapan aplikasi akan diujicobakan pada komunitas CokroSketsa.

1.4 Tujuan Penelitian

Adapun tujuan dari penelitian ini adalah sebagai berikut:

1. Merancang sistem berbasis *Artificial Intelligence* (AI) yang mampu membantu pengguna dalam menggambar sesuai dengan pola sketsa.
2. Mengimplementasikan integrasi antara *Augmented Reality* (AR) dan algoritma *Convolutional Neural Network* (CNN) untuk memberikan panduan visual dalam proses menggambar sketsa.

1.5 Manfaat Penelitian

Manfaat yang diperoleh dari penelitian ini adalah sebagai berikut:

1. Aplikasi ini dirancang untuk membuat pembelajaran seni sketsa lebih interaktif dan menarik.
2. Integrasi *Artificial Intelligence* (AI) dan *Convolutional Neural Network* (CNN) memungkinkan deteksi kesalahan sketsa secara otomatis.
3. Integrasi *Augmented Reality* (AR) dan *Convolutional Neural Network* (CNN) memungkinkan aplikasi untuk mendeteksi dan menyesuaikan konten berdasarkan jenis media sketsa yang digunakan.



BAB II

TINJAUAN PUSTAKA

2.1 Penelitian Terdahulu

Pada bagian ini, akan disajikan tinjauan terhadap penelitian-penelitian terdahulu yang relevan dengan topik penelitian ini. Tinjauan ini mencakup perkembangan teori serta hasil penelitian terbaru yang memberikan landasan yang kuat bagi penelitian yang sedang dilakukan.

Tabel 2. 1 Penelitian Terdahulu

No	Judul	Bahasan	Pembeda	Hasil Penelitian
1.	<i>“Hand-drawn sketch recognition with a double-channel convolutional neural network”</i> , 2021 [6]	Pengenalan tangan menggunakan jaringan konvolusional dual-channel. Metode ini melibatkan proses preprocessing untuk mendapatkan sketsa yang halus dan kontur sketsa, kemudian menggunakan CNN dua kanal untuk ekstraksi fitur dan klasifikasi. Hasil eksperimen menunjukkan peningkatan tingkat pengenalan tangan secara efektif. Kerja lanjutan akan difokuskan pada meningkatkan akurasi	Penelitian ini memberikan penekanan yang signifikan pada identifikasi sketsa tangan menggunakan metode <i>Convolutional Neural Network</i> (CNN).	Metode ini secara signifikan meningkatkan tingkat pengenalan dibandingkan dengan metode pengenalan sketsa lainnya, seperti HOG-SVM, <i>SIFT-Fisher</i> , dan <i>AlexNet</i> . Secara khusus, algoritma ini menunjukkan peningkatan akurasi hingga 5,1% dibandingkan dengan <i>AlexNet</i> , serta keunggulan dalam waktu pelatihan dan konsumsi energi

		pengenalan input tulisan tangan.	
2.	"A Review of Neural Network Approac h on Enginee ring Drawin g", 2020 [7]	Membahas tentang Fokus pada pengenalan elemen analisis gambar teknik, jaringan saraf, menggunakan jaringan sedangkan khususnya CNN, saraf. Tinjauan ini penelitian ini menunjukkan hasil fokus pada peran akan yang sangat baik dalam jaringan saraf dalam mengembangk pengenalan elemen meningkatkan deteksi an aplikasi elemen gambar teknik simbol dan pengenalan pembelajaran dibandingkan dengan elemen gambar teknik. AR untuk metode tradisional Hasilnya meningkatkan keterampilan jaringan saraf dapat menggambar. meningkatkan akurasi pengenalan elemen gambar teknik secara efektif.	Penelitian menemukan bahwa penggunaan jaringan saraf, CNN, menunjukkan hasil yang sangat baik dalam pengenalan elemen gambar teknik dibandingkan dengan metode tradisional
3.	"Ar-powered educatio nal applicat ion based upon convolut ional neural	Membahas tentang pentingnya menggunakan jaringan saraf konvolusional (CNN) dalam aplikasi pendidikan berbasis AR untuk menyediakan proses belajar dengan visualisasi yang impresif. Penelitian ini	Membahas aplikasi ini mampu meningkatkan keterlibatan siswa dan pemahaman materi pelajaran. Penggunaan AR membuat pembelajaran lebih menarik dan menyenangkan.

network menunjukkan bahwa AR untuk
s", 2020 CNN dapat meningkatkan
 [8] meningkatkan akurasi keterampilan
 lokasi dan menggambar.
 pengenalan objek
 dalam sistem AR,
 sehingga aplikasi ini
 dapat digunakan untuk
 analisis data besar dan
 penelitian *real-time*.
 Hasil penelitian ini
 menunjukkan bahwa
 penggunaan CNN
 dalam aplikasi AR
 dapat meningkatkan
 kualitas belajar dan
 interaksi siswa secara
 signifikan.

4. "*Student training program of building drawing and Augmented Reality in SMKN 2*" Membahas tentang Berfokus pada Program pelatihan ini program pelatihan program berhasil meningkatkan siswa yang difokuskan pelatihan pengetahuan dan pada teknik gambar siswa, keterampilan siswa bangunan dan realitas sedangkan dalam bidang gambar tambahan di SMKN 2 penelitian ini teknik dan *Augmented Reality*. *Feedback* dari Program ini bertujuan mengembangk peserta juga untuk meningkatkan an aplikasi menunjukkan respons keterampilan keras pembelajaran positif terhadap siswa dengan AR untuk kegiatan ini, yang SMKN melibatkan pelatihan meningkatkan dianggap bermanfaat 2 teknik gambar untuk pengembangan
-

	Pangkal pinang", 2020 [9]	bangunan untuk keterampilan menggambar. Model Bangunan dan Desain Informasi, serta pelatihan terkait realitas tambahan untuk Departemen Multimedia. Hasilnya adalah produksi karya-karya desain gambar bangunan dan produk realitas tambahan.		kemampuan mereka di bidang konstruksi dan multimedia
5.	"Developing Feedback Literacy Capabilities through an Automated Feedback Tool", 2020[4]	Membahas tentang penggunaan alat umpan balik otomatis berbasis AI untuk meningkatkan kemampuan literasi umpan balik siswa. Penelitian ini akan menunjukkan bahwa alat umpan balik otomatis dapat membantu siswa memahami, menghargai, dan mengambil tindakan atas umpan balik, yang merupakan komponen penting dalam literasi umpan balik. Hasil	Merujuk pada kemampuan pengembangan umpan balik, literasi umpan balik, sedangkan penelitian ini mengembangkan aplikasi pembelajaran AR yang menggunakan CNN dan memanfaatkan umpan balik dari AI untuk membantu siswa	membantu siswa mengakui umpan balik sebagai proses timbal balik, memungkinkan mereka untuk lebih kritis dan reflektif terhadap umpan balik yang diberikan. Dengan menghilangkan nilai penilaian dari interaksi umpan balik, siswa menjadi lebih mampu mengevaluasi umpan balik secara objektif, yang berkontribusi pada pengembangan penilaian evaluatif mereka.

penelitian ini meningkatkan
menunjukkan bahwa keterampilan
penggunaan alat menggambar.
umpan balik otomatis
dapat meningkatkan
kemampuan siswa
dalam
mengaktualisasikan
umpan balik dan
meningkatkan kualitas
belajar.

Berdasarkan temuan-temuan tersebut, penelitian ini menggabungkan keunggulan CNN untuk pengenalan sketsa dan teknologi umpan balik otomatis berbasis AI dalam satu sistem pembelajaran sketsa yang inovatif. Berbeda dengan penelitian sebelumnya, fokus penelitian ini adalah pada pengembangan aplikasi yang tidak hanya mendukung proses pengenalan sketsa secara real-time, tetapi juga memberikan umpan balik otomatis yang dapat meningkatkan keterampilan menggambar siswa. Hal ini bertujuan untuk menciptakan pendekatan pembelajaran yang lebih efektif, personal, dan interaktif, sehingga mendukung tercapainya hasil belajar yang optimal.

2.2 Landasan Teori

2.2.1 *Artificial Intelligence Feedback (Umpan Balik AI)*

Umpan Umpan balik AI merupakan salah satu fitur utama dalam sistem pembelajaran berbasis teknologi yang memungkinkan interaksi otomatis antara pengguna dan aplikasi. Dalam konteks aplikasi pembelajaran seni sketsa, AI digunakan untuk menganalisis goresan pensil pengguna dan memberikan umpan balik real-time. Sistem ini dapat mendeteksi ketidaksesuaian antara pola sketsa yang telah ditentukan dengan gambar yang dibuat oleh pengguna, seperti kesalahan pada garis, proporsi, atau bentuk. Dengan demikian, AI membantu pengguna dalam

memperbaiki teknik menggambar mereka tanpa memerlukan instruktur langsung.[4], [10]

Untuk memberikan umpan balik yang tepat dan akurat, AI memanfaatkan berbagai metrik untuk mengevaluasi hasil gambar pengguna. Salah satu metrik utama adalah *Mean Squared Error* (MSE), yang mengukur perbedaan antara gambar yang dihasilkan oleh pengguna dengan pola yang telah ditentukan dalam aplikasi. Selain itu, *accuracy* digunakan untuk menilai sejauh mana hasil prediksi oleh *model* sesuai dengan pola yang diinginkan. Metrik-metrik ini berperan penting dalam memberikan umpan balik yang berguna dan relevan untuk membantu pengguna memperbaiki kesalahan dan meningkatkan keterampilan menggambar mereka.[4] [11]

Berikut adalah rumus *Mean Squared Error* (MSE) digunakan untuk mengukur kesalahan antara output yang diprediksi dan nilai sebenarnya

$$MSE = \frac{1}{n \cdot 3} \sum_{i=1}^n [(R_i - R'_i)^2 + (G_i - G'_i)^2 + (B_i - B'_i)^2] \div 225 \quad (2.1)$$

Keterangan

- a. *MSE*: *Mean Squared Error*, nilai rata-rata kuadrat dari kesalahan prediksi.
- b. *n*: Jumlah piksel (lebar × tinggi gambar)
- c. *R_i, G_i, B_i* : Nilai sebenarnya pada data ke-*i*.
- d. *R'_i, G'_i, B'_i*: Nilai prediksi pada data ke-*i*.
- e. 3: Faktor karena terdapat tiga kanal warna (RGB)
- f. 225: Normalisasi ke rentang 0–1 (karena RGB bernilai 0–255)

Penerapan AI dalam aplikasi pembelajaran seni sketsa menawarkan banyak keuntungan. Salah satunya adalah kemampuan untuk memberikan umpan balik secara otomatis tanpa perlu keterlibatan instruktur, sehingga mempercepat proses pembelajaran. AI akan secara menunjukkan kesalahan-kesalahan dalam gambar pengguna, baik dalam hal proporsi, garis, atau elemen sketsa lainnya, dan memberikan saran perbaikan yang memungkinkan pengguna untuk memperbaiki gambar mereka secara *real-time*. [12]

Selain itu, dengan menggunakan teknik pembelajaran mesin seperti *Convolutional Neural Network* (CNN), sistem AI dapat meningkatkan

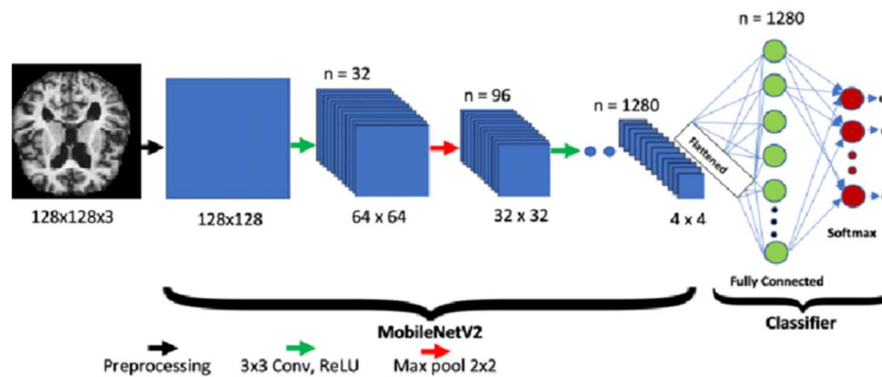
kemampuannya dalam mendeteksi dan memahami pola dalam gambar. CNN berperan untuk memfasilitasi proses pengenalan pola dan kesalahan pada gambar yang dibuat oleh pengguna, yang selanjutnya digunakan untuk memberikan umpan balik yang lebih akurat.

Secara keseluruhan, penerapan umpan balik AI pada aplikasi pembelajaran seni sketsa ini meningkatkan kualitas pengajaran, memungkinkan pengguna untuk lebih cepat mengidentifikasi dan memperbaiki kesalahan mereka, serta mempercepat kemajuan mereka dalam menggambar sketsa.

2.2.2 Convolutional Neural Network (CNN)

Convolutional Neural Network (CNN) adalah jenis jaringan saraf tiruan yang dirancang untuk mengolah data visual seperti gambar. CNN mampu mengenali pola-pola visual seperti garis, bentuk, dan tekstur secara otomatis, menjadikannya sangat efektif digunakan dalam berbagai tugas pengolahan citra, termasuk klasifikasi gambar, deteksi objek, hingga segmentasi area tertentu pada gambar.[34]

Dalam penelitian ini, digunakan arsitektur CNN yang disebut *MobileNetV2*. *MobileNetV2* merupakan salah satu *model* deep learning yang dirancang khusus untuk perangkat mobile. Keunggulan dari *MobileNetV2* terletak pada efisiensinya yang tinggi. *Model* ini memiliki jumlah parameter yang jauh lebih sedikit dibandingkan CNN konvensional, namun tetap mampu memberikan akurasi yang baik dalam pengenalan gambar. Hal ini menjadikannya sangat cocok untuk aplikasi real-time berbasis Android seperti aplikasi pembelajaran seni sketsa. [13]



Gambar 2.1 Cara kerja CNN *MobileNetV2*

MobileNetV2 merupakan pengembangan dari MobileNet yang memperkenalkan dua komponen utama yakni Inverted Residual Blocks dan Linear Bottlenecks. Secara umum, proses kerja *MobileNetV2* diawali dengan memasukkan citra sketsa berukuran tetap sebagai input dengan tiga kanal warna (RGB). Input ini kemudian melewati serangkaian lapisan konvolusional yang berfungsi mengekstraksi fitur spasial, seperti kontur, orientasi garis, dan struktur lokal lainnya. *MobileNetV2* tidak menggunakan konvolusi standar secara langsung, melainkan mengadopsi pendekatan efisien bernama depthwise separable convolution, yang memisahkan proses konvolusi menjadi dua tahap: depthwise convolution dan pointwise convolution (1×1 convolution).

Pada tahap depthwise convolution, setiap kanal input dikonvolusikan secara terpisah menggunakan satu filter, sehingga proses ini fokus pada spasialitas intra-channel. Kemudian, hasilnya digabung dan diproses oleh pointwise convolution yang bertugas menggabungkan informasi antar kanal (inter-channel) melalui operasi konvolusi 1×1 . Pendekatan ini secara drastis mengurangi jumlah parameter dan kompleksitas komputasi tanpa menurunkan performa *model* secara signifikan, menjadikannya sangat sesuai untuk deployment di perangkat mobile dengan sumber daya terbatas.

Setelah tahap ekstraksi fitur selesai, tensor hasil ekstraksi biasanya berdimensi tinggi diringkas melalui proses global average *pooling*, yang merata-ratakan setiap *channel* fitur menjadi satu nilai. Hal ini tidak hanya mereduksi dimensi data, tetapi juga membantu mencegah *overfitting*. Selanjutnya, tensor hasil

pooling diteruskan ke lapisan fully connected (Dense layer) sebagai representasi vektor yang siap untuk proses klasifikasi.

Di bagian output, digunakan fungsi aktivasi softmax yang mengubah vektor logit (output mentah dari Dense layer) menjadi distribusi probabilitas yang ternormalisasi. Softmax memastikan bahwa total skor prediksi pada semua kelas bernilai 1, sehingga memungkinkan pemilihan kelas dengan probabilitas tertinggi sebagai hasil prediksi *model*. Secara sistematis fungsi Softmax dirumuskan sebagai berikut:

$$\sigma(z_i) = \frac{e^{z_i}}{\sum_{j=1}^K e^{z_j}} \quad (2.2)$$

Keterangan:

- a. $\sigma(z_i)$: probabilitas prediksi untuk kelas ke- i
- b. z_i : logit untuk neuron ke- i
- c. K : jumlah total kelas yang diprediksi *model*.

Nilai probabilitas tertinggi yang dihasilkan oleh fungsi softmax akan dianggap sebagai label prediksi akhir, dan dalam konteks sistem ini, nilai tersebut ditampilkan sebagai confidence score untuk menunjukkan tingkat keyakinan *model* terhadap prediksi yang diberikan.

Dalam konteks aplikasi pembelajaran sketsa, *MobileNetV2* berperan penting dalam menganalisis hasil gambar yang dibuat oleh pengguna. *Model* ini dapat mengenali apakah gambar sketsa sudah sesuai dengan pola yang ditentukan, misalnya apakah proporsinya sudah benar atau apakah garis-garisnya sudah sejajar dan tepat. Jika ditemukan kesalahan, sistem akan memberikan umpan balik otomatis untuk membantu pengguna memperbaiki gambar mereka. Dengan pendekatan ini, pengguna tidak hanya belajar menggambar, tetapi juga mendapatkan arahan yang akurat dan langsung dari sistem berbasis kecerdasan buatan.

Integrasi *MobileNetV2* dalam sistem pembelajaran berbasis AI dan *Augmented Reality* juga memberikan pengalaman belajar yang lebih menarik dan interaktif. Panduan visual yang ditampilkan secara real-time membuat pengguna dapat memahami pola sketsa dengan lebih mudah, sekaligus memperbaiki

kesalahan mereka secara langsung. Hal ini diharapkan mampu meningkatkan efektivitas proses belajar dan membantu pengguna mengembangkan keterampilan menggambar secara lebih cepat dan menyenangkan.

2.2.3 *Augmented Reality (AR)*

Augmented Reality (AR) adalah teknologi yang memungkinkan integrasi elemen digital dengan dunia nyata melalui perangkat seperti *smartphone* atau gawai. Teknologi AR memanfaatkan kamera untuk mengenali lingkungan sekitar dan menambahkan elemen virtual seperti objek 2D, 3D, animasi atau teks yang seolah-olah hadir di dunia fisik. Manfaat utama AR adalah kemampuannya untuk meningkatkan pengalaman pengguna dengan menyediakan visualisasi yang intuitif dan interaktif.[14], [15]

Salah satu media pembelajaran yang dapat digunakan untuk membuat siswa aktif melalui pengalaman langsung dan menarik minat belajar mereka adalah *Augmented Reality (AR)*. AR merupakan teknologi yang memungkinkan penggabungan antara dunia nyata dan virtual dengan menampilkan objek tiga dimensi (3D) di dunia nyata melalui media kamera. Hal ini menciptakan ilusi bahwa objek 3D tersebut hadir di lingkungan fisik, sehingga AR juga dapat menampilkan ilustrasi yang sulit diwujudkan secara konkret. [14], [15]

Komponen utama dalam AR adalah objek 3D yang muncul ketika marker dipindai oleh aplikasi. Interaksi dengan objek 3D yang dianimasikan ini dapat meningkatkan ketertarikan siswa dalam belajar. Kelebihan AR meliputi efektivitas penggunaannya, aplikasi dalam berbagai bentuk media, kemudahan pengoperasian, biaya pembuatan yang relatif rendah, serta interaktivitas yang lebih tinggi dibandingkan media Pembelajaran lainnya. [16]

Teknologi *Augmented Reality (AR)* memungkinkan aplikasi pembelajaran sketsa untuk menciptakan pengalaman belajar yang interaktif dan imersif, di mana siswa dapat berinteraksi langsung dengan objek virtual yang ditampilkan di lingkungan nyata. Integrasi AR dalam aplikasi pembelajaran sketsa tidak hanya memperkaya pengalaman visual tetapi juga memudahkan pengguna dalam memahami konsep menggambar yang kompleks. Dengan panduan visual yang

jelas, pengguna dapat mengikuti pola sketsa dengan lebih mudah, sehingga meningkatkan keterampilan menggambar mereka.[16]

2.2.4 Sketsa

Sketsa adalah bentuk seni gambar yang menggunakan garis-garis sederhana untuk menggambarkan objek atau ide secara cepat dan tanpa detail berlebihan. Pada dasarnya, sketsa berfungsi sebagai kerangka kasar atau panduan awal dalam karya seni visual, membantu seniman mengungkapkan komposisi, proporsi, dan bentuk dasar suatu objek sebelum masuk ke tahap lebih detail.[17], [18]

a. Proporsi

Proporsi adalah teknik untuk memastikan ukuran bagian objek yang digambar saling sesuai. Dalam sketsa, proporsi membantu menciptakan keseimbangan dan realisme. Aplikasi ini akan membantu pengguna memahami proporsi dengan menyediakan panduan atau pola yang dapat diikuti, memungkinkan latihan mengukur dan menyeimbangkan elemen gambar.[18]

b. Perspektif

Perspektif memberikan efek kedalaman pada gambar dua dimensi, dengan objek dekat terlihat lebih besar dan yang jauh lebih kecil. Aplikasi ini akan mengajarkan perspektif melalui pola atau garis bantu untuk membantu pengguna menciptakan sketsa yang lebih hidup dan realistis.[18], [19]

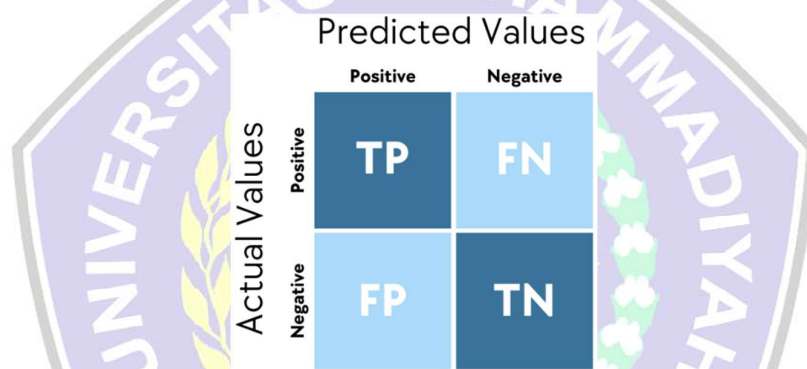
c. Garis

Garis adalah elemen dasar dalam sketsa yang digunakan untuk membentuk dan mendefinisikan objek. Aplikasi akan mendeteksi ketepatan garis dan memberikan umpan balik otomatis jika garis pengguna tidak sesuai dengan pola, membantu meningkatkan kontrol garis dalam sketsa.[18], [19]

2.2.5 Python

Python adalah bahasa pemrograman yang menggunakan interpreter untuk menjalankan kodenya. Interpreter ini mampu menerjemahkan kode secara langsung, memungkinkan *Python* untuk dijalankan di berbagai platform seperti *Windows*, *Linux*, dan lainnya. *Python* mengadopsi berbagai paradigma pemrograman dari beberapa bahasa lain, termasuk pemrograman prosedural seperti *C*, pemrograman berorientasi objek seperti *Java*, dan pemrograman fungsional seperti *Lisp*. Kombinasi dari berbagai paradigma ini memudahkan para programmer dalam mengembangkan berbagai proyek menggunakan *Python*. [20]

2.2.6 Confusion Matrix



		Predicted Values	
		Positive	Negative
Actual Values	Positive	TP	FN
	Negative	FP	TN

Gambar 2. 2 *Confusion Matrix*

Empat komponen utama dalam confusion matrix meliputi, True Positive (TP) Prediksi benar untuk kelas yang sesuai. False Positive (FP) Prediksi salah ketika *model* mengklasifikasikan data ke kelas yang salah. False Negative (FN): *Model* gagal mengenali data dari kelas yang seharusnya. True Negative (TN): Prediksi benar bahwa data bukan berasal dari kelas tertentu. Komponen ini menjadi dasar dalam mengevaluasi performa *model* serta memahami jenis kesalahan yang sering terjadi.

Dalam konteks pengembangan aplikasi IMAJI, label dikategorikan berdasarkan jenis objek (anjing, kucing, ikan) dan tahap menggambar (step 1 hingga step 4). Oleh karena itu, confusion matrix yang dihasilkan memuat 12 kelas. Melalui analisis confusion matrix, dapat diketahui kelas-kelas mana yang sering

tertukar. Misalnya, sistem terkadang salah memprediksi gambar “kucing step 2” sebagai “anjing step 2”, yang dapat disebabkan oleh kemiripan bentuk visual antar objek atau kekurangan data pada kelas tertentu

2.2.7 *TensorFlow Lite*

TensorFlow Lite adalah *framework* pembelajaran mesin yang dirancang untuk membawa kemampuan kecerdasan buatan (AI) ke dalam perangkat dengan proses pemrograman yang terbatas, seperti *Smartphone* atau perangkat digital lainnya. *Framework* ini menawarkan keunggulan dalam hal efisiensi, dengan ukuran *model* yang ringan dan optimalisasi untuk prosesor *ARM*, *GPU*, maupun *Neural Processing Unit* (NPU). Salah satu manfaat utama *TensorFlow Lite* adalah kemampuannya untuk menjalankan inferensi *model* pembelajaran mesin secara lokal tanpa memerlukan koneksi internet, sehingga mengurangi latensi dan meningkatkan privasi data pengguna. [20]

Dalam konteks pengembangan sistem berbasis AI, *framework* ini sangat berguna untuk mendukung pengenalan pola melalui *model Convolutional Neural Network* (CNN) secara real-time pada perangkat seluler. Kemampuannya memungkinkan pemrosesan data gambar langsung dari kamera perangkat untuk memberikan hasil analisis yang cepat dan akurat, menjadikannya pilihan ideal untuk aplikasi yang memadukan teknologi pembelajaran mesin dan interaksi pengguna secara langsung.[20]

2.2.8 *Kotlin*

Kotlin merupakan bahasa pemrograman modern yang andal dan kini menjadi standar utama dalam pengembangan aplikasi Android. Kotlin dikenal dengan sintaksisnya yang ringkas, aman terhadap *null* (*null safety*), serta mendukung paradigma berorientasi objek dan fungsional. Dibandingkan dengan Java, Kotlin menawarkan efisiensi penulisan kode yang lebih tinggi dan integrasi yang lebih baik dengan berbagai pustaka modern. Salah satu keunggulan Kotlin adalah kemampuannya untuk bekerja mulus dengan *framework* dan API Android, termasuk untuk penerapan fitur berbasis *Augmented Reality* (AR) dan *Artificial*

Intelligence (AI). Dalam pengembangan aplikasi Android, Kotlin digunakan untuk mengelola logika inti aplikasi, komunikasi antar-komponen, serta integrasi *model* AI dan fitur antarmuka interaktif berbasis AR. Dengan dukungan komunitas yang kuat dan kompatibilitas penuh dengan ekosistem Android Studio, Kotlin memberikan fondasi yang efisien, stabil, dan modern bagi pengembangan aplikasi seperti IMAJI.[21]

2.2.9 *AR Sceneform*

AR Sceneform adalah pustaka (SDK) yang dikembangkan oleh Google untuk mempermudah pembuatan aplikasi *Augmented Reality* (AR) berbasis ARCore pada perangkat Android. Sceneform menyediakan antarmuka yang lebih ramah bagi pengembang dengan memungkinkan visualisasi objek 3D langsung dalam lingkungan Android Studio menggunakan bahasa pemrograman Java atau Kotlin, tanpa memerlukan penguasaan mendalam terhadap OpenGL.

Sceneform menyederhanakan proses pengembangan AR dengan menyediakan fitur-fitur seperti rendering objek 3D, penempatan objek virtual secara real-time, serta interaksi dengan permukaan fisik di dunia nyata melalui kamera perangkat. Dalam implementasinya, Sceneform bekerja di atas ARCore dengan memanfaatkan kemampuan pelacakan gerakan, deteksi permukaan (plane detection), dan estimasi pencahayaan untuk menjaga stabilitas dan akurasi elemen virtual yang ditampilkan.

Dalam konteks pengembangan aplikasi IMAJI, AR *Sceneform* digunakan untuk menampilkan panduan visual berupa overlay gambar atau pola sketsa di atas media gambar fisik (seperti kertas). Dengan bantuan teknologi ini, pengguna dapat melihat dan mengikuti langkah-langkah menggambar secara langsung melalui kamera perangkat, sehingga memberikan pengalaman belajar yang interaktif, presisi tinggi, dan imersif. Penggunaan *Sceneform* juga memungkinkan pengaturan posisi, rotasi, dan skala objek secara dinamis agar sesuai dengan media gambar di dunia nyata.[22]

2.2.10 *Visual Studio Code*

Visual Studio Code adalah editor teks yang kaya fitur, dirancang untuk mendukung berbagai bahasa pemrograman dan alat pengembangan. *Editor* ini menyediakan dukungan lintas platform, ekstensi yang dapat disesuaikan, debugging terintegrasi, dan kemampuan untuk mengelola kode dengan efisien. Manfaat utama *Visual Studio Code* adalah fleksibilitas dan kemampuannya untuk meningkatkan produktivitas pengembang melalui fitur seperti *intelliSense*, integrasi *Git*, dan terminal bawaan. Dalam proses pengembangan, alat ini menjadi platform yang kokoh untuk menulis, menguji, dan memelihara kode, serta mendukung pengelolaan proyek yang terorganisasi dengan baik.[23], [24]

2.2.11 *Android Studio*

Android Studio adalah *Integrated Development Environment* (IDE) resmi untuk mengembangkan aplikasi *Android*. Dilengkapi dengan fitur seperti editor kode yang canggih, emulator bawaan, dan alat analisis performa, *Android Studio* memungkinkan pengembang untuk merancang, membangun, dan menguji aplikasi dengan efisien. Manfaatnya meliputi kemudahan dalam membangun antarmuka pengguna, kompatibilitas dengan berbagai versi *Android*, dan kemampuannya untuk mendukung pengintegrasian pustaka eksternal. Sebagai alat utama untuk membangun aplikasi berbasis *Android*, IDE ini memberikan lingkungan kerja yang lengkap untuk pengembangan fitur-fitur kompleks.[25]

2.2.12 *Database*

Database adalah sistem yang digunakan untuk menyimpan, mengelola, dan mengakses data secara terstruktur. Dengan menggunakan database, data dapat disimpan dengan efisien dan diakses secara *real-time* sesuai kebutuhan. Manfaat utama *database* adalah organisasi data yang rapi, kemampuan untuk memproses kueri dengan cepat, dan skalabilitas yang memungkinkan pengelolaan data dalam jumlah besar. Dalam konteks aplikasi berbasis teknologi, *database* menjadi tulang punggung untuk menyimpan informasi pengguna, data aktivitas, dan berbagai aset digital lainnya dengan aman dan terstruktur.[25]

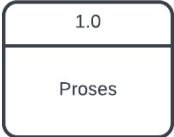
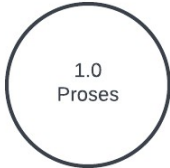
2.2.13 Datasets

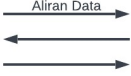
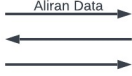
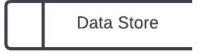
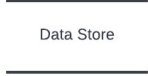
Datasets merujuk pada kumpulan data yang digunakan untuk melatih, menguji, atau mengevaluasi *model* dalam sistem berbasis pembelajaran mesin. *Dataset* dapat terdiri dari data numerik, gambar, teks, atau bentuk lainnya, tergantung pada kebutuhan aplikasi. Manfaat *Dataset* adalah kemampuannya untuk menyediakan data yang diperlukan untuk meningkatkan akurasi dan keandalan *model* yang dikembangkan. Dalam penerapan teknologi AI atau pembelajaran mesin, *Dataset* yang relevan memainkan peran krusial untuk membangun *model* yang sesuai dengan tujuan sistem. [13]

2.2.14 Data Flow Diagram (DFD)

Data Flow Diagram (DFD) adalah representasi grafik dari sebuah sistem. DFD menggambarkan komponen-komponen sebuah sistem, aliran-aliran data diantara komponen-komponen tersebut, asal, tujuan dan penyimpanan dari data tersebut. Gambar dibawah ini menunjukkan simbol yang digunakan dalam DFD baik dalam versi *E.Yourdan* dan *De Marco* maupun versi *Chris Gane* dan *Trish Sarson*. [26]

Tabel 2. 2 Atribut DFD

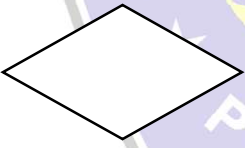
Gane/Sarson	Yourdon/De Marco	Keterangan
		Entitas eksternal dapat berupa orang/unit terkait yang berinteraksi dengan sistem tetapi diluar sistem
		Orang/ unit yang mempergunakan atau melakukan transformasi data. Komponen fisik tidak diidentifikasi.

		Aliran data dengan arah khusus dari sumber ke tujuan.
		Penyimpanan data atau tempat data dilihat oleh proses.

2.2.15 Entity Relationship Diagram (ERD)

Entity Relationship Diagram (ERD) adalah diagram berbentuk notasi grafis yang berada dalam pembuatan database yang menghubungkan antara data satu dengan yang lain. Fungsi ERD adalah sebagai alat bantu dalam pembuatan database dan memberikan gambaran bagaimana kerja database yang akan dibuat. Di dalam ERD terdapat 4 elemen dasar, yaitu entitas, atribut, relasi, dan garis.[27], [28]

Tabel 2. 3 Atribut ERD

Simbol	Keterangan
	Entitas, yaitu kumpulan dari objek yang dapat diidentifikasi secara unik
	Relasi, yaitu hubungan yang terjadi antara salah satu lebih entitas. Jenis hubungan antara lain, one to one, one to many, dan many to many.
	Atribut, yaitu karakteristik dari entitas atau relasi yang merupakan penjelasan detail tentang entitas.
	<i>Connector</i> , Hubungan antara entitas dengan atributnya dan himpunan entitas dengan himpunan relasinya.

a. Entitas

Entitas merupakan objek yang akan menjadi perhatian dalam suatu database. Entitas dapat berupa manusia, tempat, benda, atau kondisi mengenai data yang dibutuhkan. Simbol dari entitas berbentuk persegi panjang.

b. Relasi

Relasi di dalam ERD merupakan hubungan antara dua atau lebih entitas. Simbol dari relasi berbentuk belah ketupat. Relasi yang dapat dimiliki oleh ERD ada beberapa macam, yaitu:

1. *One to One*, Satu anggota entitas dapat berelasi dengan satu anggota entitas lain.
2. *One to Many*, Satu anggota entitas dapat berelasi dengan beberapa anggota entitas lain.
3. *Many to Many*, Beberapa anggota entitas dapat berelasi dengan beberapa anggota entitas lain.

c. Atribut

Atribut merupakan informasi yang terdapat dalam entitas. Sebuah entitas harus memiliki *Primary Key* sebagai ciri khas entitas dan atribut deskriptif. Atribut biasanya terletak dalam tabel entitas atau dapat juga terpisah dari tabel. Simbol dari atribut berbentuk elips.

d. *Connector*

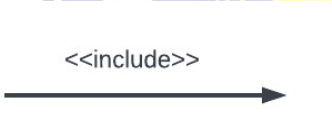
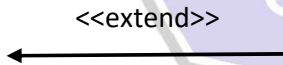
Hubungan antara entitas direpresentasikan dengan garis yang menghubungkan entitas tersebut.

2.2.16 Use Case Diagram

Use case merupakan deskripsi fungsi dari sebuah sistem dari perspektif atau sudut pandang para pengguna sistem. *Use case* mendefinisikan apa yang akan diproses oleh sistem dan komponen – komponennya. *Use case* bekerja dengan menggunakan scenario yang merupakan deskripsi dari urutan atau langkah –

langkah yang menjelaskan apa yang dilakukan oleh *User* terhadap sistem maupun sebaliknya. Berikut simbol atau atribut untuk *Use Case*:

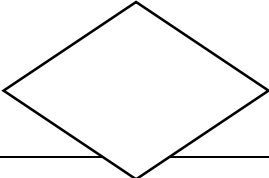
Tabel 2. 4 Atribut *Use Case Diagram*

Simbol	Keterangan
	Aktor, Mewakili Peran Orang, sistem yang lain, atau alat ketika berkomunikasi dengan <i>Use Case</i> .
	<i>Use Case</i> , Abstraksi dan interaksi antara aktor dengan use case.
	<i>Association</i> , Abstraksi dari penghubung antara aktor dengan <i>use case</i> .
	Generalisasi, spesialisasi aktor untuk dapat berpartisipasi dengan <i>use case</i> .
	Menunjukkan bahwa suatu <i>use case</i> seluruhnya merupakan fungsionalitas dari <i>use case</i> lainnya.
	Menunjukkan bahwa suatu <i>use case</i> seluruhnya merupakan fungsionalitas dari <i>use case</i> lainnya, jika suatu kondisi terpenuhi.

2.2.17 Flowchart

Flowchart adalah penggambaran secara grafik dari langkah-langkah dan urutan prosedur dari suatu program. *Flowchart* dapat dengan jelas menunjukkan Algoritma menjelaskan cara melakukan serangkaian operasi secara logis dan sistematis. [29] Berikut Atribut atau symbol-simbol Untuk Flowchart:

Tabel 2. 5 Atribut *Flowchart*

Simbol	Keterangan
	Permulaan/ Akhir program.
	Arah aliran Program.
	Instalasi Pemberian nilai awal.
	Proses Penghitungan/ pengolahan data.
	Input/ output data, parameter, informasi.
	Permulaan sebuah program/ proses menjalankan sub program.
	Perbandingan pernyataan, penyeleksian data yang memberikan pilihan untuk langkah selanjutnya.



Penghubung bagian-bagian *flowchart* yang berada di satu halaman.



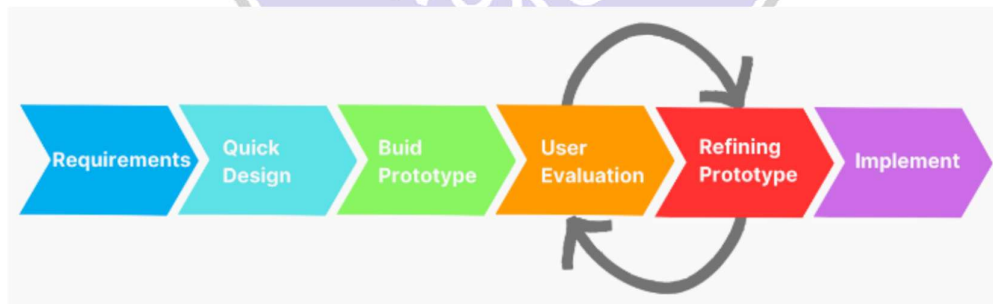
Penghubung bagian-bagian *flowchart* yang berada pada halaman yang berbeda.

2.2.18 Perancangan Sistem

Perancangan sistem dimulai dengan proses analisis kebutuhan, yang bertujuan untuk mengidentifikasi kebutuhan fungsional dan nonfungsional sistem. Proses ini menghasilkan dokumen spesifikasi kebutuhan yang menjadi acuan dalam perancangan. Selanjutnya, *model* sistem dirancang menggunakan alat bantu seperti *Data Flow Diagram* (DFD) untuk memvisualisasikan alur data.[30]

DFD membantu memahami bagaimana data mengalir melalui sistem, misalnya dari pengguna, ke server, hingga ke hasil keluaran. Selain itu, *Entity Relationship Diagram* (ERD) digunakan untuk mendefinisikan hubungan antar entitas dalam *database*. Dalam perancangan UI/UX, alat seperti *Figma* dimanfaatkan untuk membuat wireframe dan *prototype* desain. Desain UI/UX harus memperhatikan hirarki visual, aksesibilitas, dan navigasi yang intuitif.[30], [31]

2.2.19 Pengembangan Sistem



Gambar 2.3 Alur Metode *Prototype*

Seperti pada gambar 2.2 pengembangan sistem dalam penelitian ini menggunakan metode *prototype* yang dilakukan secara iteratif untuk memastikan

aplikasi memenuhi kebutuhan pengguna secara optimal. Proses dimulai dengan identifikasi masalah, di mana kendala dalam pembelajaran ilustrasi sketsa, seperti kebutuhan akan pola gambar berbasis AR, umpan balik otomatis menggunakan *Artificial Intelligence* (AI) dengan algoritma *Convolutional Neural Network* (CNN), serta antarmuka pengguna yang sederhana, dianalisis secara mendalam.[4], [32]

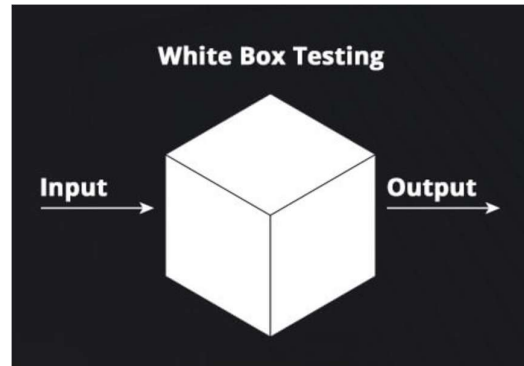
Setelah itu, dilakukan analisis kebutuhan untuk merumuskan spesifikasi sistem yang akan dikembangkan. Langkah ini mencakup identifikasi fitur utama dan pengumpulan *Dataset* gambar sketsa sebagai data pelatihan *model* CNN. Hasil analisis menjadi dasar untuk merancang *prototype* sistem, yang meliputi desain antarmuka pengguna (UI), struktur sistem, dan pembuatan wireframe untuk memvisualisasikan alur kerja aplikasi.[33]

Prototype awal kemudian dievaluasi melalui evaluasi *prototype*, melibatkan komunitas pengguna untuk mendapatkan umpan balik. Berdasarkan hasil evaluasi, *prototype* diperbaiki hingga mencapai desain yang sesuai dengan kebutuhan pengguna. Setelah disetujui, sistem mulai dikembangkan melalui pengkodean sistem menggunakan *framework TensorFlow*, Integrasi *ARCore* untuk mendukung fitur AR, dan *Firebase Firestore* sebagai *database*. [32]

Tahap berikutnya adalah evaluasi sistem, di mana kinerja *model* CNN dalam mengenali pola gambar diuji, termasuk keakuratan prediksi dan stabilitas aplikasi. Sistem juga menjalani pengujian untuk memastikan seluruh fungsionalitas berjalan dengan baik pada data uji. Jika sistem telah teruji, dilakukan penggunaan sistem, yaitu implementasi aplikasi secara penuh di lingkungan nyata untuk mendukung pembelajaran ilustrasi sketsa.[34]

Melalui pendekatan metode *prototype* ini, setiap tahapan dilakukan secara bertahap dan iteratif untuk menghasilkan aplikasi yang adaptif, terarah, dan sesuai kebutuhan pengguna, seperti yang digambarkan dalam diagram alur metode *prototype*.

2.2.20 Pengujian Sistem



Gambar 2. 4 Pengujian *WhiteBox*

Pengujian sistem dilakukan menggunakan metode *White Box Testing*, yang berfokus pada analisis struktur internal dan alur logika sistem untuk memastikan setiap bagian kode berfungsi sebagaimana mestinya. Metode ini dipilih untuk menguji bagaimana aplikasi dalam penelitian menangani pemrosesan data secara internal, terutama dalam algoritma *Convolutional Neural Network* (CNN), *rendering* AR, serta integrasi dengan database *PostgreSQL*. [25]

Secara umum, pengujian sistem dalam aplikasi ini melibatkan beberapa aspek teknis, yaitu:

- a. Analisis Struktur Kode, Dilakukan pemeriksaan terhadap modul pemrosesan gambar, algoritma CNN, serta fungsi ARCore yang menangani deteksi posisi kertas dan rendering objek secara real-time.
- b. Pembuatan Kasus Uji, Pengujian mencakup skenario seperti uji validasi input sketsa, respons CNN terhadap berbagai jenis gambar, serta akurasi penempatan objek AR di atas kertas.
- c. Pelaksanaan Pengujian, Setiap fungsi diuji dengan berbagai parameter, seperti resolusi gambar input, kontras sketsa, pencahayaan sekitar, dan ketepatan deteksi tanda silang pada kertas untuk memastikan stabilitas aplikasi.
- d. Analisis Hasil, Hasil output dibandingkan dengan ekspektasi sistem untuk menilai akurasi dan konsistensi. Setiap ketidaksesuaian dicatat untuk dianalisis lebih lanjut.

- e. Perbaikan dan Pengujian Ulang, Jika ditemukan kesalahan, seperti prediksi sketsa yang salah atau *overlay* gambar yang meleset, maka perbaikan diterapkan pada *model* AI atau AR Sceneform, lalu diuji ulang untuk memastikan stabilitas.

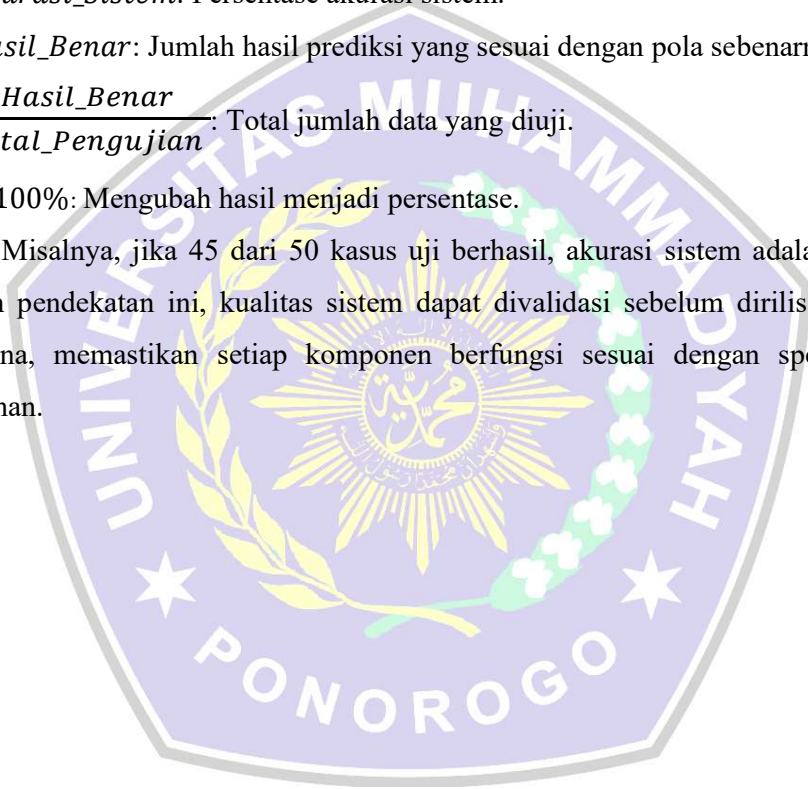
Akurasi sistem dihitung menggunakan rumus:

$$Akurasi_Sistem = \frac{Hasil_Benar}{Total_Pengujian} \times 100\% \quad (2.3)$$

Keterangan:

1. *Akurasi_Sistem*: Persentase akurasi sistem.
2. *Hasil_Benar*: Jumlah hasil prediksi yang sesuai dengan pola sebenarnya.
3. $\frac{Hasil_Benar}{Total_Pengujian}$: Total jumlah data yang diuji.
4. $\times 100\%$: Mengubah hasil menjadi persentase.

Misalnya, jika 45 dari 50 kasus uji berhasil, akurasi sistem adalah 90%. Dengan pendekatan ini, kualitas sistem dapat divalidasi sebelum dirilis kepada pengguna, memastikan setiap komponen berfungsi sesuai dengan spesifikasi kebutuhan.

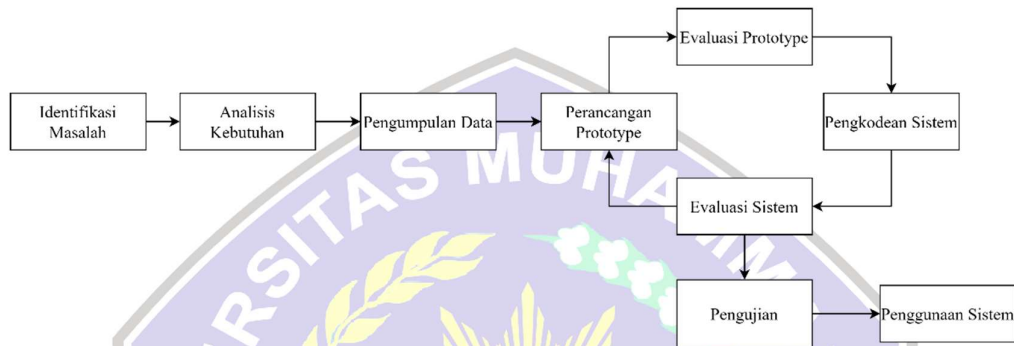


BAB III

METODE PENELITIAN

3.1 Tahapan Penelitian

Bab ini membahas metodologi penelitian dengan menggunakan metode *prototype*, yang terdiri dari beberapa tahapan utama, yaitu identifikasi masalah, analisis kebutuhan, pengumpulan data, perancangan *prototype*, evaluasi *prototype*, pengkodean sistem, evaluasi sistem, pengujian sistem, dan penggunaan sistem.



Gambar 3.1 Alur Tahapan Penelitian

Penelitian ini menggunakan metode *prototype* yang terdiri dari beberapa tahapan utama. Pada Gambar 3.1, Tahap pertama adalah identifikasi masalah untuk memahami kendala dalam pembelajaran ilustrasi sketsa dan menentukan kebutuhan sistem. Selanjutnya, dilakukan analisis kebutuhan dan pengumpulan *Dataset* sketsa sebagai data pelatihan *model*. *Prototype* sistem dirancang meliputi desain antarmuka pengguna (UI) dan struktur sistem, lalu dievaluasi untuk memastikan kesesuaiannya.

Setelah *prototype* disetujui, sistem dikembangkan melalui pengkodean menggunakan algoritma CNN dengan *framework TensorFlow* serta *Firestore* sebagai *database*. Sistem kemudian dievaluasi untuk mengukur kinerja *model*, diuji untuk memastikan fungsionalitas, dan diimplementasikan dalam lingkungan nyata untuk mendukung pembelajaran ilustrasi sketsa. Semua tahapan dirancang secara terstruktur untuk menghasilkan aplikasi pembelajaran berbasis AI yang solutif.

3.1.1 Identifikasi Masalah

Pada tahap ini, identifikasi masalah berfokus pada tantangan yang dihadapi dalam pembelajaran sketsa, terutama bagi pemula yang kesulitan menggambar pola dengan akurat. Metode pembelajaran tradisional seringkali terbatas pada buku atau demonstrasi manual, yang membuat pengguna kesulitan memahami proporsi, perspektif, dan komposisi garis. Keterbatasan ini mempengaruhi proses belajar karena pengguna tidak mendapatkan umpan balik secara langsung atas kesalahan yang dibuat.

Solusi yang ditawarkan adalah pengembangan aplikasi berbasis teknologi AI dan *Augmented Reality*. Aplikasi ini memungkinkan pengguna untuk menggambar dengan panduan visual secara *real-time* melalui *overlay* AR dan mendapatkan umpan balik otomatis menggunakan algoritma *Convolutional Neural Network* (CNN).

3.1.2 Analisis Kebutuhan

Analisis kebutuhan bertujuan untuk menentukan elemen-elemen penting yang diperlukan dalam pengembangan aplikasi pembelajaran sketsa berbasis AI dan AR. Pada tahap ini, kebutuhan perangkat lunak, perangkat keras, dan *Dataset* diidentifikasi untuk memastikan aplikasi dapat berjalan secara optimal. Dengan pemahaman yang mendalam terhadap kebutuhan sistem, proses pengembangan dapat diarahkan agar sesuai dengan tujuan dan harapan pengguna.

a. Perangkat Lunak

1. Sistem operasi Android untuk mendukung aplikasi yang dikembangkan.
2. *Framework TensorFlow* untuk implementasi CNN.
3. *ARCore* untuk mendukung teknologi *Augmented Reality*.
4. *Supabase* sebagai database untuk menyimpan data pengguna dan hasil analisis sketsa.
5. *Python* sebagai bahasa pemrograman utama untuk pelatihan *model* AI.
6. *Android Studio* untuk pengembangan antarmuka aplikasi.

b. Perangkat Keras

1. *Smartphone* Android dengan spesifikasi minimal: RAM 4GB, prosesor dengan kemampuan AI, dan kamera berkualitas tinggi untuk mendukung AR.
2. Laptop atau komputer dengan spesifikasi minimum: prosesor *intel core i5* atau setara, RAM 8GB, GPU dengan dukungan CUDA untuk pelatihan *model CNN*.

c. *Dataset*

Dataset yang digunakan terdiri dari kumpulan gambar sketsa dalam berbagai kategori, seperti hewan, bunga, dan bentuk geometris. Setiap kategori memiliki langkah-langkah pola (*step*) untuk memandu pengguna, seperti pola dasar, detail tambahan, outline, dan pewarnaan. *Dataset* ini harus diproses dengan teknik seperti *grayscale* atau deteksi tepi untuk meningkatkan efektivitas pelatihan *model*.

3.1.3 Pengumpulan Data

Pengumpulan data dilakukan untuk mendukung pengembangan aplikasi dengan informasi yang relevan dan akurat. Data diperoleh melalui berbagai metode, seperti studi literasi, wawancara, dan observasi, yang dirancang untuk memahami kebutuhan pengguna serta tantangan dalam pembelajaran sketsa. Proses ini bertujuan untuk menyediakan dasar yang kuat bagi desain dan implementasi aplikasi.

a. Studi Literasi

Dilakukan dengan membaca literatur dan penelitian sebelumnya tentang teknologi AI dan AR dalam pembelajaran, algoritma CNN untuk analisis gambar, serta aplikasi berbasis AR di bidang pendidikan.

b. Wawancara

Melibatkan wawancara dengan seniman, seorang pendiri komunitas CokroSketsa yang memiliki pengalaman mengajar menggambar. Wawancara bertujuan untuk memahami tantangan yang dihadapi pengguna dalam pembelajaran sketsa dan kebutuhan fitur yang relevan untuk aplikasi.

c. Observasi

Observasi dilakukan dengan melihat langsung bagaimana pemula belajar menggambar sketsa menggunakan metode tradisional. Tujuan observasi adalah mengidentifikasi kelemahan metode tradisional yang dapat diatasi dengan aplikasi berbasis teknologi.

3.1.4 Objek Penelitian

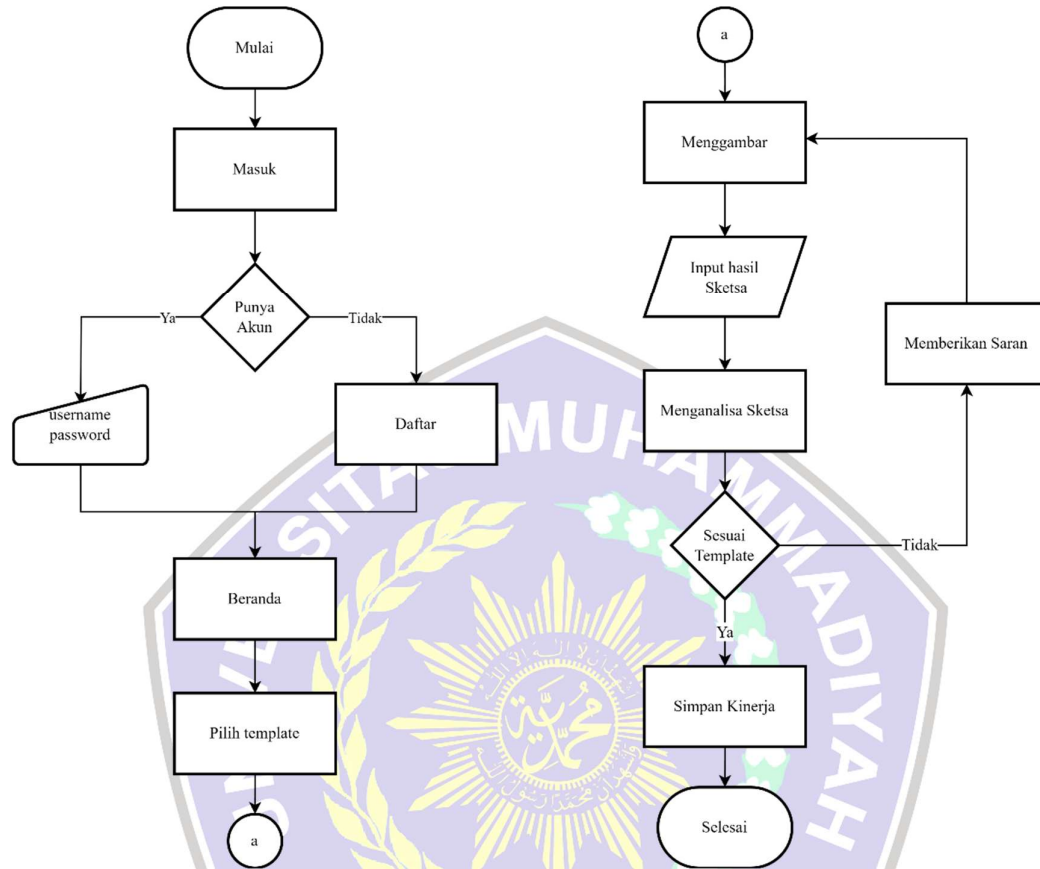
Objek penelitian ini adalah pengembangan aplikasi pembelajaran seni sketsa yang bertujuan untuk membantu pengguna dalam menggambar sketsa menggunakan teknologi *Artificial Intelligence* (AI) dan *Augmented Reality* (AR). Aplikasi ini menggunakan algoritma *Convolutional Neural Network* (CNN) untuk menganalisis dan mendeteksi kesalahan sketsa secara real-time, seperti ketidaksesuaian garis, proporsi, dan bentuk dibandingkan dengan *template* pola gambar yang telah ditentukan.

Augmented Reality (AR) berperan dalam menampilkan *template* gambar sketsa secara overlay di media gambar melalui kamera ponsel, sehingga pengguna dapat mengikuti panduan visual yang lebih interaktif. Selain itu, Integrasi AI melalui CNN memungkinkan sistem memberikan umpan balik otomatis yang membantu pengguna dalam memperbaiki kesalahan sketsa mereka secara lebih cepat dan akurat.

Penelitian ini akan diuji coba secara langsung pada komunitas CokroSketsa Ponorogo sebagai subjek penelitian. Komunitas ini dipilih karena memiliki fokus utama pada pengembangan seni sketsa dan rutin melakukan aktivitas menggambar sketsa bersama. Dengan melibatkan komunitas ini, diharapkan pengujian aplikasi dapat memberikan gambaran nyata terkait efektivitas dan penerimaan teknologi dalam mendukung proses pembelajaran seni sketsa.

3.2 Perancangan Sistem

3.2.1 Flowchart Sistem



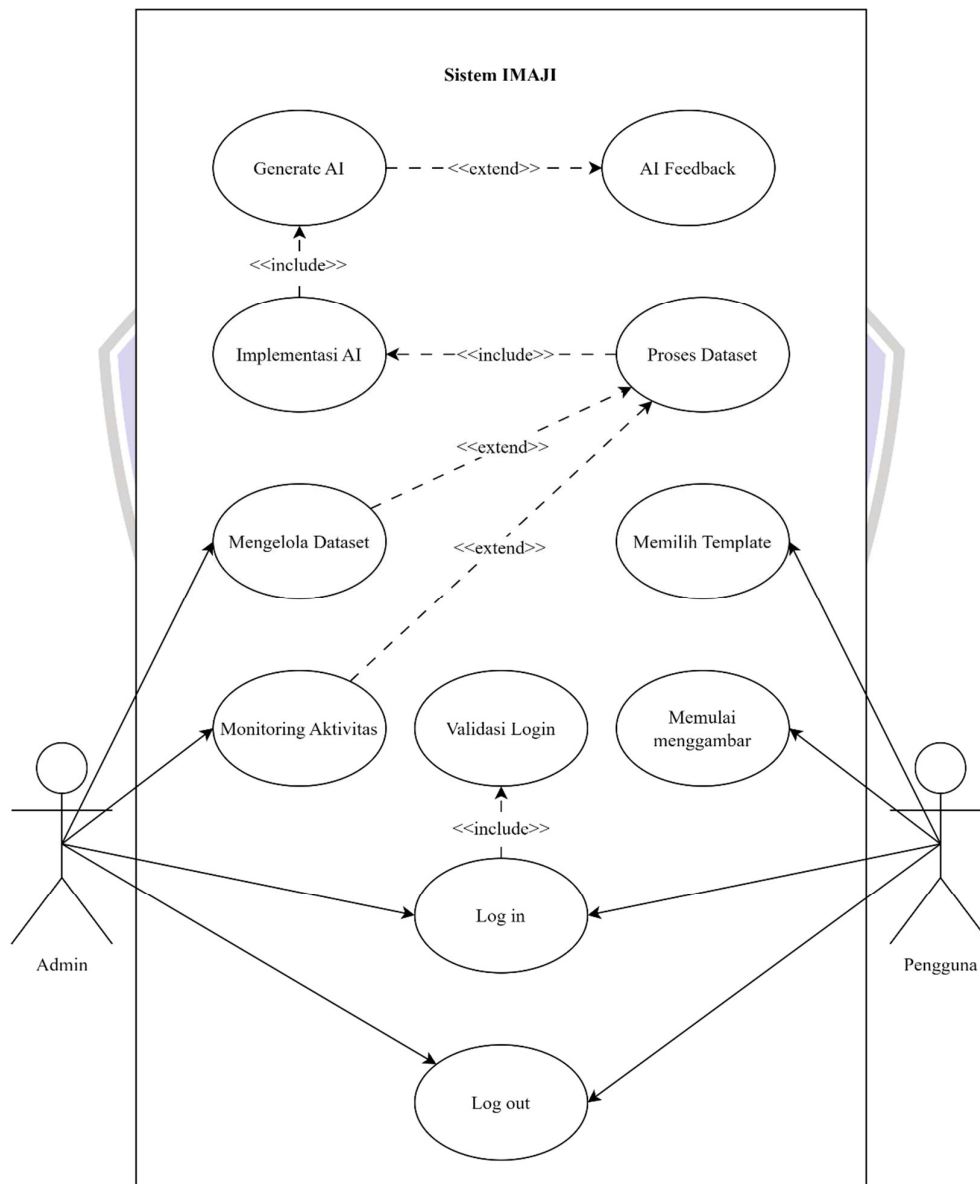
Gambar 3.2 Flowchart Sistem

Diagram alur sistem ini menggambarkan proses interaksi pengguna dengan aplikasi mulai dari awal hingga akhir. Pada Gambar 3.2 Proses dimulai dengan pengguna masuk ke aplikasi. Jika pengguna tidak memiliki akun, mereka diarahkan untuk mendaftar terlebih dahulu. Setelah berhasil masuk, pengguna diarahkan ke beranda untuk memilih *template* sketsa yang diinginkan.

Setelah memilih *template*, pengguna memulai proses menggambar dengan panduan dari aplikasi. Hasil gambar kemudian diunggah melalui fitur input hasil sketsa, di mana sistem akan menganalisis sketsa tersebut untuk menentukan tingkat kesesuaian dengan *template*. Jika hasil analisis menunjukkan ketidaksesuaian, sistem akan memberikan saran kepada pengguna untuk perbaikan.

Jika hasil sketsa sesuai dengan *template*, aplikasi akan menyimpan kinerja pengguna sebagai data yang dapat digunakan untuk evaluasi lebih lanjut. Proses ini berakhir dengan tahap selesai, yang menandakan bahwa semua langkah dalam aplikasi telah dilalui. Diagram ini mencerminkan alur kerja yang sistematis dan dirancang untuk mendukung pengguna dalam meningkatkan keterampilan menggambar mereka.

3.2.2 Diagram Use Case



Gambar 3.3 Use Case Diagram

Rancangan *use case* aplikasi ini melibatkan dua aktor utama, yaitu Pengguna dan *Admin*. Seperti yang terlihat pada Gambar 3.3 Pengguna menggunakan aplikasi untuk berbagai aktivitas seperti menggambar dengan panduan *template*, menerima umpan balik dari AI, dan mengunggah hasil gambar untuk dianalisis oleh sistem. Sedangkan, *Admin* memiliki peran untuk mengelola *template* gambar, pengguna, serta data terkait aplikasi.

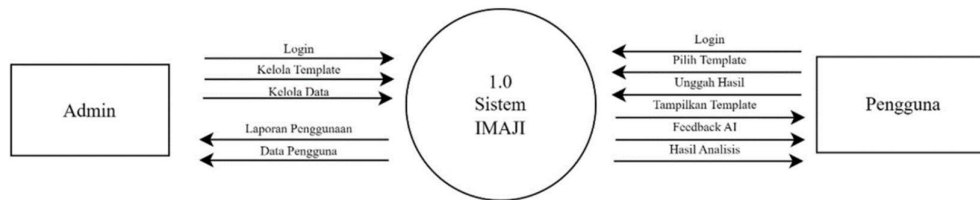
Proses dimulai dengan fitur *Login*, di mana baik Pengguna maupun *Admin* dapat masuk atau mendaftar ke dalam aplikasi menggunakan kredensial masing-masing. Setelah berhasil masuk, *User* dapat melanjutkan dengan memilih *template* yang akan digunakan untuk menggambar. Sistem kemudian memanfaatkan teknologi AR dalam fitur Proses Menggambar dengan AR untuk menampilkan *template* secara visual sebagai panduan menggambar.

Setelah proses menggambar selesai, *User* mengunggah hasil gambarnya melalui fitur Analisis Hasil Gambar, di mana sistem menggunakan algoritma CNN untuk membandingkan hasil gambar dengan *template* yang dipilih. Jika ditemukan kesalahan, sistem memberikan *Feedback* AI berupa saran perbaikan kepada pengguna.

Di sisi lain, *Admin* memiliki kemampuan untuk mengelola *template*, termasuk menambah, memperbarui, atau menghapus *template* gambar. Selain itu, *admin* juga dapat mengelola data pengguna, seperti melihat atau memperbarui informasi pengguna yang terdaftar. Rancangan ini memastikan aplikasi berjalan dengan baik untuk mendukung aktivitas *User* dan mempermudah pengelolaan oleh *Admin*.

3.2.3 Data Flow Diagram (DFD)

a. DFD Level 0 (ConTEXT Diagram)



Gambar 3.4 DFD Level 0

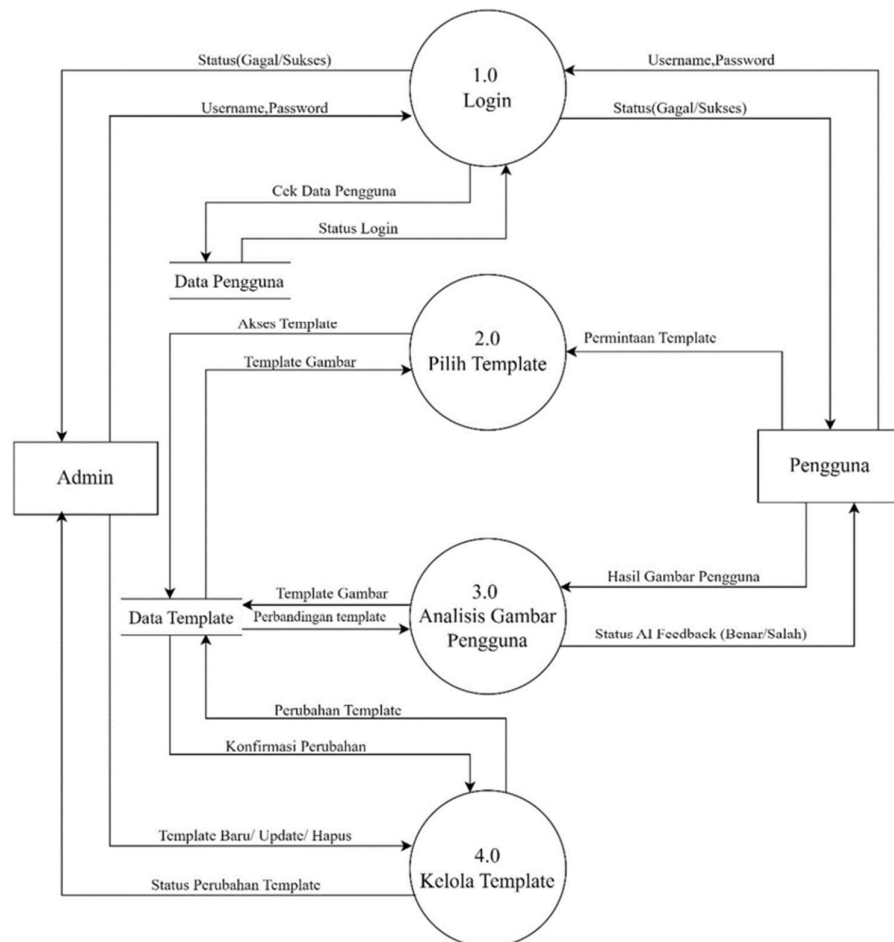
DFD Level 0 untuk Sistem utama menggambarkan hubungan antara entitas eksternal, proses utama, dan alur data. Entitas Eksternal yang terlibat terdiri dari *Pengguna* dan *Admin*, yang berinteraksi langsung dengan Sistem Imaji sebagai proses utama.

User berinteraksi dengan sistem melalui beberapa alur data, seperti *Login*, *Pilih Template*, dan *Unggah Hasil Gambar*. Sebagai respon, sistem mengirimkan data kembali ke *User*, seperti *Tampilan Template*, *Feedback AI*, dan *Hasil Analisis Gambar*. Hal ini mendukung pengalaman pengguna dalam menggambar dan menerima saran perbaikan secara otomatis.

Di sisi lain, *Admin* menggunakan sistem untuk *Login*, Mengelola Data Pengguna. Sebagai keluaran, sistem memberikan *Laporan Penggunaan* dan *Data Pengguna* kepada *admin* untuk membantu dalam pengelolaan aplikasi.

Dengan alur data ini, Sistem utama berfungsi sebagai pusat pengolahan data yang mendukung kebutuhan pengguna dan *admin* dalam satu kesatuan proses yang terintegrasi.

b. DFD level 1



Gambar 3.5 DFD level 1

DFD Level 1 merupakan dekomposisi dari proses utama yang ada pada DFD Level 0. Diagram ini menjelaskan secara lebih rinci alur data dan interaksi antara proses, entitas eksternal, dan penyimpanan data dalam sistem. Dijelaskan pada Gambar 3.5, sistem ini memiliki empat proses utama yang saling terhubung, yaitu proses login, pemilihan *template*, analisis gambar pengguna, dan pengelolaan *template*.

Berikut adalah penjelasan masing-masing proses pada DFD Level 1:

1. Proses *Login* (1.0)

Proses ini bertujuan untuk mengautentikasi pengguna dan *admin* yang ingin mengakses sistem. Pengguna atau *admin* memasukkan *Username* dan *password* ke dalam sistem. Sistem akan memvalidasi data yang diterima dengan mencocokkan

informasi yang tersimpan dalam database. Jika data valid, sistem memberikan status "Sukses" dan pengguna diarahkan ke halaman beranda atau fitur yang sesuai. Jika data tidak valid, sistem memberikan status "Gagal" dan meminta pengguna untuk mengulangi proses *login* atau mendaftar jika belum memiliki akun.

2. Proses Pilih *Template* (2.0)

Proses ini digunakan oleh pengguna untuk memilih *template* gambar yang ingin digunakan sebagai panduan menggambar. Sistem menampilkan daftar *template* yang tersedia berdasarkan data yang telah diunggah oleh *admin*. Setelah pengguna memilih *template*, data *template* tersebut dikirimkan ke pengguna untuk digunakan dalam fitur *Augmented Reality* (AR).

3. Proses Analisis Gambar Pengguna (3.0)

Pada proses ini, pengguna mengunggah hasil gambar yang telah dibuat berdasarkan *template* yang dipilih. Sistem akan menganalisis gambar tersebut menggunakan algoritma *Convolutional Neural Network* (CNN). Hasil analisis berupa status kecocokan gambar dengan *template* dan umpan balik berupa rekomendasi atau koreksi diberikan kepada pengguna. Proses ini bertujuan untuk meningkatkan kualitas gambar pengguna dengan memberikan evaluasi berbasis AI.

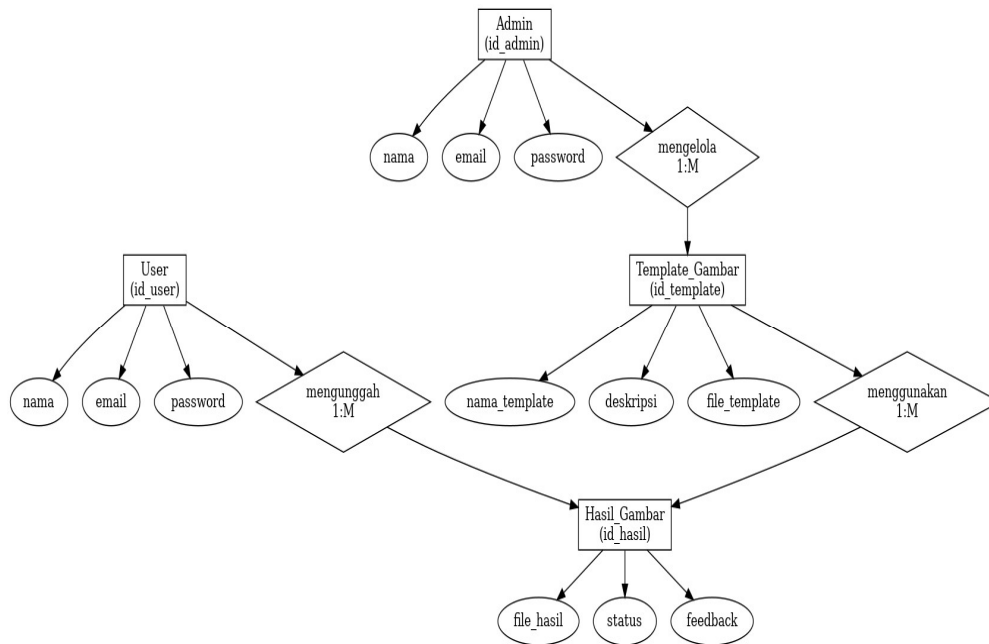
4. Proses Kelola *Template* (4.0)

Proses ini hanya dapat diakses oleh *admin*. Di sini, *admin* memiliki kontrol penuh untuk melakukan pengelolaan *template*, baik itu menambahkan *template* baru, memperbarui, maupun menghapus *template* yang sudah ada. Setiap perubahan yang dilakukan *admin* akan langsung berdampak pada data *template* yang tersedia bagi pengguna. Setelah perubahan dilakukan, sistem akan memberikan konfirmasi bahwa status perubahan berhasil. Proses ini memastikan bahwa hanya *template* yang relevan dan terbaru yang tersedia di sistem.

5. Alur Data

Setiap proses di DFD Level 1 terhubung melalui alur data yang mencakup informasi penting seperti kredensial pengguna, data *template*, hasil analisis, dan laporan sistem. Data yang mengalir dari pengguna dan *admin* ke sistem, maupun sebaliknya, direpresentasikan secara jelas untuk memastikan sistem berjalan sesuai dengan alur kerja yang telah dirancang.

3.2.4 Entity Relationship Diagram (ERD)



Gambar 3.6 Perancangan ERD

Di tahap ini, Pada gambar 3.6 dirancang struktur basis data untuk mendukung fungsionalitas aplikasi. Desain basis data diwujudkan dalam bentuk *Entity-Relationship Diagram* (ERD), yang menggambarkan entitas utama, atribut, serta relasi antar entitas.

Berikut adalah penjelasan dari masing-masing entitas yang terdapat pada ERD:

a. Entitas *Admin*

Entitas *Admin* merepresentasikan pihak pengelola sistem yang bertugas mengelola *template* gambar serta data pengguna dalam aplikasi.

1. Atribut:

- id_admin*: Identitas unik untuk setiap *admin*.
- nama*: Nama *admin*.
- email*: Alamat *email admin* untuk login.
- password*: Kata sandi untuk autentikasi.

2. Relasi:

- Admin* memiliki relasi 1:M dengan entitas *Template Gambar*, menunjukkan bahwa satu *admin* dapat mengelola banyak *template*.

b. Entitas *User*

Entitas *User* merepresentasikan pengguna aplikasi yang menggunakan *template* untuk menggambar, mengunggah hasil gambar, dan menerima umpan balik dari sistem.

1. Atribut:

- a. *id_user*: Identitas unik untuk setiap pengguna.
- b. *nama*: Nama pengguna.
- c. *email*: Alamat *email* pengguna untuk login.
- d. *password*: Kata sandi untuk autentikasi.

2. Relasi

- a. *User* memiliki relasi 1:M dengan entitas Hasil Gambar, yang menunjukkan bahwa satu *User* dapat menghasilkan dan mengunggah banyak hasil gambar.
- b. *User* juga memiliki relasi 1:M dengan entitas *Template* Gambar, yang menunjukkan bahwa satu *User* dapat menggunakan banyak *template* untuk menggambar.

b. Entitas *Template* Gambar

Entitas *Template* Gambar menyimpan informasi terkait *template* gambar yang dapat digunakan sebagai panduan oleh pengguna.

1. Atribut:

- a. *id_template*: Identitas unik untuk setiap *template*.
- b. *nama_template*: Nama *template* gambar.
- c. *deskripsi*: Deskripsi singkat mengenai *template*.
- d. *file_template*: *File template* berupa gambar atau sketsa.

2. Relasi:

- a. *Template* gambar memiliki relasi 1:M dengan entitas Hasil Gambar, menunjukkan bahwa satu *template* dapat digunakan untuk menghasilkan banyak hasil gambar.
- b. *Template* ini dikelola oleh *admin* melalui relasi 1:M dengan entitas *Admin*.

c. Entitas Hasil Gambar

Entitas Hasil Gambar menyimpan informasi terkait hasil gambar yang diunggah oleh *User*. Entitas ini juga mencatat status analisis dan umpan balik yang diberikan oleh sistem berbasis CNN.

1. Atribut:

- a. *id_hasil*: Identitas unik untuk setiap hasil gambar.
- b. *file_hasil*: *File* hasil gambar yang diunggah oleh *User*.
- c. *status*: Status analisis, seperti "Sesuai" atau "Tidak Sesuai".
- d. *Feedback*: Umpan balik yang diberikan oleh sistem untuk membantu perbaikan gambar.

2. Relasi:

- a. Hasil gambar memiliki relasi M:1 dengan entitas *Template* Gambar, menunjukkan bahwa satu hasil gambar dibuat berdasarkan satu *template* tertentu.
- b. Hasil gambar juga memiliki relasi M:1 dengan entitas *User*, menunjukkan bahwa satu hasil gambar diunggah oleh satu *User*.

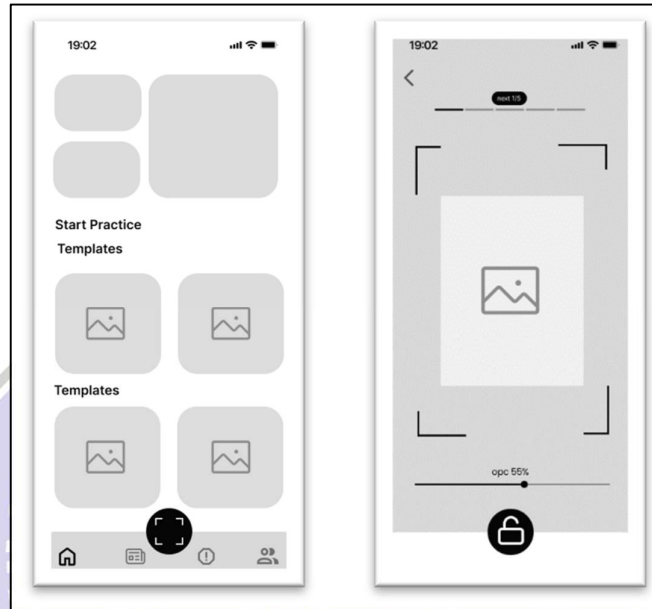
d. Relasi Antar Entitas

ERD ini menggambarkan hubungan logis antara entitas dalam sistem, dengan rincian sebagai berikut:

1. *Admin* Mengelola *Template* (1:M): Satu *admin* dapat mengelola banyak *template* gambar.
2. *User* Menggunakan *Template* (1:M): Satu *User* dapat memilih dan menggunakan banyak *template* untuk menggambar.
3. *Template* Digunakan untuk Hasil Gambar (1:M): Satu *template* dapat digunakan untuk menghasilkan banyak hasil gambar.
4. *User* Mengunggah Hasil Gambar (1:M): Satu *User* dapat mengunggah banyak hasil gambar ke sistem.

3.2.5 Perancangan Antarmuka

Perancangan antarmuka pada sistem ini dirancang untuk memberikan pengalaman pengguna yang nyaman dan terarah dalam menjalankan aktivitas menggambar dengan bantuan teknologi kecerdasan buatan (AI).



Gambar 3.7 Antarmuka Sistem

Interface utama terdiri dari beberapa tahap yang dirancang secara berurutan, seperti pada gambar 3.7, dimulai dari halaman *Home* yang menampilkan pilihan untuk memulai praktik dengan berbagai *template* gambar yang tersedia. Setelah pengguna memilih *template*, sistem menampilkan area praktik yang memandu pengguna dalam menggambar dengan memperlihatkan *template* secara transparan di tengah layar. Fitur indikator langkah-langkah di bagian atas membantu pengguna memantau tahap penyelesaian aktivitas yang sedang berlangsung.

Selama proses menggambar, sistem memberikan umpan balik langsung (*real-time Feedback*) berdasarkan hasil evaluasi kesesuaian gambar pengguna terhadap *template*. Setelah pengguna menekan tombol "Check Drawing", sistem akan menghitung nilai Mean Squared Error (MSE) antara gambar pengguna dan *template*. Berdasarkan nilai MSE tersebut, sistem menampilkan umpan balik berupa pesan teks otomatis seperti, "Hebat! Gambar Anda sangat mirip contoh." jika sangat

mirip, “Proporsi sudah baik, bisa sedikit dirapikan.” jika cukup mirip , dan “Perhatikan bentuk dan garis agar lebih sesuai contoh.” jika sama sekali tidak mirip.



Gambar 3. 8 Antarmuka Sistem (2)

Setelah proses menggambar selesai, sistem melakukan analisis kesesuaian gambar dengan membandingkannya dengan *template* yang dipilih, kemudian menampilkan persentase kemiripan dalam bentuk *similarity score* seperti pada gambar 3.8. Fitur ini memastikan pengguna dapat menilai kualitas hasil gambar mereka dengan lebih objektif.

Untuk mendukung kemudahan navigasi, desain *Interface* menggunakan ikon yang jelas seperti ikon kunci untuk mengunci gambar saat praktik berlangsung, serta ikon centang dan simpan yang berfungsi sebagai konfirmasi hasil akhir dan penyimpanan data. Perancangan ini mengutamakan kesederhanaan dan fungsionalitas, sehingga cocok digunakan dalam aplikasi edukasi berbasis latihan menggambar dengan bantuan AI.

3.2.6 Perancangan Database.

Database merupakan komponen penting dalam aplikasi Imaji yang digunakan untuk menyimpan data pengguna, *admin*, *template* gambar, serta hasil

gambar yang dihasilkan pengguna. Pada bagian ini, dijelaskan perancangan tabel-tabel yang digunakan, struktur kolomnya, serta relasi antar tabel.

a. Tabel *user*

Tabel *user* digunakan untuk menyimpan data pengguna aplikasi. Data ini mencakup informasi pribadi dan kredensial login. Struktur tabel adalah sebagai berikut:

Tabel 3.1 Struktur tabel *user*

Kolom	Tipe Data	Keterangan
<i>id_user</i>	INT	<i>Primary Key</i>
<i>nama</i>	<i>VARCHAR</i> (20)	
<i>email</i>	<i>VARCHAR</i> (50)	
<i>password</i>	<i>VARCHAR</i> (20)	

b. Tabel *admin*

Tabel *admin* digunakan untuk menyimpan data *admin* aplikasi. *Admin* memiliki akses untuk mengelola *template* gambar dan memantau aktivitas pengguna. Struktur tabel adalah sebagai berikut:

Tabel 3.2 Struktur tabel *admin*

Kolom	Tipe Data	Keterangan
<i>id_admin</i>	INT	<i>Primary Key</i>
<i>nama</i>	<i>VARCHAR</i> (20)	
<i>email</i>	<i>VARCHAR</i> (50)	
<i>password</i>	<i>VARCHAR</i> (20)	

c. Tabel *template_gambar*

Tabel *template_gambar* digunakan untuk menyimpan data *template* gambar yang tersedia di aplikasi. *Template* ini digunakan sebagai panduan oleh pengguna untuk menggambar. Struktur tabel adalah sebagai berikut:

Tabel 3.3 Struktur tabel *template_gambar*

Kolom	Tipe Data	Keterangan
<i>id_template</i>	INT	<i>Primary Key</i>

<i>nama_template</i>	<i>VARCHAR (20)</i>	
<i>deskripsi</i>	<i>TEXT</i>	
<i>file_template</i>	<i>VARCHAR (255)</i>	<i>Path/ File Template</i>

d. Tabel *hasil_gambar*

Tabel *hasil_gambar* digunakan untuk menyimpan data hasil gambar yang diunggah oleh pengguna setelah menggambar menggunakan panduan *template*. Tabel ini juga menyimpan status kesesuaian hasil dengan *template* serta *Feedback* dari sistem. Struktur tabel adalah sebagai berikut:

Tabel 3.4 Struktur tabel *hasil_gambar*

Kolom	Tipe Data	Keterangan
<i>id_hasil</i>	INT	<i>Primary Key</i>
<i>id_user</i>	INT	<i>Foreign Key(user)</i>
<i>id_template</i>	INT	<i>Foreign Key(template_gambar)</i>
<i>file_hasil</i>	<i>VARCHAR (255)</i>	<i>Path/ File Template</i>
<i>status</i>	<i>ENUM</i>	<i>“Correct”/ “Incorrect”</i>
<i>Feedback</i>	<i>TEXT</i>	

e. Relasi antar tabel

Relasi antar tabel dalam database ini dirancang sebagai berikut:

1. Relasi antara Tabel *user* dan *hasil_Gambar*

id_user pada tabel *hasil_gambar* menjadi *Foreign Key* yang mengacu pada *Primary Key* *id_user* di tabel *user*. Relasi ini memastikan setiap hasil gambar dikaitkan dengan pengguna yang mengunggahnya.

2. Relasi antara Tabel *template_gambar* dan *hasil_gambar*

id_template pada tabel *hasil_gambar* menjadi *Foreign Key* yang mengacu pada *Primary Key* *id_template* di tabel *template_gambar*. Relasi ini memastikan setiap hasil gambar dikaitkan dengan *template* yang digunakan. Relasi ini membantu dalam pengorganisasian data dan memudahkan pengambilan informasi secara terstruktur.

3.2.7 Pembagian Dataset

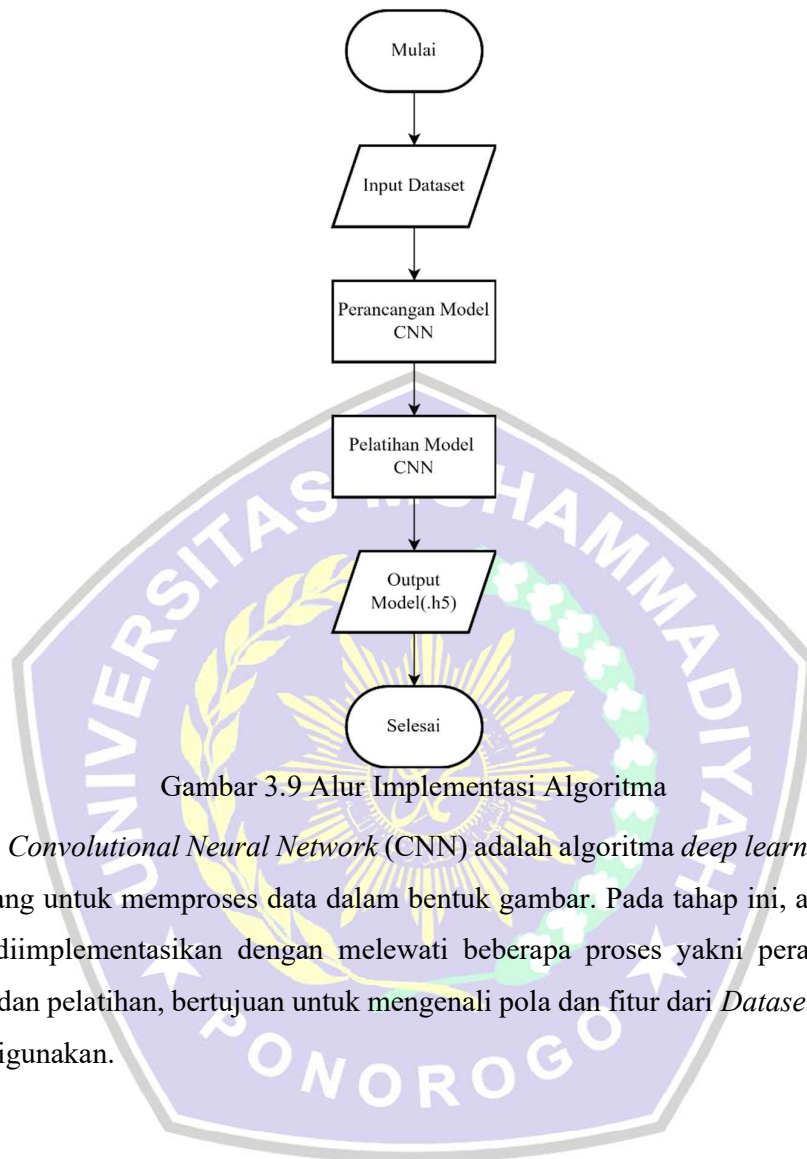
Proses pembagian data pada *Dataset* dilakukan setelah melalui tahap *preprocessing* gambar sketsa. Dalam penelitian ini, terdapat dua fase utama, yaitu fase *training* (pelatihan) dan fase *testing* (pengujian). Fase *training* menggunakan data latih (*train*) dan data validasi (*validation*). Data latih digunakan untuk melatih *model* CNN dalam mengenali pola gambar sketsa dari tiga kategori objek, yaitu kucing, ikan, dan anjing, yang masing-masing terdiri dari empat tahapan gambar (step1 hingga step4). Data validasi digunakan untuk mengevaluasi performa *model* selama proses pelatihan, guna memastikan *model* tidak mengalami *overfitting* dan mampu melakukan generalisasi dengan baik.

Sementara itu, data uji (test) digunakan dalam fase pengujian untuk menilai performa akhir *model* setelah proses pelatihan selesai. Pembagian *Dataset* dilakukan secara manual dengan rasio komposisi data sebesar 70% data latih, 15% data validasi, dan 15% data uji, dengan memperhatikan distribusi yang seimbang untuk setiap kelas dan setiap tahapan gambar sketsa. Berikut tabel Pembagian *Dataset* penelitian ini:

Tabel 3.5 Pembagian Dataset

Kelas	Jumlah Data	Data Latih	Data Validasi	Data Uji
Anjing_step1	800	500	150	150
Anjing_step2	800	500	150	150
Anjing_step3	800	500	150	150
Anjing_step4	800	500	150	150
Ikan_step1	800	500	150	150
Ikan_step2	800	500	150	150
Ikan_step3	800	500	150	150
Ikan_step4	800	500	150	150
Kucing_step1	800	500	150	150
Kucing_step2	800	500	150	150
Kucing_step3	800	500	150	150
Kucing_step4	800	500	150	150
Total	9600	6000	1800	1800

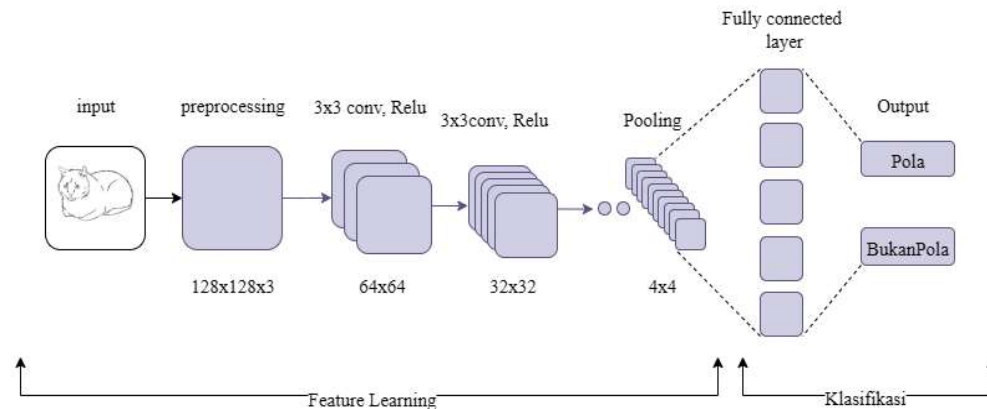
3.2.8 Implementasi Algoritma



Gambar 3.9 Alur Implementasi Algoritma

Convolutional Neural Network (CNN) adalah algoritma *deep learning* yang dirancang untuk memproses data dalam bentuk gambar. Pada tahap ini, algoritma CNN diimplementasikan dengan melewati beberapa proses yakni perancangan *model* dan pelatihan, bertujuan untuk mengenali pola dan fitur dari *Dataset* gambar yang digunakan.

a. Perancangan *Model* CNN



Gambar 3.10 Perancangan Arsitektur *Model* CNN *MobileNetV2*

Pada Gambar 3.10 menunjukkan rancangan arsitektur *model* yang digunakan dalam penelitian ini, yaitu *MobileNetV2*. *Model* ini dipilih karena memiliki keunggulan dalam hal kecepatan dan efisiensi, sehingga cocok untuk diterapkan pada perangkat mobile seperti smartphone Android.

MobileNetV2 merupakan arsitektur jaringan saraf konvolusional yang dikembangkan khusus untuk penggunaan pada perangkat dengan sumber daya terbatas. *Model* ini menggunakan struktur *depthwise separable convolution* dan *inverted residuals* yang membuatnya lebih ringan dibandingkan CNN konvensional, namun tetap mempertahankan performa yang kompetitif dalam pengenalan citra.

Pada tahap awal, citra sketsa yang telah diproses akan dimasukkan sebagai input ke dalam *model*. *MobileNetV2* akan melakukan ekstraksi fitur visual secara bertahap melalui blok-blok konvolusi yang saling terhubung. Setiap blok terdiri dari kombinasi operasi konvolusi, fungsi aktivasi ReLU, dan normalisasi batch untuk menjaga stabilitas pelatihan.

Citra akan diproses hingga menghasilkan vektor fitur berdimensi 1280, yang kemudian akan diteruskan ke tahap klasifikasi. Pada tahap klasifikasi, vektor fitur tersebut akan diproses oleh lapisan *fully connected* dengan fungsi aktivasi softmax untuk menghasilkan output klasifikasi multi-kelas. Dalam penelitian ini,

model dirancang untuk mengenali 12 kelas sketsa, yaitu gabungan dari 3 kategori objek (ikan, kucing, anjing) dan 4 tahapan menggambar.

b. Pelatihan *Model* CNN

Proses pelatihan *model MobileNetV2* dilakukan dengan pendekatan transfer learning, yaitu memanfaatkan bobot awal dari *model* yang telah dilatih pada *Dataset* besar (seperti *ImageNet*), kemudian disesuaikan (*fine-tuned*) dengan *Dataset* sketsa buatan sendiri.

Dataset pelatihan dibagi menjadi beberapa *batch* untuk efisiensi memori, dan pelatihan dilakukan selama beberapa *epoch*. Proses ini bertujuan untuk menyesuaikan *model* dengan karakteristik citra sketsa, sehingga dapat mengenali pola-pola garis dan bentuk yang khas dari gambar tangan.

Setelah pelatihan selesai, *model* diekspor ke dalam format .h5, lalu dikonversi ke format .tflite agar dapat diintegrasikan ke dalam aplikasi *Android*. Format ini memungkinkan *model* dijalankan secara efisien di perangkat *mobile* tanpa mengganggu performa sistem.

Dengan menggunakan *MobileNetV2*, sistem dapat melakukan proses klasifikasi gambar sketsa secara cepat dan akurat, sekaligus tetap ringan untuk dijalankan pada perangkat dengan spesifikasi menengah ke bawah.

3.2.9 Pengintegrasian *Model* CNN

Pengintegrasian *model* CNN dalam penelitian Imaji bertujuan untuk menghubungkan hasil pelatihan *model* dengan aplikasi yang digunakan oleh pengguna. *Model* CNN yang telah terlatih untuk mendeteksi dan mengenali pola sketsa akan diintegrasikan ke dalam aplikasi berbasis *Android* yang dikembangkan. Proses ini melibatkan penerapan *model* yang dilatih menggunakan *TensorFlow Lite*, yang memungkinkan aplikasi untuk menjalankan *model* CNN secara efisien di perangkat *mobile*. Dengan pengintegrasian ini, pengguna dapat mendapatkan umpan balik secara *real-time* saat membuat sketsa, dimana sistem akan mendeteksi kesalahan atau ketidaksesuaian dengan pola yang diinginkan dan memberikan koreksi atau saran untuk perbaikan.

3.2.10 Penerapan CNN untuk AI *Feedback*

Sistem AI *Feedback* dalam aplikasi IMAJI memanfaatkan *model* CNN *MobileNetV2* yang telah dikonversi ke dalam format *TensorFlow Lite* (.tflite) untuk memungkinkan inferensi langsung di perangkat *Android*. Pemilihan *MobileNetV2* didasarkan pada efisiensi komputasi dan ukuran *model* yang ringan, sehingga cocok untuk aplikasi *real-time* di perangkat *mobile* dengan keterbatasan sumber daya. *Model* ini dilatih untuk mengenali pola-pola visual pada sketsa berdasarkan kategori objek dan tahap gambar (*step*).

Secara teknis, ketika pengguna selesai menggambar, sistem melakukan akuisisi gambar melalui tangkapan layar (*screenshot*) dari hasil sketsa pengguna. Gambar tersebut kemudian diubah ukurannya menjadi 256x256 piksel dan dikonversi ke *ByteBuffer* dengan format RGB normalisasi $[0,1][0,1][0,1]$ untuk memenuhi spesifikasi input *model* *MobileNetV2*. Hasil inferensi berupa vektor probabilitas $y=[y_1, y_2, \dots, y_n]$ dengan n adalah jumlah kelas (misalnya 12 label dari 3 objek x 4 step). Label dengan nilai probabilitas maksimum $y'=\max(y_i)$ dianggap sebagai hasil prediksi *model*.

Namun, karena prediksi label saja belum cukup untuk menilai kualitas visual sketsa, sistem juga menghitung nilai *Mean Squared Error*(MSE) antara gambar pengguna (I) dan gambar *template* (T) dengan persamaan, menggunakan rumus 2.1 :

$$MSE = \frac{1}{n \cdot 3} \sum_{i=1}^n [(R_i - R'_i) + (G_i - G'_i) + (B_i - B'_i)^2] \div 225$$

Nilai MSE tersebut dikategorikan dalam tiga tingkatan evaluasi: sangat mirip ($MSE < 0.02$), cukup baik ($0.02 \leq MSE < 0.05$), dan perlu perbaikan ($MSE \geq 0.05$). Kategori ini digunakan sistem untuk menyusun umpan balik yang kontekstual. Dengan demikian, MSE berfungsi sebagai dasar logika pengambilan keputusan dalam sistem evaluasi otomatis.

Secara keseluruhan, integrasi CNN dalam sistem ini tidak hanya digunakan untuk klasifikasi, tetapi juga sebagai basis pengambilan keputusan dalam memberikan koreksi gambar berbasis AI. Analisis prediksi probabilistik dan

pengukuran jarak piksel (MSE) dikombinasikan untuk membentuk sistem *Feedback* yang adaptif, obyektif, dan efisien dalam membantu pengguna meningkatkan keterampilan menggambar secara bertahap.

3.3 Evaluasi Sistem

Evaluasi sistem dilakukan untuk menilai sejauh mana sistem yang dikembangkan, yakni aplikasi IMAJI, dapat mencapai tujuan fungsional dan performa yang telah ditetapkan. Evaluasi ini menjadi acuan untuk memastikan bahwa sistem mampu berjalan secara optimal, baik dari segi teknis maupun pengalaman pengguna akhir. Terdapat tiga aspek utama yang dievaluasi dalam penelitian ini, yaitu evaluasi *model* CNN, evaluasi sistem umpan balik berbasis AI, dan evaluasi aplikasi secara keseluruhan.

a. Evaluasi *model* CNN

Evaluasi terhadap *model Convolutional Neural Network* (CNN) bertujuan untuk mengukur performa *model* dalam mengklasifikasikan gambar sketsa berdasarkan objek dan tahapannya. *Model* yang digunakan adalah *MobileNetV2*, yang telah dilatih menggunakan *Dataset* sketsa ilustrasi. Evaluasi dilakukan menggunakan data uji (*test set*) yang belum pernah dilihat *model* sebelumnya. Metrik evaluasi yang digunakan mencakup:

1. Akurasi, Mengukur rasio prediksi benar terhadap seluruh data uji.
2. *Confusion Matrix*, Menunjukkan distribusi hasil prediksi *model* terhadap kelas-kelas yang tersedia.
3. *Classification Report*, Menyajikan nilai *precision*, *recall*, dan *F1-score* untuk setiap kelas, serta nilai rata-rata makro dan tertimbang.

Tujuan dari evaluasi ini adalah untuk menilai kemampuan generalisasi *model* serta mengidentifikasi potensi bias terhadap kelas tertentu.

b. Evaluasi AI *Feedback*

Evaluasi sistem umpan balik otomatis dilakukan untuk menilai efektivitas fungsi *showFeedbackBottomSheet()*, yang memberikan saran atau koreksi

berdasarkan nilai *Mean Squared Error*(MSE) antara gambar pengguna dan *template*. Nilai MSE digunakan sebagai indikator tingkat kemiripan visual.

Evaluasi dilakukan dengan menguji beberapa sketsa pengguna dan mencocokkannya dengan sketsa acuan (*template*). Rentang nilai MSE digunakan untuk menentukan kategori *Feedback* yang akan diberikan oleh sistem. Aspek yang dievaluasi meliputi:

1. Akurasi pemberian *Feedback* berdasarkan nilai MSE.
2. Kesesuaian pesan terhadap kualitas gambar.
3. Keterbacaan dan relevansi pesan umpan balik bagi pengguna.

Evaluasi ini bertujuan untuk memastikan bahwa sistem mampu memberikan koreksi yang tepat, mudah dipahami, dan mendukung proses belajar pengguna.

c. Evaluasi Aplikasi

Evaluasi aplikasi mencakup penilaian terhadap fungsi-fungsi utama yang terdapat dalam aplikasi IMAJI. Penilaian dilakukan melalui pengujian fitur-fitur sebagai berikut:

1. Fungsi klasifikasi gambar dan pemrosesan *model* AI.
2. Stabilitas dan kecepatan sistem, termasuk dalam kondisi normal maupun saat menggunakan fitur *Augmented Reality* (AR).
3. Responsivitas UI, terutama dalam menampilkan hasil klasifikasi dan *Feedback*.
4. Kesesuaian fitur dengan tujuan pembelajaran menggambar, seperti panduan overlay, mode AR/manual, dan penyimpanan hasil gambar.

3.4 Pengujian Sistem

Penelitian ini menggunakan metode *White Box Testing* untuk menguji fungsionalitas sistem berdasarkan analisis struktur internal, alur logika, dan jalur kode yang digunakan dalam pengembangan aplikasi IMAJI. Metode ini dipilih karena sistem memiliki komponen kompleks seperti pemrosesan *model* CNN, pengolahan gambar, serta interaksi logika antara modul *Augmented Reality* (AR), input pengguna, dan evaluasi berbasis AI *Feedback*.

Pengujian dilakukan pada fungsi-fungsi utama di dalam aplikasi, seperti pemanggilan *model TensorFlow Lite*, pengolahan gambar bitmap dari kamera, pemberian umpan balik otomatis, dan penyimpanan hasil latihan ke database Supabase. Setiap fungsi diuji dengan berbagai parameter, seperti resolusi gambar input, tingkat kontras sketsa, stabilitas pencahayaan sekitar, serta ketepatan deteksi pola berdasarkan langkah menggambar yang aktif.

Salah satu aspek penting dalam pengujian adalah mengukur akurasi sistem dalam mendeteksi dan memberikan umpan balik terhadap sketsa yang digambar pengguna. Akurasi sistem dapat dihitung dengan menggunakan rumus (2.2) berikut:

$$Akurasi\ Sistem = \frac{Hasil\ Benar}{Total\ Pengujian} \times 100\%$$

Dengan menggunakan rumus ini, kita dapat mengevaluasi sejauh mana sistem mampu memberikan umpan balik yang tepat dan akurat pada setiap skenario pengujian. Berikut adalah tabel skenario pengujian untuk fitur utama aplikasi pembelajaran sketsa :

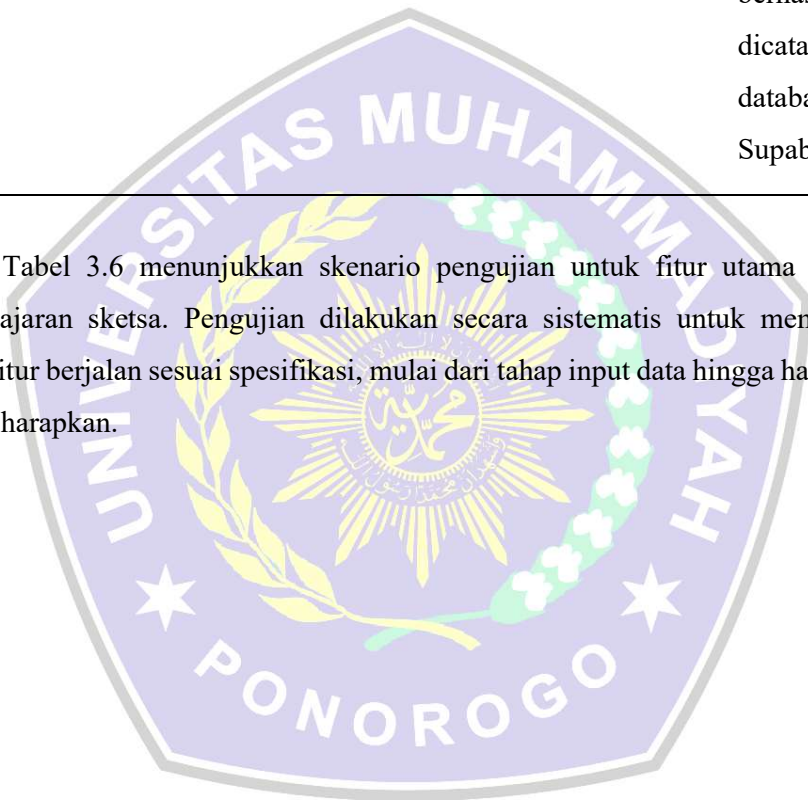
Tabel 3.6 Skenario Pengujian Sistem

No.	Fitur yang Diuji	Fungsi/Kode yang Diuji	Skenario Pengujian	Input	Hal yang Diharapkan
1	Panduan Gambar AR (Overlay)	<code>placeImageOnPlane()</code>	Menampilkan overlay AR jika anchor valid	Posisi anchor ARCore dari kamera	Gambar <i>template</i> muncul stabil di atas kertas dan tidak berpindah
2	Inferensi <i>Model CNN</i>	<code>runInference()</code>	Mengeksekusi <i>model</i> dan mengambil label serta confidence score	Gambar hasil sketsa (bitmap)	Sistem mengembalikan label prediksi dan nilai probabilitas secara benar

No.	Fitur yang Diuji	Fungsi/Kode yang Diuji	Skenario Pengujian	Input	Hal yang Diharapkan
3	Evaluasi Kemiripan (MSE)	<code>calculateMSE(img1, img2)</code>	Menghitung nilai kesalahan gambar	Dua bitmap (hasil gambar dan <i>template</i>)	Nilai MSE dikembalikan dengan akurasi tinggi (rentang 0–1)
4	Saran Koreksi Sketsa Otomatis	<code>showFeedbackBottomSheet()</code>	Memberikan saran berdasarkan nilai MSE	Nilai MSE dari hasil evaluasi	Pesan umpan balik ditampilkan sesuai rentang nilai MSE
5	Mode Manual Overlay	<code>switchToManualOverlay()</code>	Menampilkan overlay tanpa AR dan dapat dipindahkan secara manual	Overlay ditempatkan di layar	Gambar <i>template</i> muncul, dan bisa diatur posisi manual
6	Simpan dan Upload Hasil	<code>finishDrawing()</code> + upload API Retrofit	Menyimpan gambar hasil dan metadata ke Supabase	Bitmap gambar, <code>userId</code> , skor MSE	Data berhasil diunggah dan tersimpan di database serta storage

No.	Fitur yang Diuji	Fungsi/Kode yang Diuji	Skenario Pengujian	Input	Hal yang Diharapkan
7	Upload Template(A dmin)	uploadTemplateTo Supabase ()	Mengunggah h template baru untuk pengguna	File gambar template	Template baru tersimpan ke storage, dan metadata berhasil dicatat di database Supabase

Tabel 3.6 menunjukkan skenario pengujian untuk fitur utama aplikasi pembelajaran sketsa. Pengujian dilakukan secara sistematis untuk memastikan setiap fitur berjalan sesuai spesifikasi, mulai dari tahap input data hingga hasil akhir yang diharapkan.



BAB IV

ANALISA DATA DAN PEMBAHASAN

4.1 Analisis Data

Analisis data pada penelitian ini bertujuan untuk menjelaskan tahapan pengolahan dataset, proses pelatihan model, serta evaluasi hasil pengujian sistem IMAJI. Melalui analisis ini, penulis dapat menilai sejauh mana model Convolutional Neural Network (CNN) yang dikembangkan mampu mengenali pola sketsa sesuai dengan struktur dataset yang telah disusun

4.1.1 Pengolahan *Dataset*

Pengolahan *Dataset* merupakan tahap awal yang penting dalam proses pelatihan *model Convolutional Neural Network* (CNN) pada sistem IMAJI. *Dataset* pada penelitian ini dikembangkan secara mandiri, disesuaikan dengan kebutuhan aplikasi pembelajaran ilustrasi sketsa. Data terdiri dari gambar-gambar sketsa yang dibagi berdasarkan kategori objek serta urutan langkah menggambar, mulai dari bentuk dasar hingga hasil akhir yang lebih kompleks. Setiap gambar dikategorikan ke dalam empat langkah, yang merepresentasikan alur belajar pengguna dalam membentuk gambar secara bertahap.

Penulis menggunakan *Dataset* berdasarkan pembagian *Dataset* yang ada pada tabel 3.5, dengan struktur direktori yang sistematis, di mana setiap folder utama merepresentasikan satu jenis gambar (misalnya kucing, anjing dan ikan), setiap subfolder di dalamnya mewakili tahapan tertentu dalam proses menggambar, seperti *step1* hingga *step4*. Struktur ini memudahkan dalam mengelola dan mengelompokkan data berdasarkan progres menggambar. Berikut struktur direktori *Dataset* yang digunakan :

```
Dataset/  
├─ train/  
│   ├─ kucing_step1/  
│   │   .../  
│   ├─ kucing_step4/  
│   ├─ ikan_step1/  
│   │   .../  
│   ├─ anjing_step1/  
│   │   .../  
└─ .../  
└─ .../
```

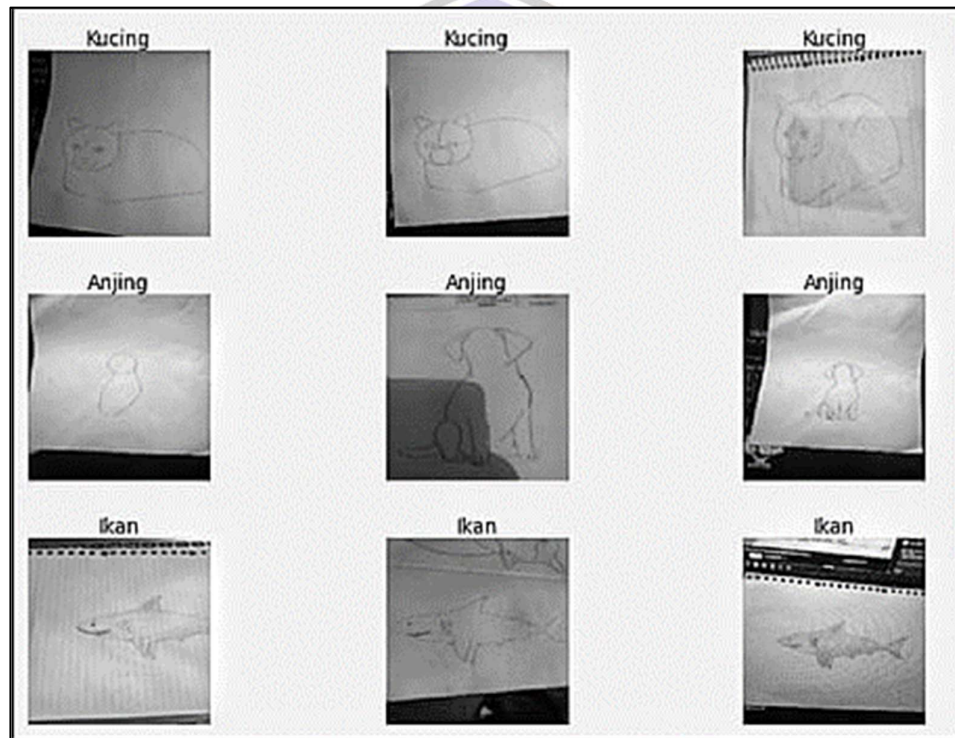
└─ val/...

└─ test/...

Dataset dibagi menjadi tiga bagian utama dengan komposisi sebagai berikut:

- Train*: 70%, Digunakan untuk melatih *model*.
- Validation*: 15%, Digunakan untuk memantau kinerja *model* selama pelatihan.
- Test*: 15%, Digunakan untuk menguji performa akhir *model*.

Untuk memberikan gambaran visual terhadap data yang digunakan dalam pelatihan *model* CNN, berikut ditampilkan sejumlah contoh gambar dari *Dataset* hasil pengumpulan dan klasifikasi berdasarkan kategori objek



Gambar 4.1 Contoh Gambar didalam *Dataset*

Gambar 4.1 menampilkan cuplikan data dari tiga kelas utama dalam *Dataset*, yaitu kucing, ikan, dan anjing. Setiap kelas ditampilkan sebanyak lima gambar yang diambil secara acak dari keseluruhan *Dataset*. Visualisasi ini menunjukkan variasi gaya sketsa dalam setiap kelas, baik dari segi bentuk, detail goresan, maupun sudut pandang gambar. Gambar-gambar ini dikumpulkan dan dikategorikan secara manual untuk memastikan kualitas data dan kesesuaian dengan tahapan proses menggambar.

Keberagaman pada setiap kelas ini penting untuk memastikan *model* CNN yang dilatih dapat mengenali pola dan karakteristik unik dari setiap objek, serta meningkatkan generalisasi *model* saat digunakan dalam aplikasi nyata.

4.1.2 Arsitektur *model* CNN

Model CNN yang digunakan dalam sistem IMAJI dibangun dengan memanfaatkan arsitektur *MobileNetV2*, yaitu *model pretrained* yang dirancang untuk efisiensi dan kinerja optimal pada perangkat mobile. *MobileNetV2* memiliki keunggulan dalam hal kecepatan inferensi dan ukuran *model* yang ringan, sehingga sangat cocok diimplementasikan dalam aplikasi *Android* seperti IMAJI.

Model ini dibangun menggunakan pustaka *TensorFlow* dan *Keras*, dengan pendekatan *transfer learning* dari *model MobileNetV2* yang telah dilatih sebelumnya pada *Dataset ImageNet*. Dengan metode ini, sebagian besar bobot *pretrained* digunakan ulang, dan hanya beberapa layer akhir yang disesuaikan dan dilatih ulang sesuai kebutuhan klasifikasi gambar sketsa dalam aplikasi.

Proses pembangunan arsitektur dilakukan dengan menambahkan beberapa *layer fully connected* setelah *base model MobileNetV2*. Penyesuaian dilakukan dengan menambahkan *layer GlobalAveragePooling*, *Dense*, *BatchNormalization*, serta *Dropout* untuk mengurangi risiko *overfitting*. Berikut potongan kode-nya:

```
base_model = MobileNetV2(weights='imagenet', include_top=False,
input_shape=(224, 224, 3))
base_model.trainable = True

for layer in base_model.layers[:-30]:
    layer.trainable = False

model = models.Sequential([
    base_model,
    layers.GlobalAveragePooling2D(),
    layer.Dense(256, activation='relu'),
    batchNormalization(),
    layers.Dropout(0.3),
    layers.Dense(128, activation='relu'),
    layers.Dropout(0.3),
```

```

layers.Dense(num_classes, activation='softmax')
])
model = Model(inputs=base_model.input, outputs=output)
model.summary()

```

Setelah arsitektur *model* berhasil dibangun, dilakukan pemeriksaan struktur *model* menggunakan fungsi *model.summary()*. Fungsi ini menampilkan detail dari setiap *layer*, bentuk output, serta jumlah parameter yang dapat dan tidak dapat dilatih. Ilustrasi arsitektur *model MobileNetV2* yang digunakan dapat dilihat pada Gambar 4.2 berikut,

Model: "sequential"		
Layer (type)	Output Shape	Param #
rescaling_1 (Rescaling)	(None, 224, 224, 3)	0
mobilenetv2_1.00_224 (Functional)	(None, 7, 7, 1280)	2257984
global_average_pooling2d_1 (GlobalAveragePooling2D)	(None, 1280)	0
dense_3 (Dense)	(None, 256)	327936
batch_normalization_1 (Batch Normalization)	(None, 256)	1024
dropout_2 (Dropout)	(None, 256)	0
dense_4 (Dense)	(None, 128)	32896
dropout_3 (Dropout)	(None, 128)	0
dense_5 (Dense)	(None, 12)	1548
=====		
Total params: 2621388 (10.00 MB)		
Trainable params: 1889292 (7.21 MB)		
Non-trainable params: 732096 (2.79 MB)		

Gambar 4.2 Arsitektur *Model CNN MobilNetV2*

Gambar 4.2 menunjukkan struktur layer dari *model CNN* yang digunakan. *Model* terdiri dari *layer rescaling*, *MobileNetV2* sebagai *backbone*, kemudian diikuti dengan *layer dense*, *batch normalization*, dan *dropout*. *Layer output*

memiliki 12 *neuron*, sesuai dengan jumlah total kelas pada *Dataset* yang digunakan. Terlihat bahwa *model* memiliki total 2.6 juta parameter, dengan sekitar 1.88 juta parameter dapat dilatih, yang menunjukkan kombinasi antara efisiensi dan fleksibilitas untuk fine-tuning.

4.1.3 Pelatihan *Model* CNN

Pelatihan *model* CNN dilakukan untuk menghasilkan *model* kecerdasan buatan yang mampu mengenali gambar sketsa sesuai kategori dan urutan langkah menggambar. *Model* ini dilatih menggunakan *Dataset* yang telah diproses dan dibagi ke dalam data latih, validasi, dan uji. Pelatihan dilakukan dengan memanfaatkan pustaka *TensorFlow* dan *Keras*, menggunakan arsitektur yang disesuaikan dengan jenis citra dan target klasifikasi.

a. Parameter Pelatihan

Model CNN dilatih dengan sejumlah parameter penting untuk memastikan proses konvergensi berjalan optimal. Pemilihan nilai parameter ini didasarkan pada praktik umum dalam pelatihan jaringan saraf konvolusional, serta melalui percobaan awal yang telah dilakukan.

Tabel 4. 1 Parameter Pelatihan

Parameter	Nilai
<i>Optimizer</i>	<i>Adam</i>
<i>Learning Rate</i>	0.001
<i>Loss Function</i>	<i>Categorical Crossentropy</i>
<i>Batch Size</i>	32
<i>Jumlah Epoch</i>	20
<i>Activation Function</i>	<i>ReLU (hidden), Softmax (output)</i>
<i>Dropout Rate</i>	0.3
<i>Input Image Size</i>	256 x 256
<i>Jumlah Kelas</i>	12 (3 objek × 4 langkah)

b. Visual hasil pelatihan

Selama pelatihan *model* berlangsung, tensorflow menghasilkan log pada setiap epoch yang mencatat nilai metrik *loss*, *accuracy*, *val_loss* dan *val_accuracy*.

```
Epoch 1/30
188/188 ————— 1385s 7s/step - accuracy: 0.4253 - loss: 2.0180 - val_accuracy: 0.4717 - val_loss: 1.7429
Epoch 2/30
188/188 ————— 35s 42ms/step - accuracy: 0.9022 - loss: 0.8792 - val_accuracy: 0.6600 - val_loss: 1.3851
Epoch 3/30
188/188 ————— 8s 41ms/step - accuracy: 0.9598 - loss: 0.7663 - val_accuracy: 0.8078 - val_loss: 1.0709
Epoch 4/30
...
Epoch 29/30
188/188 ————— 10s 48ms/step - accuracy: 0.9992 - loss: 0.5760 - val_accuracy: 0.9972 - val_loss: 0.5423
Epoch 30/30
188/188 ————— 8s 40ms/step - accuracy: 0.9996 - loss: 0.5733 - val_accuracy: 0.9972 - val_loss: 0.5460
```

Gambar 4.3 menunjukkan potongan *Log* progres pelatihan selama 30 *epoch*. Terlihat bahwa pada epoch pertama, akurasi *model* masih rendah (sekitar 42%) dengan nilai *val_accuracy* sebesar 47%. Namun, pada epoch ke-3, akurasi meningkat signifikan menjadi 95.9%, dan *val_accuracy* mencapai 80%. Pada akhir pelatihan (*epoch* 30), *model* berhasil mencapai akurasi *training* sebesar 99.96% dan akurasi validasi sebesar 99.72%, menunjukkan bahwa *model* belajar dengan sangat baik dan mampu menggeneralisasi data validasi.

The figure consists of two line graphs side-by-side, both sharing an x-axis representing 'Epoch' from 0 to 30.

The left graph is titled 'Training and Validation Accuracy'. The y-axis is labeled 'Accuracy' and ranges from 0.5 to 1.0. It shows two data series: 'Training Accuracy' (blue line with diamond markers) and 'Validation Accuracy' (orange line with square markers). Both accuracies start at approximately 0.45 at epoch 0 and rise sharply, reaching a plateau of 1.0 by epoch 10 and remaining stable thereafter.

Epoch	Training Accuracy	Validation Accuracy
0	0.45	0.45
1	0.62	0.66
2	0.92	0.81
3	0.96	0.89
4	0.98	0.95
5	0.99	0.98
10	1.00	1.00
15	1.00	1.00
20	1.00	1.00
25	1.00	1.00
30	1.00	1.00

The right graph is titled 'Training and Validation Loss'. The y-axis is labeled 'Loss' and ranges from 0.6 to 1.8. It shows two data series: 'Training Loss' (blue line with diamond markers) and 'Validation Loss' (orange line with square markers). Both losses start at approximately 1.75 at epoch 0 and decrease sharply, reaching a plateau of approximately 0.55 by epoch 10 and remaining stable thereafter.

Epoch	Training Loss	Validation Loss
0	1.75	1.75
1	1.52	1.38
2	0.84	1.08
3	0.76	0.88
4	0.71	0.76
5	0.68	0.68
10	0.63	0.58
15	0.61	0.55
20	0.60	0.54
25	0.59	0.54
30	0.58	0.53

61

Berdasarkan Gambar 4.4, grafik akurasi menunjukkan bahwa nilai akurasi *training* meningkat dengan sangat cepat sejak awal pelatihan, mencapai lebih dari 98% pada epoch ke-5 dan tetap stabil hingga akhir pelatihan. Akurasi validasi juga mengalami peningkatan tajam dan konsisten, mendekati nilai akurasi *training*, sehingga menandakan kemampuan generalisasi *model* yang sangat baik.

Pada grafik loss, baik *training* loss maupun *validation* loss mengalami penurunan yang signifikan seiring bertambahnya epoch. Titik minimum loss tercapai sekitar epoch ke-16, setelah itu nilai loss cenderung stabil dengan sedikit fluktuasi pada *validation* loss. Secara keseluruhan, tren loss tetap menurun dan tidak menunjukkan gejala *overfitting* yang berarti, karena gap antara *training* loss dan *validation* loss sangat kecil.

Dengan demikian, *model* CNN *MobileNetV2* yang digunakan telah berhasil melakukan pembelajaran secara efektif dan optimal, serta mampu melakukan generalisasi dengan baik pada data validasi.

4.1.4 Integrasi Model CNN

Setelah proses pelatihan selesai, *model* CNN yang telah dikembangkan dikonversi ke format *TensorFlow Lite* (.tflite) agar dapat dijalankan secara efisien pada perangkat Android. Proses konversi dilakukan menggunakan *TensorFlow Lite Converter*, kemudian *model* diintegrasikan ke dalam aplikasi IMAJI melalui *Android Studio*. Integrasi ini memungkinkan sistem untuk melakukan prediksi langsung dari input gambar pengguna secara real-time.

a. Konversi model

Model CNN awalnya disimpan dalam format .h5 setelah proses pelatihan menggunakan *TensorFlow* dan *Keras*. Untuk digunakan di Android, *model* ini perlu dikonversi ke format .tflite dengan menggunakan *TensorFlow Lite Converter*. Berikut adalah potongan kode untuk melakukan konversi *model*:

```
model.save("model_sketsa.h5")
converter = tf.lite.TFLiteConverter.from_keras_model(model)
tflite_model = converter.convert()
with open("model_imaji_new.tflite", "wb") as f:
    f.write(tflite_model)
```

```
print("Model berhasil dikonversi ke model_sketsa.tflite")
```

b. Penempatan *model*

Setelah dikonversi, file `model_cnn_imaji.tflite` ditempatkan ke dalam direktori proyek Android di:

```
app/src/main/assets/
```

Penempatan *model* pada folder *assets* memungkinkan *model* diakses sebagai sumber daya statis oleh aplikasi dan dimuat saat runtime melalui *interpreter TensorFlow Lite*.

c. Implementasi *TFLite* di Android

Integrasi *model* ke dalam aplikasi IMAJI dilakukan menggunakan *API TensorFlow Lite Interpreter* di dalam *Android Studio* (dalam kasus ini, menggunakan *Kotlin*). Proses pemanggilan *model* dilakukan di salah satu file yakni *CameraFragment*, yang bertanggung jawab mengambil gambar dari kamera dan mengirimkannya ke *model* untuk dilakukan inferensi.

```
val model = FileUtil.loadMappedFile(requireContext(),
    "model_imaji_new.tflite")
val options = Interpreter.Options()
// Preprocessing input bitmap -> ByteBuffer
// Menjalankan inferensi
tflite = Interpreter(model, options)
```

Model akan menerima gambar input yang telah diubah ukurannya (misalnya 256x256 piksel, *grayscale*/RGB sesuai *model*) dan mengembalikan output berupa skor probabilitas untuk masing-masing kelas gambar sketsa.

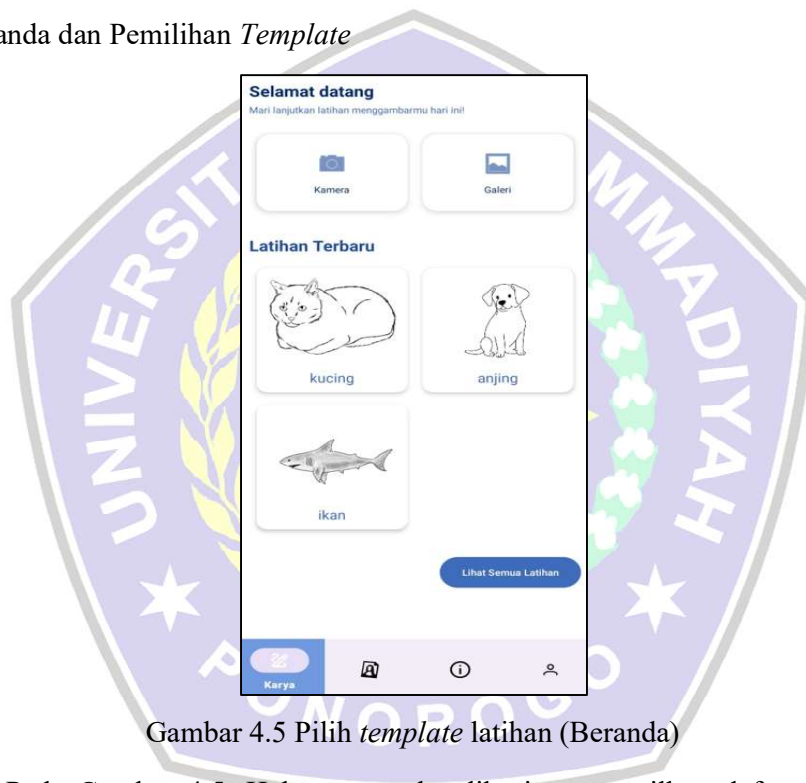
4.2 Pembahasan

Pembahasan pada penelitian ini difokuskan pada proses implementasi serta evaluasi kinerja sistem IMAJI yang telah dikembangkan. Bagian ini menjelaskan bagaimana integrasi antara model Convolutional Neural Network (CNN), teknologi Augmented Reality (AR), serta pengelolaan data dilakukan sehingga menghasilkan aplikasi pembelajaran ilustrasi sketsa yang interaktif dan adaptif terhadap kebutuhan pengguna.

4.2.1 Implementasi Sistem

Setelah *model* CNN berhasil diintegrasikan ke dalam aplikasi IMAJI, tahap implementasi mencakup penggabungan berbagai komponen utama, termasuk antarmuka pengguna (UI), teknologi *Augmented Reality* (AR), proses inferensi *model* AI, dan pengelolaan data menggunakan *Supabase PostgreSQL*. Untuk memudahkan pemahaman, berikut ini disajikan alur implementasi sistem berdasarkan tampilan-tampilan utama yang digunakan pengguna dalam aplikasi IMAJI.

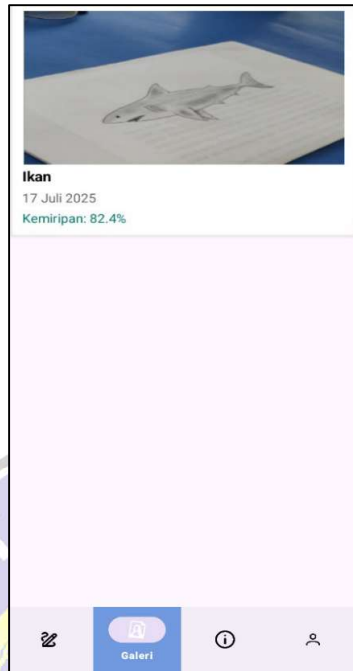
a. Beranda dan Pemilihan *Template*



Gambar 4.5 Pilih *template* latihan (Beranda)

Pada Gambar 4.5, Halaman awal aplikasi menampilkan daftar *template* yang dapat dipilih oleh pengguna untuk memulai sesi latihan menggambar. Pemilihan *template* akan menentukan gambar panduan yang ditampilkan pada fitur AR.

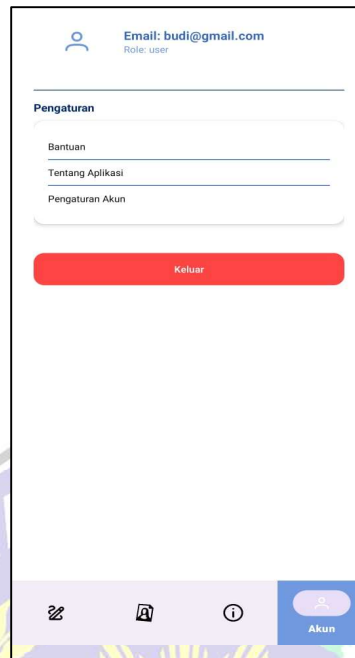
b. Galeri Hasil Latihan



Gambar 4.6 Galeri progres pengguna

Pada Gambar 4.6, Fitur ini menampilkan riwayat latihan pengguna, termasuk gambar yang telah dicoba dan skor yang diperoleh pada setiap langkah. Pengguna dapat meninjau kemajuan mereka dari waktu ke waktu.

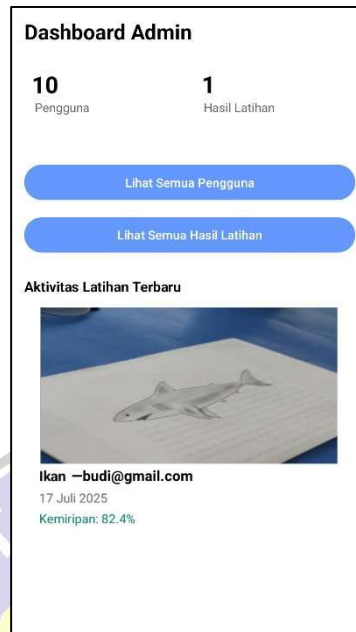
c. Profil dan Akun Pengguna



Gambar 4.7 Halaman Profil Pengguna

Gambar 4.7 menunjukkan aplikasi mendukung fitur autentikasi menggunakan *Supabase*, memungkinkan pengguna untuk login dan mengelola akun mereka. Halaman profil juga menampilkan informasi akun serta ringkasan skor dan jumlah latihan yang telah dilakukan.

d. Dashboard *Admin*



Gambar 4.8 Halaman Utama *Admin*

Untuk mendukung pengelolaan data latihan, *template*, dan pengguna, aplikasi menyediakan halaman khusus untuk peran *Admin*. Fitur ini hanya dapat diakses oleh akun yang memiliki hak akses khusus.

Dengan implementasi sistem yang menyeluruh ini, aplikasi IMAJI mampu memberikan pengalaman belajar menggambar berbasis teknologi AI dan AR secara real-time, efektif, dan menyenangkan.

4.2.2 Evaluasi Sistem

Evaluasi sistem dilakukan untuk menilai efektivitas dan kinerja IMAJI setelah proses implementasi. Penilaian mencakup performa *model* CNN, keakuratan umpan balik AI, serta kesesuaian antara fitur-fitur aplikasi dengan tujuan penelitian. Hasil evaluasi ini juga dibandingkan dengan beberapa penelitian terdahulu sebagai dasar validasi kualitas sistem yang dikembangkan.

a. Evaluasi *model*

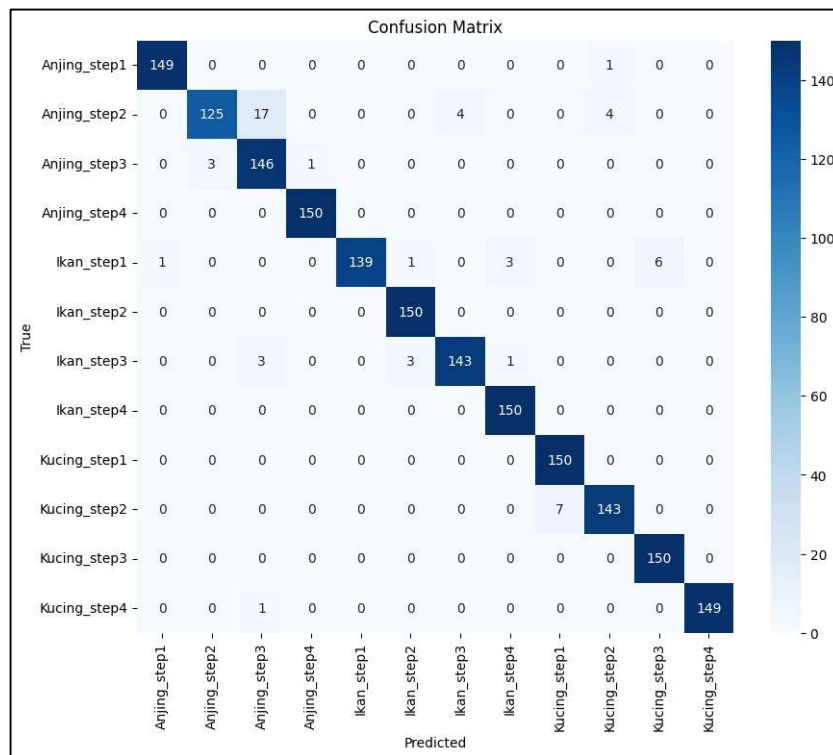
Evaluasi *model* dilakukan menggunakan data uji (*test set*) yang belum pernah dilihat oleh *model* selama proses pelatihan. Tujuannya adalah untuk menilai

seberapa baik *model MobileNetV2* mampu mengklasifikasikan gambar sketsa berdasarkan jenis objek (anjing, kucing, ikan) dan tahap menggambar (step 1–4). Penilaian dilakukan menggunakan beberapa metrik, yaitu akurasi, *confusion matrix*, dan *classification report*.

1. Akurasi Model:

Test Accuracy: 97%

Test Loss: 0.4847



Gambar 4.9 *Confusion Matrix CNN MobileNetV2*

Pada gambar 4.5, confusion matrix terlihat bahwa mayoritas prediksi berada pada diagonal utama, yang menunjukkan bahwa *model* mampu melakukan klasifikasi dengan akurasi tinggi. Beberapa kesalahan klasifikasi terjadi terutama pada kelas Anjing_step2 dan Ikan_step1, yang kemungkinan disebabkan oleh kemiripan visual antar tahap ketika sketsa mulai menyerupai bentuk akhir objek.

Classification Report:				
	precision	recall	f1-score	support
Anjing_step1	0.99	0.99	0.99	150
Anjing_step2	0.98	0.83	0.90	150
Anjing_step3	0.87	0.97	0.92	150
Anjing_step4	0.99	1.00	1.00	150
Ikan_step1	1.00	0.93	0.96	150
Ikan_step2	0.97	1.00	0.99	150
Ikan_step3	0.97	0.95	0.96	150
Ikan_step4	0.97	1.00	0.99	150
Kucing_step1	0.96	1.00	0.98	150
Kucing_step2	0.97	0.95	0.96	150
Kucing_step3	0.96	1.00	0.98	150
Kucing_step4	1.00	0.99	1.00	150
accuracy			0.97	1800
macro avg	0.97	0.97	0.97	1800
weighted avg	0.97	0.97	0.97	1800

Gambar 4.10 Hasil *Classification Report*

Berdasarkan hasil *classification report* pada gambar 4.6, *model MobileNetV2* menunjukkan performa yang sangat baik dengan rata-rata nilai *precision*, *recall*, dan *f1-score* sebesar 0.97, baik pada perhitungan macro average maupun weighted average. Hal ini menunjukkan bahwa *model* tidak hanya akurat secara keseluruhan, tetapi juga seimbang dalam menangani setiap kelas. Hampir semua kelas memiliki nilai *f1-score* di atas 0.95, yang menandakan bahwa *model* mampu melakukan klasifikasi secara konsisten dan stabil terhadap berbagai variasi gambar sketsa pada data uji. Tingginya nilai metrik ini menjadi indikator kuat bahwa *model* memiliki kemampuan generalisasi yang baik, serta minim bias terhadap kelas tertentu.

Dengan performa yang tinggi dan stabil, *MobileNetV2* terbukti efektif sebagai arsitektur utama untuk klasifikasi gambar sketsa dalam aplikasi IMAJI. Selain itu, *model* ini memiliki keunggulan dari segi efisiensi ukuran dan kecepatan prediksi, sehingga cocok untuk diterapkan pada perangkat Android dengan format *model TensorFlow Lite (.tflite)*.

b. Evaluasi AI *Feedback*

Fitur AI *Feedback* dalam aplikasi IMAJI dirancang untuk memberikan saran otomatis kepada pengguna setelah menyelesaikan proses menggambar. Evaluasi dilakukan berdasarkan nilai Mean Squared Error (MSE) yang dihitung

antara gambar pengguna dengan *template* referensi. Semakin kecil nilai MSE, semakin besar tingkat kemiripan gambar.

Umpan balik diberikan melalui fungsi *showFeedbackBottomSheet()*, yang membagi hasil evaluasi menjadi tiga kategori yakni, nilai mse ketika dibawah 0.02 akan Menampilkan *Feedback* “Hebat! Gambar Anda sangat mirip contoh.”, kemudian jika jika nilai mse dibawah 0.05 *Feedback* yang diberikan adalah “Proporsi sudah baik, bisa sedikit dirapikan.” dan kondisi terakhir ketika mse bernilai lebih dari sama dengan 0.05 maka *Feedback*-nya “Perhatikan bentuk dan garis agar lebih sesuai contoh.”.

Evaluasi ini menunjukkan bahwa sistem dapat memberikan saran koreksi yang relevan dan mudah dipahami, meskipun dalam bentuk teks. Berdasarkan hasil pengujian pada sejumlah data uji, sistem mampu membedakan tingkat kemiripan gambar dengan cukup akurat, sehingga *Feedback* yang diberikan sesuai dengan kualitas gambar pengguna. Walaupun belum menampilkan highlight visual pada kesalahan gambar, pendekatan berbasis teks ini cukup efektif dalam mendukung proses pembelajaran menggambar.

c. Evaluasi Aplikasi

Evaluasi aplikasi IMAJI dilakukan dari beberapa aspek, yaitu fungsionalitas, kecepatan respons, stabilitas sistem, dan kecocokan fitur terhadap tujuan pembelajaran sketsa. Berdasarkan hasil uji coba awal, aplikasi berhasil menjalankan seluruh proses inti dengan baik, mulai dari pengambilan gambar, pemrosesan melalui *model CNN MobileNetV2*, hingga penampilan hasil klasifikasi dan pemberian umpan balik secara otomatis.

Namun, dalam implementasi fitur *Augmented Reality* (AR), ditemukan kendala pada stabilitas posisi gambar *overlay*. Saat pengguna mulai menggambar, posisi *template* sering mengalami pergeseran yang disebabkan oleh tracking AR yang tidak konsisten, terutama ketika kamera bergerak, pencahayaan berubah, atau perangkat digunakan dalam posisi tidak stabil. Kemungkinan besar kendala ini diperparah oleh keterbatasan kompatibilitas perangkat, terutama pada smartphone dengan dukungan sensor atau spesifikasi AR yang rendah.

Hal ini terjadi karena sistem belum menerapkan penanda tetap (*anchor*) sebagai pengunci posisi objek dalam ruang AR, sehingga posisi *template* menjadi mudah bergeser. Sebagai solusi terhadap permasalahan ini, aplikasi menyediakan opsi mode manual (*non-AR*) sebagai alternatif. Dalam mode ini, *template* gambar ditampilkan secara statis di layar tanpa elemen AR, sehingga pengguna tetap dapat mengikuti panduan gambar dengan nyaman.

Meskipun mode manual tidak seakurat AR secara spasial, fitur ini terbukti efektif dalam menjaga kelangsungan penggunaan aplikasi dan tetap mendukung tujuan pembelajaran visual yang interaktif, terutama pada perangkat yang tidak mendukung AR secara optimal.

4.2.3 Pengujian Sistem

Pengujian dilakukan dengan pendekatan *white-box testing*, yaitu menelusuri dan menguji langsung struktur internal dari sistem aplikasi IMAJI, termasuk fungsi-fungsi kritis, percabangan, serta jalur eksekusi dalam kelas utama dalam aplikasi yakni *CameraFragment.kt*.

a. Panduan Gambar AR (*Overlay*) – *placeImageOnPlane()*

Fitur ini menampilkan gambar *template* di atas media gambar fisik dengan memanfaatkan teknologi Augmented Reality (AR). Fungsi *placeImageOnPlane()* digunakan untuk menempelkan gambar referensi secara presisi pada permukaan kertas yang telah diberi tanda silang di setiap sudutnya. Node gambar hanya ditampilkan jika tracking kamera aktif dan *anchor* valid, serta dikunci agar tidak berpindah setelah ditempatkan.

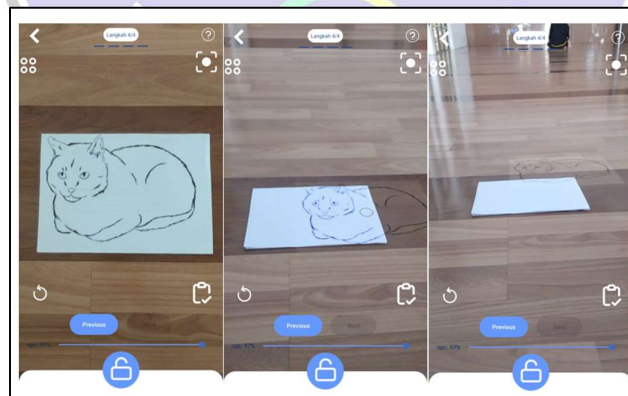
```
val anchor = hit.createAnchor()
val anchorNode = AnchorNode(anchor)
val node = TransformableNode(...).apply {
    this.renderable = renderable
    setParent(anchorNode)
}
```

Langkah pengujian dilakukan dengan menempatkan kertas pada bidang datar, kemudian menjalankan fitur AR dan mengarahkan kamera ke kertas dari sudut pandang yang bervariasi, yaitu 90°, 60°, 45°, dan 30°. Jarak antara kamera dan kertas diatur pada 30 cm, 45 cm, dan 70 cm. Tingkat pencahayaan diukur

menggunakan aplikasi lux meter dengan variasi 50 lux, 500 lux, dan 1000 lux. Selama pengujian, diamati apakah overlay tetap stabil atau mengalami pergeseran. Setiap kondisi pergeseran dicatat secara detail, termasuk besar pergeseran dan faktor penyebabnya, seperti perubahan sudut, jarak, maupun pencahayaan. Berikut tabel pengujiannya:

Tabel 4.2 Pengujian fitur Overlay

No	Sudut (°)	Jarak (cm)	Pencahayaan (lux)	Hasil	Keterangan
1	90°	30 cm	500	Stabil	Overlay presisi dan terkunci
2	60°	45 cm	1000	Stabil	Overlay presisi dan terkunci
3	60°	30 cm	100	Geser 5 cm ke kanan	Toleransi sudut mulai berkurang
4	45°	50 cm	500	Geser 13 cm ke kanan	Anchor kehilangan tracking
5	30°	70 cm	500	Geser total	Pencahayaan rendah, tracking gagal



Gambar 4.11 Pengujian AR-overlay

Berdasarkan hasil pengujian, juga dapat dilihat pada gambar 4.11, stabilitas overlay AR terbukti dipengaruhi oleh sudut pandang, jarak kamera, dan tingkat pencahayaan. Overlay tetap presisi dan terkunci pada sudut 90° dengan jarak 30 cm pada pencahayaan 500 lux, serta pada sudut 60° dengan jarak 45 cm pada pencahayaan 1000 lux. Namun, pada sudut 60° dengan jarak 30 cm dan pencahayaan 100 lux, overlay mulai bergeser sekitar 5 cm ke kanan, menandakan toleransi sudut berkurang pada cahaya rendah. Kondisi memburuk pada sudut 45° dengan jarak 50 cm dan pencahayaan 500 lux, di mana overlay bergeser 13 cm akibat anchor kehilangan tracking. Pada sudut 30° dengan jarak 70 cm dan pencahayaan hanya 500 lux, overlay gagal sepenuhnya karena jarak yang terlalu jauh.

b. Inferensi *Model CNN – runInference()*

Pengujian dilakukan untuk menilai kemampuan model CNN dalam mengenali gambar sketsa pengguna dengan tingkat kepercayaan (*confidence*) yang akurat. Proses inferensi dijalankan menggunakan fungsi *runInference()*, di mana gambar diubah menjadi format *ByteBuffer*, diproses oleh model CNN *MobileNetV2* berbasis *TensorFlow Lite*, lalu menghasilkan label prediksi beserta nilai *confidence*.

Berikut potongan kode pada sistem:

```
val inputBuffer =  
    convertBitmapToByteBuffer(inputBitmap,  
        inputSize)  
  
val result = Array(1) {  
    FloatArray(labelList.size) }  
tflite.run(inputBuffer, result)  
  
val maxIdx = result[0].indices.maxByOrNull {  
    result[0][it] } ?: -1  
val predictedLabel = if (maxIdx != -1)  
    labelList[maxIdx] else ""  
val confidence = result[0][maxIdx]
```

Fungsi ini digunakan untuk menjalankan proses inferensi *model CNN* berbasis *TensorFlow Lite* pada gambar sketsa yang telah dibuat oleh pengguna.

Gambar diubah menjadi format *ByteBuffer*, kemudian diproses oleh *model* untuk menghasilkan output berupa *array* skor prediksi (*logits*). Nilai tertinggi dalam array dipilih sebagai label prediksi dan disimpan dalam variabel *predictedLabel*, sedangkan *confidence score* disimpan di *confidence*.

Model menggunakan fungsi aktivasi *Softmax* untuk mengubah logit menjadi probabilitas, sebagaimana dirumuskan di rumus 2.2 dalam rumus:

$$\sigma(z_i) = \frac{e^{z_i}}{\sum_{j=1}^K e^{z_j}}$$

Contoh Penghitungan:

Diketahui,

$z = [1.2, 0.5, 2.1]$,

nilai konstanta $e = 2.718281828$,

$K = 3$

Penyelesaian:

Karena $K = 3$, maka,

$$\begin{aligned} \sum_{j=1}^K e^{z_j} &= e^{1.2} + e^{0.5} + e^{2.1} \\ e^{1.2} &= 3.3201 \\ e^{0.5} &= 1.6487 \\ e^{2.1} &= 8.1662 \end{aligned}$$

$$\sigma(z_i) = \frac{e^{z_i}}{\sum_{j=1}^K e^{z_j}}, \text{ Karena ada 3 kelas maka,}$$

1. Untuk kelas ke-1 (z_0):

$$\sigma(z_0) = \frac{e^{z_0}}{13.1349} = \frac{3.3201}{13.1349} = 0.2528$$

2. Untuk kelas ke-2 (z_1)

$$\sigma(z_0) = \frac{e^{z_1}}{13.1349} = \frac{1.6487}{13.1349} = 0.1256$$

3. Untuk kelas ke-3 (z_2)

$$\sigma(z_0) = \frac{e^{z_2}}{13.1349} = \frac{8.1662}{13.1349} = 0.6218$$

Tabel 4.3 Hasil contoh Prediksi 3 kelas

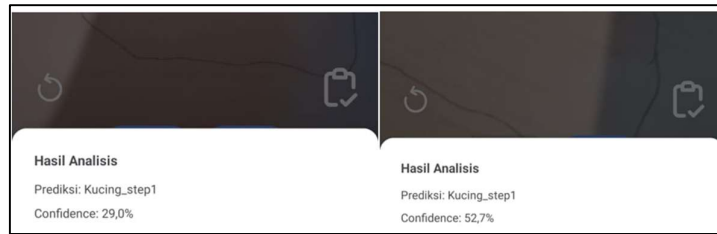
Kelas	Nilai Logit	e^z	Softmax (σ)	Probabilitas (%)
z_0	1.2	3.3201	0.2528	25.28%
z_1	0.5	1.6487	0.1256	12.56%
z_2	2.1	8.1662	0.6216	62.16%

Model memprediksi kelas ke-3 (z_2) sebagai hasil akhir dengan confidence score sebesar **62.16%**.

Pengujian inferensi model CNN dilakukan dengan menyiapkan tiga template sketsa dari dataset, kemudian masing-masing template digambar ulang menggunakan tinta hitam, tinta biru, dan pensil. Setiap hasil gambar difoto pada jarak kamera 30 cm dengan sudut 90° dan pencahayaan 500 lux, lalu diproses melalui fungsi *runInference()* untuk memperoleh nilai *confidence*. Pengujian dilanjutkan dengan kondisi pencahayaan berbeda, yaitu 200 lux dan 800 lux, guna membandingkan kinerja model pada lingkungan redup dan terang. Seluruh nilai confidence dan hasil prediksi dicatat untuk dianalisis.

Tabel 4.4 Hasil pengujian Inferensi

No	Warna Media Gambar	Pencahayaan (lux)	Confidence (%)	Hasil Prediksi	Keterangan
1	Hitam	500	52.7	Benar	Model mengenali dengan sangat baik
2	Hitam	200	29.0	Benar	Sedikit penurunan confidence pada cahaya redup
3	Biru	500	52.1	Salah	Model kurang optimal untuk tinta selain hitam
4	Biru	800	48.4	Salah	Warna di luar distribusi dataset memengaruhi prediksi
5	Pensil	500	46.9	Benar	Masih akurat untuk pensil dengan kontras tinggi



Gambar 4.12 Hasil Analisa gambar pengguna

Berdasarkan hasil pengujian, model CNN MobileNetV2 menunjukkan kinerja optimal dalam mengenali gambar tinta hitam dan pensil dengan kontras tinggi, dengan nilai confidence konsisten di atas 46% dan akurasi prediksi tetap benar meskipun pencahayaan turun hingga 200 lux (penurunan $\pm 7\%$). Namun, performa menurun signifikan pada media gambar dengan tinta berwarna seperti biru, di mana nilai confidence tetap di kisaran 48–52% tetapi prediksi menjadi salah. Hal ini terjadi karena dataset pelatihan hanya mencakup tinta hitam dan pensil, sehingga distribusi warna yang terbatas membatasi kemampuan generalisasi model terhadap variasi media gambar di luar data latih.

c. Evaluasi Kemiripan (MSE) – $calculateMSE(img1, img2)$

Pengujian fitur evaluasi kemiripan dilakukan untuk mengukur seberapa mirip hasil gambar pengguna dengan template menggunakan metode *Mean Squared Error* (MSE). MSE menghitung rata-rata selisih kuadrat antara nilai piksel gambar pengguna dan template. Semakin kecil nilai MSE, semakin tinggi tingkat kemiripan. Berikut potongan kode dari sistem:

```
val r1 = (pixels1[i] shr 16) and 0xFF
val r2 = (pixels2[i] shr 16) and 0xFF
...
sum += (r1 - r2)^2 + (g1 - g2)^2 + (b1 - b2)^2

return sum / (size * 3) / 255.0
```

Fungsi ini membandingkan dua gambar berdasarkan perbedaan nilai RGB dari masing-masing piksel. Total selisih kuadrat dihitung lalu dinormalisasi ke

rentang 0–1. Hasilnya dipakai untuk menilai kemiripan gambar sketsa dengan *template*. Perhitungan dilakukan dengan menggunakan rumus 2.1 berikut:

$$MSE = \frac{1}{n \cdot 3} \sum_{i=1}^n [(R_i - R'_i) + (G_i - G'_i) + (B_i - B'_i)^2] \div 225$$

Contoh Penghitungan:

Diberikan tiga piksel perbandingan:

Tabel 4.5 Contoh nilai piksel

Piksel	R	G	B	R'	G'	B'
P1	100	120	140	102	118	138
P2	200	180	160	198	179	161
P3	50	60	70	48	59	69

Penyelesaian:

$$MSE = \frac{1}{n \cdot 3} \sum_{i=1}^n [(R_i - R'_i) + (G_i - G'_i) + (B_i - B'_i)^2] \div 225$$

$$= (R_i - R'_i) + (G_i - G'_i) + (B_i - B'_i)^2$$

$$= (2^2 + 2^2 + 2^2) + (2^2 + 1^2 + 1^2) + (2^2 + 1^2 + 1^2)$$

$$= 4 + 4 + 4 + 1 + 1 + 4 + 1 + 1 = 24$$

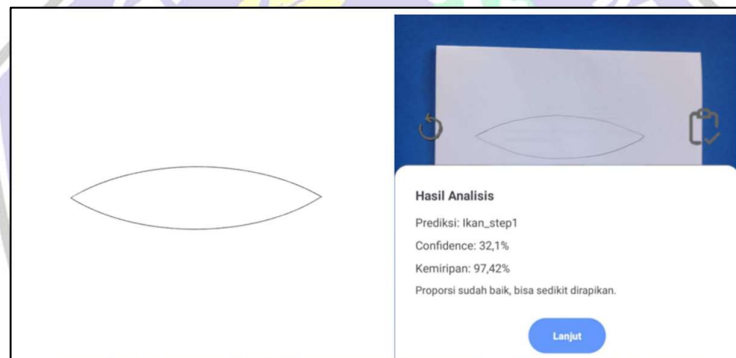
$$MSE = \frac{24}{3 \cdot 3 \cdot 225} = \frac{24}{2025} = 0.0119$$

Nilai mse = 0.0119, yang artinya gambar dari pengguna sangat mirip dengan panduan.

Pengujian fitur ini dilakukan dengan membandingkan hasil gambar pengguna dengan template acuan. Gambar dibuat menggunakan tinta hitam, biru, atau pensil, difoto pada jarak ± 30 cm, dan diuji pada pencahayaan 200, 500, dan 800 lux. Fungsi *calculateMSE*(img1, img2) menghitung selisih nilai RGB setiap piksel, di mana nilai MSE yang lebih kecil menunjukkan kemiripan lebih tinggi. Hasil MSE digunakan sebagai dasar pemberian umpan balik otomatis.

Tabel 4.6 Hasil pengujian MSE

No	Nilai MSE	Pencahayaannya (lux)	Hasil	Keterangan
1	0.0148	500	Sangat Mirip	Garis sesuai template
2	0.0234	200	Cukup Baik	Sedikit meleset pada detail
3	0.0456	800	Perlu Perbaikan	Beberapa proporsi meleset
4	0.0567	500	Tidak Mirip	Banyak garis tidak sesuai
5	0.0198	600	Sangat Mirip	Hampir identik dengan template



Gambar 4.13 Perbandingan Template dengan Hasil

Berdasarkan hasil pengujian MSE, terlihat bahwa tingkat pencahayaan dan ketelitian pengguna memengaruhi nilai kemiripan gambar secara signifikan. Pada pencahayaan optimal (500–600 lux), sistem mampu menghasilkan nilai MSE sangat rendah (≤ 0.02) yang menunjukkan kemiripan tinggi. Namun, pada pencahayaan redup (200 lux) atau terlalu terang (800 lux), nilai MSE cenderung meningkat sehingga kategori turun menjadi “cukup baik” atau “perlu perbaikan”. Selain itu, MSE hanya memberikan gambaran global perbedaan piksel tanpa mempertimbangkan lokasi kesalahan, sehingga meskipun nilai tinggi, tidak terlihat secara langsung bagian gambar yang keliru. Keterbatasan ini membuat penilaian kurang detail dalam mengarahkan pengguna untuk memperbaiki hasil gambar

d. Saran Koreksi Sketsa Otomatis – *showFeedbackBottomSheet()*

Fitur ini memberikan umpan balik otomatis kepada pengguna berdasarkan nilai Mean Squared Error (MSE) dan confidence dari model CNN MobileNetV2. Nilai MSE digunakan untuk mengukur tingkat kemiripan gambar dengan template, sedangkan confidence menunjukkan tingkat keyakinan CNN dalam mengenali gambar. Kombinasi kedua nilai tersebut dipetakan ke dalam tiga kategori:

1. "Hebat! Gambar Anda sangat mirip contoh." - sangat mirip,
2. "Proporsi sudah baik, bisa sedikit dirapikan." - cukup baik,
3. "Perhatikan bentuk dan garis agar lebih sesuai contoh." - perlu perbaikan.

Berikut implementasi kode pada sistem:

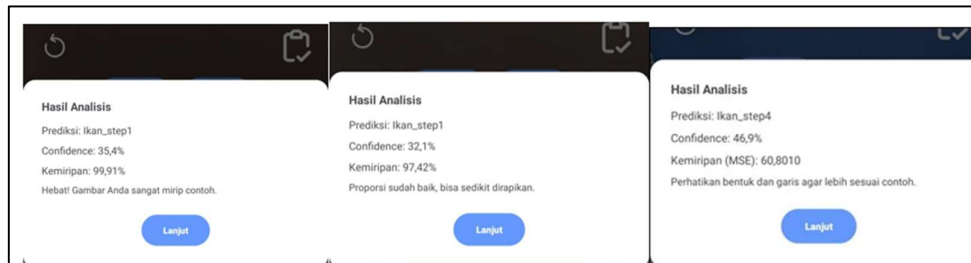
```
val Feedback = when {  
    mse < 0.02 -> "Hebat! Gambar Anda sangat mirip contoh."  
    mse < 0.05 -> "Proporsi sudah baik, bisa sedikit dirapikan."  
    else -> "Perhatikan bentuk dan garis agar lebih sesuai  
    contoh."  
}
```

Pengujian dilakukan dengan menyiapkan template gambar sebagai acuan, meminta pengguna menggambar ulang menggunakan tinta hitam, biru, atau pensil, lalu memotretnya dari jarak ± 30 cm pada variasi pencahayaan 200, 500, dan 800 lux. Sistem kemudian menghitung nilai MSE menggunakan *calculateMSE*(img1, img2) serta mengambil nilai *confidence* dari CNN untuk menentukan kategori umpan balik yang ditampilkan pada *Bottom Sheet*, yang selanjutnya diverifikasi kesesuaiannya dengan kondisi gambar sebenarnya.

Tabel 4.7 Hasil Pengujian AI Feedback

No	Nilai MSE	Confidence (%)	Kategori Feedback	Deskripsi Sistem
1	0.0148	35.4	Sangat Mirip	Garis sesuai template, hasil optimal.
2	0.0234	32.1	Cukup Baik	Detail sedikit meleset, namun proporsi utama benar.
3	0.0456	46.9	Perlu Perbaikan	Banyak proporsi dan garis yang tidak sesuai.

No	Nilai MSE	Confidence (%)	Kategori Feedback	Deskripsi Sistem
4	0.0198	82.1	Sangat Mirip	Hampir identik dengan template, perbedaan minimal.



Gambar 4.14 Hasil fitur AI feedback

Pengujian pada fitur Saran Koreksi Sketsa Otomatis menunjukkan bahwa kombinasi nilai MSE dan confidence model CNN dapat menghasilkan kategori umpan balik yang sesuai, yaitu *sangat mirip*, *cukup baik*, dan *perlu perbaikan*. Hasil menunjukkan bahwa nilai MSE yang rendah cenderung diikuti confidence yang tinggi, menghasilkan prediksi *sangat mirip*, sedangkan nilai MSE yang tinggi dengan confidence moderat atau rendah menghasilkan kategori *perlu perbaikan*. Meskipun demikian, akurasi penentuan kategori masih bergantung pada kualitas pencahayaan dan posisi kamera saat pengambilan gambar.

e. *Mode Manual - switchToManualOverlay()*

Fitur mode manual pada aplikasi IMAJI dirancang untuk memberikan fleksibilitas kepada pengguna dalam mengatur posisi, ukuran, dan rotasi gambar panduan tanpa bantuan teknologi AR. Pengujian dilakukan untuk memastikan bahwa setiap interaksi yang diberikan pengguna dapat direspons dengan baik oleh sistem, sehingga proses pembelajaran tetap berjalan meskipun AR tidak digunakan. Berikut potongan kode yang mengimplementasikan fungsi *switchToManualOverlay()* dalam sistem:

```
private fun switchToManualOverlay() {
    binding.transformOverlay.visibility = View.VISIBLE
    binding.transformOverlay.setLocked(false)

    val imageUrl = getImageUrl(templateName, currentStep)
```

```

        Glide.with(requireContext())
            .asBitmap()
            .load(imageUrl)
            .into(binding.transformOverlay)

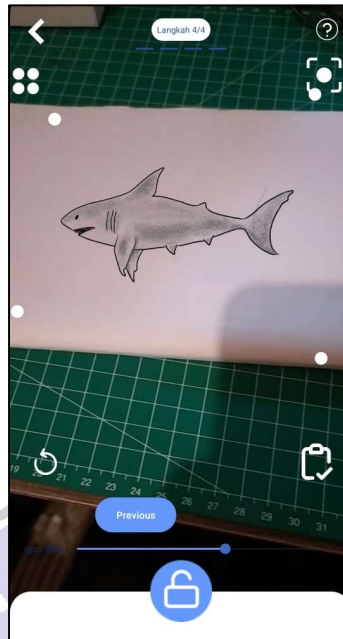
        binding.btnSwitchToManual.visibility = View.GONE
    }

```

Proses pengujian dilakukan dengan mengaktifkan mode manual melalui tombol navigasi, kemudian mencoba beberapa aksi seperti menggeser gambar panduan ke arah horizontal dan vertikal, memperbesar dan memperkecil (zoom in/zoom out), serta melakukan rotasi. Setiap aksi diamati untuk menilai sejauh mana sistem merespons input pengguna. Hasil pengujian interaksi pengguna dengan sistem ditunjukkan pada Tabel 4.8 berikut:

Tabel 4. 8 Hasil Pengujian Mode Manual

No	Aksi Pengguna	Respons Sistem	Keterangan
1	Geser gambar ke kiri/kanan	Berhasil	Gerakan responsif dan akurat
2	Geser gambar ke atas/bawah	Berhasil	Tidak ada delay pergerakan
3	Zoom in/zoom out	Berhasil	Skala proporsional
4	Rotasi gambar	Berhasil	Rotasi lancar tanpa distorsi
5	Navigasi ke step berikutnya	Gagal	Fitur hanya menampilkan 1 step



Gambar 4.15 Implementasi Mode Manual

Dari hasil pengujian tersebut, dapat disimpulkan bahwa fitur mode manual berfungsi dengan baik dalam penyesuaian posisi, ukuran, dan rotasi gambar panduan. Seluruh aksi utama berjalan lancar dan responsif. Namun, masih terdapat keterbatasan yaitu fitur belum mendukung navigasi antar langkah (*step*), sehingga pengguna hanya dapat menampilkan satu gambar panduan dalam mode ini.

f. Simpan dan *Upload Hasil* – *finishDrawing()* dan *Retrofit API*

Fitur ini berfungsi untuk mengunggah hasil gambar pengguna ke server Supabase melalui API *Retrofit*. Proses ini memastikan setiap hasil karya tersimpan secara terpusat di server, sehingga dapat diakses kembali untuk evaluasi maupun dokumentasi. Integrasi dengan Supabase juga memudahkan pengelolaan dan pencatatan data hasil gambar secara terstruktur. Berikut implementasi kode pada sistem:

```

val imageUrl = uploadImageToSupabase(bitmap, userId)
...
val result = LatihanResult(...)
private fun finishDrawing(originalBitmap: Bitmap, mse:
Double) ...

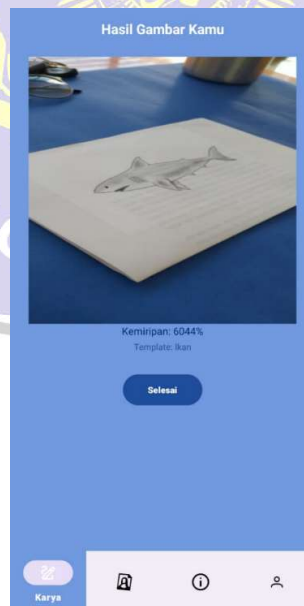
```

```
val response =
RetrofitClient.api.saveLatihanResult(listOf(result))
```

Pengujian dilakukan dengan menyelesaikan gambar di aplikasi, lalu menekan tombol "Simpan & Upload". Sistem diuji pada berbagai kondisi jaringan, seperti koneksi Wi-Fi stabil dan koneksi seluler yang tidak stabil. Setelah proses selesai, file di Supabase diperiksa untuk memastikan unggahan berhasil. Hasil pengujian menunjukkan bahwa pada koneksi stabil proses berjalan lancar, sedangkan pada koneksi tidak stabil, unggahan dapat tertunda atau gagal sehingga memerlukan mekanisme retry.

Tabel 4. 9 Hasil Pengujian fitur *Upload* Hasil

No	Kondisi Jaringan	Status Upload	Keterangan
1	Wi-Fi stabil (50 Mbps)	Berhasil	Proses cepat dan tanpa error
2	Seluler stabil (10 Mbps)	Berhasil	Kecepatan upload normal
3	Seluler tidak stabil	Gagal	Upload gagal, perlu retry
4	Tanpa koneksi internet	Gagal	gagal upload



Gambar 4.16 Hasil gambar dan upload gambar pengguna

Fitur simpan dan upload bekerja dengan baik pada jaringan stabil, namun pada jaringan tidak stabil atau tanpa koneksi internet proses upload gagal meskipun penyimpanan lokal tetap berhasil. Diperlukan fitur retry otomatis agar unggahan dapat dilanjutkan saat koneksi membaik.

g. *Upload Template - uploadTemplateToSupabase()*

Fitur ini memungkinkan admin menambahkan template gambar baru ke server Supabase tanpa perlu memperbarui aplikasi. Dengan begitu, konten latihan dapat diperbarui secara fleksibel dan langsung tersedia bagi semua pengguna. Proses ini dilakukan melalui fungsi `uploadTemplateToSupabase()`, yang mengunggah file template ke bucket penyimpanan Supabase dengan struktur direktori yang telah ditentukan. Berikut kode implementasi pada sistem:

```
val imageUrl = uploadTemplateToSupabase(bitmap, templateName)
...
val template = TemplateUploadRequest(
    name = templateName,
    category = category,
    imageUrl = imageUrl,
    steps = steps
)
val response = RetrofitClient.api.uploadTemplate(template)
```

Pengujian dilakukan dengan memilih file template dari perangkat admin, menjalankan fungsi unggah, lalu memverifikasi bahwa file berhasil tersimpan di Supabase pada path yang benar. Setelah itu, aplikasi pengguna diperiksa untuk memastikan template baru muncul di daftar latihan. Hasil pengujian menunjukkan bahwa proses berjalan lancar pada koneksi internet stabil, sementara pada koneksi lambat proses unggah memerlukan waktu lebih lama namun tetap berhasil.

Tabel 4. 10 Hasil Pengujian *Upload Template*

No	Jenis File	Ukuran File	Status Upload	Keterangan
1	PNG	120 KB	Berhasil	Template langsung muncul di aplikasi
2	JPG	500 KB	Berhasil	Tidak ada penurunan kualitas visual
3	PNG	2 MB	Berhasil	Waktu upload lebih lama ±5 detik

No	Jenis File	Ukuran File	Status Upload	Keterangan
4	PNG	5 MB	Gagal	Melebihi batas ukuran file Supabase

Gambar 4.17 Upload *template* baru (*Admin*)

Fitur ini berjalan dengan baik untuk file berukuran kecil hingga menengah. Namun, ukuran file yang terlalu besar (lebih dari batas Supabase) menyebabkan kegagalan upload. Optimalisasi kompresi file sebelum upload diperlukan untuk menghindari masalah ini.

h. Hasil Pengujian fitur

Pada Tabel 4.11, dijabarkan Hasil Uji berdasarkan skenario pengujian fungsionalitas sistem IMAJI yang sebelumnya telah dilakukan.

Tabel 4.11 Hasil pengujian Sistem

No.	Fitur yang Diuji	Jumlah	Jumlah	Jumlah	Keterangan
		Uji Coba	Berhasil	Gagal	
1	Panduan Gambar AR (<i>Overlay</i>)	5	2	3	<i>overlay</i> tampil stabil pada bidang kertas
2	Inferensi Model CNN	5	3	2	2 gambar tidak dikenali dengan benar oleh <i>model</i>

3	Evaluasi Kemiripan (MSE)	5	5	0	Nilai MSE berhasil dihitung dan akurat pada setiap pasangan gambar
4	Saran Koreksi Sketsa Otomatis	4	4	0	<i>Feedback</i> muncul sesuai rentang nilai MSE
5	Mode Manual Overlay	5	4	1	<i>Template</i> muncul dan dapat diatur posisi secara manual
6	Simpan dan Upload Hasil	4	2	2	Data hasil gambar berhasil diunggah ke <i>Supabase</i> (<i>storage</i> dan <i>database</i>)
7	Upload Template (Admin)	4	3	1	<i>Admin</i> berhasil menambahkan <i>template</i> baru
Total		32	23	9	

Penghitungan akurasi sistem:

$$Akurasi Sistem = \frac{23}{32} \times 100\% = 71.9\%$$

Berdasarkan pengujian terhadap 32 skenario uji coba, sebanyak 23 skenario berhasil dan 9 mengalami kegagalan atau ketidaksesuaian output. Fitur yang paling memerlukan perbaikan adalah Panduan Gambar AR (Overlay) yang sensitif terhadap sudut pandang dan pencahayaan, serta Saran Koreksi Sketsa Otomatis yang kadang memberikan kategori tidak sesuai dengan nilai MSE yang dihitung. Dengan akurasi keseluruhan 71.9%, sistem IMAJI dinilai cukup efektif, meskipun masih ada ruang perbaikan pada stabilitas AR dan konsistensi evaluasi otomatis.

BAB V

PENUTUP

5.1 Kesimpulan

Berdasarkan hasil penelitian dan implementasi sistem yang telah dilakukan, maka dapat disimpulkan sebagai berikut:

1. Selama proses menggambar, sistem membantu pengguna dengan memberikan umpan balik berbasis *Artificial Intelligence* (AI). Umpan balik yang diberikan terbatas pada tiga kategori, yaitu “sangat mirip”, “cukup baik” dan “perlu perbaikan”. Ketiga kategori tersebut ditentukan berdasarkan nilai *confidence* dari model CNN *MobileNetV2* dan *Mean Squared Error* (MSE) sebagai indikator tingkat kemiripan terhadap gambar template.
2. Integrasi *Augmented Reality* (AR) dan algoritma *Convolutional Neural Network* (CNN) berhasil diterapkan melalui fitur *overlay* gambar di atas media gambar fisik. Fitur ini memberikan panduan visual yang presisi dan membantu pengguna menggambar sesuai pola sketsa. Meski stabilitas *overlay* masih terpengaruh oleh sudut pandang dan pencahayaan, sistem menunjukkan akurasi keseluruhan sebesar 71.9%, yang mengindikasikan efektivitas sistem dalam mendukung proses pembelajaran sketsa.

5.2 Saran

Berdasarkan hasil penelitian dan pengalaman selama pengembangan sistem, beberapa saran yang dapat diajukan untuk pengembangan lebih lanjut adalah sebagai berikut:

1. Perlu dilakukan peningkatan stabilitas pada fitur *Augmented Reality*, terutama ketika lingkungan, sudut pandang, atau cahaya berubah. Metode pelacakan berbasis marker atau teknologi *pose estimation* yang lebih andal adalah solusi.
2. Demi mengantisipasi fitur *Overlay* AR yang rentan bergeser saat pencahayaan sekitar tidak mencukupi atau berubah-ubah, peningkatan lebih lanjut diperlukan. Pustaka *auto brightness compensation* dan validasi kondisi kamera sangat disarankan untuk menjadi solusi.

3. Dengan memperluas dataset yang lebih beragam, dari segi bentuk, gambar, garis, serta terutama warna tinta (misalnya merah atau biru), langkah tersebut dapat meningkatkan kecerdasan proses pengklasifikasian model CNN.
4. Nilai *Mean Squared Error* (MSE) dan *Confidence* dari model CNN dapat digabungkan menjadi satu skor kesesuaian dalam rentang 0–100. Hal ini akan membuat sistem penilaian lebih sederhana, informatif, dan mudah difahami.
5. Penilaian saat ini masih terbatas pada tiga kategori: "sangat mirip", "cukup baik", dan "perlu perbaikan". Untuk meningkatkan interaktivitas, sistem sebaiknya memberikan umpan balik langsung saat pengguna menggambar, seperti menyorot bagian garis yang meleset dan menunjukkan letak kesalahan. Penerapan teknologi seperti *OpenCV* sangat disarankan untuk mendukung analisis bentuk, gaya, dan tekanan goresan secara lebih akurat.
6. Mode manual sebaiknya memungkinkan pengguna mengatur posisi, ukuran, dan rotasi gambar panduan secara lebih leluasa. Selain itu, fitur navigasi antar langkah (*step*) juga perlu ditambahkan agar pengguna dapat mengikuti proses menggambar secara bertahap meskipun tanpa bantuan AR.
7. Pengujian sistem di masa mendatang sebaiknya melibatkan lebih banyak responden dari berbagai usia dan latar belakang pendidikan seni untuk memperoleh gambaran menyeluruh mengenai efektivitas sistem dalam konteks penggunaan nyata.

DAFTAR PUSTAKA

- [1] A. Amri, "Penerapan Model Pembelajaran Interaktif Untuk Meningkatkan Prestasi Belajar Seni Rupa Siswa Kelas XIMipa1 SMA Negeri 1 Polewali Tahun pembelajaran 2017/2018," *Jurnal Ilmiah Tarbiyah Umat*, vol. 9, no. 1, pp. 1–8, Mar. 2021, doi: 10.36915/jitu.v9i1.68.
- [2] K. Purwo, "Perjalanan Komunitas Cokro Sketsa Ponorogo," Cokro Sketsa Ponorogo, Ponorogo, Oct. 24, 2024.
- [3] H. Ristanto, "Cokro Sketsa Berhimpun di Jalan HOS Cokroaminoto Ponorogo," Jawa Pos | RADAR MADIUN. Accessed: Oct. 29, 2024. [Online]. Available: <https://radarmadiun.jawapos.com/features/801214658/cokro-sketsa-berhimpun-di-jalan-hos-cokroaminoto-ponorogo>
- [4] L. Tubino and C. Adachi, "Developing feedback literacy capabilities through an AI automated feedback tool," *ASCILITE Publications*, 2022, doi: 10.14742/apubs.2022.39.
- [5] W. C. Huang *et al.*, "A 28-nm 25.1 TOPS/W Sparsity-Aware CNN-GCN Deep Learning SoC for Mobile Augmented Reality," in *Digest of Technical Papers - Symposium on VLSI Technology*, 2022. doi: 10.1109/VLSITechnologyandCir46769.2022.9830261.
- [6] L. Zhang, "Hand-drawn sketch recognition with a double-channel convolutional neural network," *EURASIP J Adv Signal Process*, vol. 2021, no. 1, 2021, doi: 10.1186/s13634-021-00752-4.
- [7] M. S. Mohd Yazed, E. F. Ahmad Shaubari, and M. H. Yap, "A Review of Neural Network Approach on Engineering Drawing Recognition and Future Directions," *JOIV : International Journal on Informatics Visualization*, vol. 7, no. 4, 2023, doi: 10.30630/joiv.7.4.01716.
- [8] O. M. Gushchina, A. V. Ochepovsky, N. N. Rogova, and A. A. Pupykina, "Ar-powered educational application based upon convolutional neural

- network,” in *Journal of Physics: Conference Series*, 2019. doi: 10.1088/1742-6596/1278/1/012036.
- [9] M. Arif and Y. B. Pratama, “Student training program of building drawing and augmented reality in SMKN 2 Pangkalpinang,” *Community Empowerment*, vol. 8, no. 5, 2023, doi: 10.31603/ce.8375.
- [10] H. Takahashi, S. Ojima, T. Kawai, A. Yamaguchi, and Y. Omae, “ANALYSIS OF IMPACT OF DATA SCIENCE AND ARTIFICIAL INTELLIGENCE EDUCATION ON MOTIVATION AND CAREER DEVELOPMENT,” *ICIC Express Letters, Part B: Applications*, vol. 14, no. 12, 2023, doi: 10.24507/icicelb.14.12.1273.
- [11] G. Attard and Y. M. Zhornokui, “THE USE OF ARTIFICIAL INTELLIGENCE IN ESPORTS: PROSPECTS AND PROBLEMS,” *Lex Sportiva*, no. 1, 2023, doi: 10.32782/lexsportiva/2023.1.1.
- [12] S. Raschka, J. Patterson, and C. Nolet, “Machine learning in python: Main developments and technology trends in data science, machine learning, and artificial intelligence,” 2020. doi: 10.3390/info11040193.
- [13] V. Muralidharan and V. Vijayalakshmi, “ANALYSIS OF KINECT FALL DETECTION SYSTEM AND REHABILITATION GAMING EXERCISES USING AUGMENTED REALITY (AR) USER INTERFACE AND MULTI - PATH CONVOLUTIONAL NEURAL NETWORK (MP - CNN),” *J Theor Appl Inf Technol*, vol. 100, no. 18, 2022.
- [14] P. W. Aditama, I. N. W. Adnyana, and K. A. Ariningsih, “Augmented Reality Dalam Multimedia Pembelajaran,” *Prosiding Seminar Nasional Desain dan Arsitektur (SENADA)*, vol. 2, 2021.
- [15] F. E. E. Kusuma, M. B. Setyawan, and I. A. Zulkarnain, “Penerapan Teknologi Augmented Reality Berbasis Android Sebagai Media Pembelajaran Pengenalan Aksara Jawa Di Sdn 1 Sidorejo Ponorogo,” *KOMPUTEK*, vol. 3, no. 1, pp. 61–67, 2019.
- [16] F. Arena, M. Collotta, G. Pau, and F. Termine, “An Overview of Augmented Reality,” 2022. doi: 10.3390/computers11020028.

- [17] M. Nurcahyo, "Kajian peran sketsa dalam proses kreatif dan pendidikan desain (Kasus pengalaman belajar desain di era digital)," *LINTAS RUANG: Jurnal Pengetahuan dan Perancangan Desain Interior*, vol. 10, no. 2, 2022.
- [18] G. G. Maheswari, N. K. R. Astuti, and I. W. A. E. Cahyadi, "PERANCANGAN KARAKTER 3 DIMENSI 'LAIK' SEBAGAI INTELLECTUAL PROPERTY," *AMARASI: JURNAL DESAIN KOMUNIKASI VISUAL*, vol. 4, no. 01, 2023, doi: 10.59997/amarasi.v4i01.1953.
- [19] R. Manapa *et al.*, "Filter Citra Sketsa Wajah Menggunakan Deteksi Tepian Prewitt," *Jurnal Nasional Sains dan Terapan*, 2022.
- [20] L. J. Chandra, "Implementasi Deep Learning Menggunakan Convolutional Neural Network untuk Indentifikasi Jenis Bunga Berbasis Mobile Menggunakan Framework TensorFlow Lite," *E-journal Universitas Atma Jaya Yogyakarta*, 2022.
- [21] I. Salamah, S. Humairoh, and S. Soim, "Implementasi Convolutional Neural Network Pada Alat Klasifikasi Kematangan dan Ukuran Buah Nanas Berbasis Android," *INOVTEK Polbeng - Seri Informatika*, vol. 8, no. 2, 2023, doi: 10.35314/isi.v8i2.3413.
- [22] A. W. Ismail, M. Y. F. Aladin, and M. N. A. Nor'a, "Real Hand Gesture in Augmented Reality Drawing with Markerless Tracking on Mobile," *International Journal of Computing and Digital Systems*, vol. 12, no. 4, 2022, doi: 10.12785/ijcds/120186.
- [23] amar Nanda Syarif, N. Pambudiyatno, W. Utomo, J. I. Jemur Andayani No, and K. Siwalankerto Kec Wonocolo, "RANCANGAN SISTEM PRESENSI DAN REKAPITULASI JURNAL KEGIATAN OJT MENGGUNAKAN VISUAL STUDIO CODE BERBASIS WEB DI AIRNAV CABANG MATSC," *PROSIDING Seminar Nasional Inovasi Teknologi Penerbangan (SNITP) Tahun*, 2023.
- [24] D. Sari, R. Suwartika Kusumadiarti, and E. Gunawan, "Perancangan Sistem Informasi Index Penyakit Pasien Rawat Inap Menggunakan Microsoft Visual

- Studio 2010 di RSUD H. Bob Bazar, SKM,” *Jurnal Ilmiah Hospitality*, vol. 12, no. 1, 2023.
- [25] R. C. Castanyer, S. Martinez-Fernandez, and X. Franch, “Integration of convolutional neural networks in mobile applications,” in *Proceedings - 2021 IEEE/ACM 1st Workshop on AI Engineering - Software Engineering for AI, WAIN 2021*, 2021. doi: 10.1109/WAIN52551.2021.00010.
- [26] L. Frizziero, C. Leon-Cardenas, M. Freddi, A. Grassoni, and A. Liverani, “Augmented reality applied to design for disassembly assessment for a volumetric pump with rotating cylinder,” *Prod Manuf Res*, vol. 11, no. 1, 2023, doi: 10.1080/21693277.2023.2199815.
- [27] K. 'Afiifah, Z. F. Azzahra, and A. D. Anggoro, “Analisis Teknik Entity-Relationship Diagram dalam Perancangan Database Sebuah Literature Review,” *INTECH*, vol. 3, no. 2, 2022, doi: 10.54895/intech.v3i2.1682.
- [28] S. M. Pulungan, R. Febrianti, T. Lestari, N. Gurning, and N. Fitriana, “Analisis Teknik Entity-Relationship Diagram Dalam Perancangan Database,” *Jurnal Ekonomi Manajemen dan Bisnis (JEMB)*, vol. 1, no. 2, 2023, doi: 10.47233/jemb.v1i2.533.
- [29] N. Khesya, “Mengenal Flowchart dan Pseudocode Dalam Algoritma dan Pemrograman,” *Preprints (Basel)*, vol. 1, 2021.
- [30] Fauziah, “Perancangan dan Implementasi Sistem Deteksi Pergerakan Kepala, Mata dan Alis Berbasis Machine Learning,” *Perancangan Dan Implementasi Sistem Deteksi Pergerakan Kepala, Mata Dan Alis Berbasis Machine Learning*, vol. 23218302, 2021.
- [31] A. S. Tiwari, K. K. Bhagat, and G. Lampropoulos, “Designing and evaluating an augmented reality system for an engineering drawing course,” *Smart Learning Environments*, vol. 11, no. 1, 2024, doi: 10.1186/s40561-023-00289-z.
- [32] W. Kurniawan, I. Prihandi, and N. Husufa, “PROTOTYPE FIREBASE AUTHENTICATION MENGGUNAKAN FITUR FIREBASE PADA

APLIKASI ANDROID,” *JURNAL SATYA INFORMATIKA*, vol. 4, no. 1, 2023, doi: 10.59134/jsk.v4i1.406.

[33] M. Arsal, B. Agus Wardijono, and D. Anggraini, “Face Recognition Untuk Akses Pegawai Bank Menggunakan Deep Learning Dengan Metode CNN,” *Jurnal Nasional Teknologi dan Sistem Informasi*, vol. 6, no. 1, 2020, doi: 10.25077/teknosi.v6i1.2020.55-63.

[34] N. Dewi and F. Ismawan, “IMPLEMENTASI DEEP LEARNING MENGGUNAKAN CNN UNTUK SISTEM PENGENALAN WAJAH,” *Faktor Exacta*, vol. 14, no. 1, 2021, doi: 10.30998/faktorexacta.v14i1.8989.

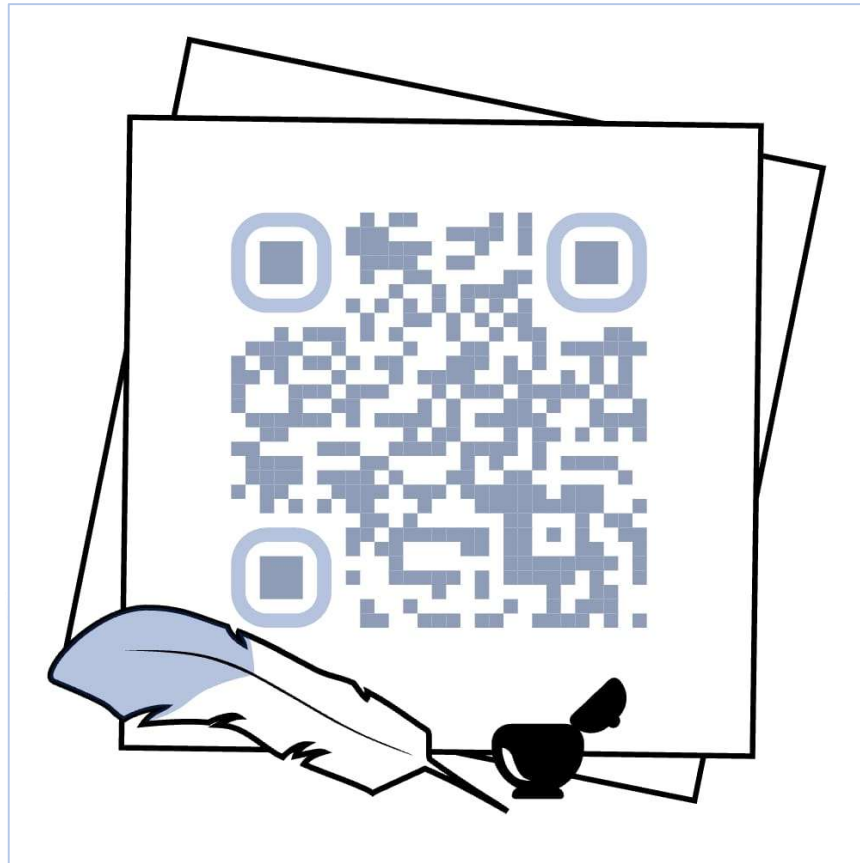


LAMPIRAN

a. Dokumentasi di tempat penelitian (Cokro Sketsa)



b. Aplikasi IMAJI



c. Kontak Penulis

* mbgigihbaskoroashari@gmail.com

* @toro.rii



d. Hasil cek Plagiasi Penelitian (Turnitin)



Page 1 of 98 - Cover Page

Submission ID trnoid::1:3299638209

PERPUSTAKAAN UMPO

Imaji-fix

KARYA ILMIAH 138

TUGAS AKHIR (Skripsi,KTI,Thesis)

Universitas Muhammadiyah Ponorogo

Document Details

Submission ID

trnoid::1:3299638209

89 Pages

Submission Date

Jul 21, 2025, 11:26 AM GMT+7

13,748 Words

Download Date

Jul 21, 2025, 11:31 AM GMT+7

91,840 Characters

File Name

BAB1-5.docx

File Size

3.4 MB



Page 1 of 98 - Cover Page

Submission ID trnoid::1:3299638209

14% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

Top Sources

- 12%  Internet sources
- 7%  Publications
- 9%  Submitted works (Student Papers)

Top Sources

12% Internet sources
7% Publications
9% Submitted works (Student Papers)

Top Sources

The sources with the highest number of matches within the submission. Overlapping sources will not be displayed.

1	Internet	elib.pnc.ac.id	1%
2	Internet	repository.ub.ac.id	<1%
3	Student papers	Universitas Nahdlatul Ulama Lampung	<1%
4	Internet	repository.amikomsolo.ac.id	<1%
5	Internet	docplayer.info	<1%
6	Internet	www.coursehero.com	<1%
7	Internet	eprintslib.ummgl.ac.id	<1%
8	Internet	tahtamedia.co.id	<1%
9	Student papers	Universitas Muhammadiyah Purwokerto	<1%
10	Internet	repo.itera.ac.id	<1%
11	Publication	Ananda Rizki Dani, Irma Handayani. "Classification of Yogyakarta Batik Motifs Usl...	<1%

12	Internet	eprints.umpo.ac.id	<1%
13	Student papers	STKIP Sumatera Barat	<1%
14	Student papers	University of Huddersfield	<1%
15	Internet	dspace.uil.ac.id	<1%
16	Publication	Doly Ilham Saputra Huta Julu, Dewi Nurdiah. "KLASIFIKASI SAMPAH ORGANIK D..."	<1%
17	Student papers	Telkom University	<1%
18	Internet	kc.umn.ac.id	<1%
19	Student papers	Universitas Negeri Surabaya	<1%
20	Internet	4programmers.net	<1%
21	Internet	adoc.pub	<1%
22	Publication	Zidan Indra Nugraha, Arnita, Kana Saputra S, Abi Setiawan, Raysa Maharani, Fir...	<1%
23	Internet	eprints.umg.ac.id	<1%
24	Internet	digilib.unila.ac.id	<1%
25	Internet	repository.uin-suska.ac.id	<1%

26	Student papers	
Universitas Andalas		<1%
27	Internet	
e-journal.upr.ac.id		<1%
28	Internet	
es.scribd.com		<1%
29	Publication	
Azizah Azizah, Setio Ardy Nuswantoro, Firman Jaya, Rahmat Shofan Razaqi, Ansor...		<1%
30	Publication	
Indah Clara Sari. "INTEGRASI MODEL DEEP LEARNING EFFICIENTNET-B0 UNTUK D...		<1%
31	Student papers	
STT PLN		<1%
32	Internet	
id.123dok.com		<1%
33	Student papers	
Udayana University		<1%
34	Internet	
d3jelex.files.wordpress.com		<1%
35	Internet	
journal.ipm2kpe.or.id		<1%
36	Internet	
repo.darmajaya.ac.id		<1%
37	Student papers	
West University Of Timisoara		<1%
38	Internet	
text-id.123dok.com		<1%
39	Internet	
123dok.com		<1%

40	Internet	
ejournal.arraayah.ac.id		<1%
41	Internet	
jurnal.stkipgritlungagung.ac.id		<1%
42	Internet	
ojs.trigunadharma.ac.id		<1%
43	Internet	
repository.amikom.ac.id		<1%
44	Student papers	
Higher Education Commission Pakistan		<1%
45	Publication	
Zurni Mardian, Sarjon Defit, Sumijan Sumijan. "Implementasi Augmented Reality ...		<1%
46	Internet	
simki.unpkediri.ac.id		<1%
47	Internet	
upi-yptk.ac.id		<1%
48	Internet	
openlibrarypublications.telkomuniversity.ac.id		<1%
49	Internet	
cris.vtt.fi		<1%
50	Internet	
ejurnal.ung.ac.id		<1%
51	Internet	
eprints.dinus.ac.id		<1%
52	Internet	
informatika.uin-suka.ac.id		<1%
53	Student papers	
Sultan Agung Islamic University		<1%

54	Student papers	Universitas Brawijaya	<1%
55	Student papers	Universitas PGRI Semarang	<1%
56	Internet	ejournal.unib.ac.id	<1%
57	Internet	www.ejournal.uui.ac.id	<1%
58	Internet	jurnal.pancabudi.ac.id	<1%
59	Internet	repository.its.ac.id	<1%
60	Internet	repository.teknokrat.ac.id	<1%
61	Internet	www.jiip.stkipyapisdompu.ac.id	<1%
62	Student papers	Konsorsium Turnitin Relawan Jurnal Indonesia	<1%
63	Internet	adoc.tips	<1%
64	Internet	eprints.itn.ac.id	<1%
65	Internet	etheses.uin-malang.ac.id	<1%
66	Internet	git.applefritter.com	<1%
67	Internet	repository.ar-raniry.ac.id	<1%

68	Internet	www.slideshare.net	<1%
69	Internet	24hour.id	<1%
70	Internet	eprints.uty.ac.id	<1%
71	Internet	intellipaat.com	<1%
72	Internet	lib.unnes.ac.id	<1%
73	Internet	repositori.usu.ac.id	<1%
74	Internet	www.kominfo.go.id	<1%
75	Publication	Fajar Mubarraq, Nazaruddin Ahmad. "Perancangan Desain Sistem Evaluasi Data ...	<1%
76	Student papers	Konsorsium Perguruan Tinggi Swasta I 2023	<1%
77	Publication	Luis Valencia-Cabrera, David Orellana-Martín, Miguel A. Martínez-del-Amor, Agus...	<1%
78	Publication	Muhammad Ulumul Ikhsanil Huda, Kustiyono. "Identification of Digital Signatur...	<1%
79	Publication	Ni Putu Irene Pasca Nurahdika, Joko Sutopo. "Analisa Performa Convolution Neur...	<1%
80	Internet	access-terapan.blogspot.com	<1%
81	Internet	digilib.uin-suka.ac.id	<1%

82	Internet	lo.traffine.com	<1%
83	Internet	jurnal-ltsi.org	<1%
84	Internet	kabar-terhangat.blogspot.com	<1%
85	Internet	repository.usni.ac.id	<1%
86	Internet	www.fstpt.or.id	<1%
87	Internet	www.hackster.io	<1%
88	Internet	www.microthings.id	<1%
89	Internet	www.scribd.com	<1%
90	Publication	Winna Dharmayanti, Dini Oktarika, "ANALISIS EFEKTIVITAS PENGEMBANGAN MO...	<1%
91	Publication	Misna Asqia, Raihan Daffa Aziz. "Web-Based New Student Registration Applicatio...	<1%
92	Internet	doku.pub	<1%
93	Internet	journal.unpas.ac.id	<1%