

**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

## BAB II

### LANDASAN TEORI

#### Penelitian Terkait

Penelitian terkait pada penelitian ini digunakan sebagai sumber acuan sebelum melakukan penelitian. Penelitian terkait dengan penelitian ini dapat dilihat pada Tabel 2.1 di bawah :

**Tabel 2.1** Penelitian terkait.

| Jurnal                                                                                            | Judul Penelitian                                                                     | Hasil                                                                                                                                                                                            | Keterbatasan                                                                                                                                                                         |
|---------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Zhang et al. (2022)</b><br><br><i>Pattern Recognition Letters</i>                              | Lightweight Face Recognition for Edge Computing based on MobileFaceNet               | Model MobileFaceNet tervalidasi sangat efisien dalam penggunaan memori pada perangkat edge, menghasilkan waktu inferensi di bawah 200 milidetik dengan akurasi pengenalan mencapai 95%.          | Pengujian sistem hanya dilakukan pada basis data wajah yang statis. Tidak ada mekanisme untuk penambahan identitas pengguna baru secara lokal tanpa melalui proses retraining penuh. |
| <b>Pratama et al. (2023)</b><br><br><i>Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)</i> | Enhancing IoT Smart Door Lock Security using Image Enhancement and Cosine Similarity | Menunjukkan bahwa penerapan peningkatan citra sebelum proses ekstraksi wajah dapat menaikkan nilai kecocokan hingga 5%, terutama dalam kondisi pencahayaan ruangan ( <i>indoor</i> ) yang redup. | Tahapan pra-pemrosesan yang diusulkan terlalu kompleks untuk perangkat berdaya rendah, menyebabkan waktu tunggu (latensi) sistem kontrol akses menjadi lambat.                       |
| <b>Wang et al. (2023)</b>                                                                         | Efficient Incremental Learning for Resource-Constrained Edge Devices                 | Membuktikan bahwa teknik pendaftaran inkremental ( <i>incremental learning</i> ) untuk menambahkan pengguna baru mampu                                                                           | Pengujian performa sistem tidak difokuskan pada skenario kontrol akses fisik ( <i>smart door lock</i> ) dan tidak mengukur keterkaitan antara latensi pendaftaran dengan metrik      |

## © Hak Cipta milik Politeknik Negeri Jakarta

### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

|                                                                          |                                                                                                              |                                                                                                                                                                                                                                                                                                                                      |                                                                                                                                                                                                                        |
|--------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <i>Proceedings of the IEEE/CVF (ICCV)</i>                                |                                                                                                              | memangkas waktu komputasi secara signifikan dibandingkan melatih ulang model secara utuh.                                                                                                                                                                                                                                            | keamanan (FAR dan FRR).                                                                                                                                                                                                |
| <b>Kumar et al. (2022)</b><br><br><i>IEEE Internet of Things Journal</i> | Edge-based Face Authentication using ArcFace and Edge-TPU                                                    | Penerapan fungsi kerugian ArcFace terbukti meningkatkan pemisahan matematis antar fitur wajah yang berbeda ( <i>inter-class discrepancy</i> ), sehingga sangat aman untuk sistem autentikasi.                                                                                                                                        | Sistem sangat bergantung pada perangkat keras akselerator tambahan (Edge-TPU) untuk mengeksekusi model, sehingga meningkatkan biaya dan kompleksitas arsitektur IoT.                                                   |
| <b>Rahman et al. (2024)</b><br><br><i>IEEE Access</i>                    | Localized Authentication for Smart Home using ESP32-CAM and Raspberry Pi                                     | Berhasil membangun arsitektur sistem keamanan rumah berbasis komputasi tepi sepenuhnya, menggunakan ESP32-CAM sebagai sumber citra dan Raspberry Pi sebagai peladen lokal.                                                                                                                                                           | Tingkat penolakan keliru ( <i>False Rejection Rate/FRR</i> ) melonjak tinggi ketika pengguna menggunakan aksesoris ringan (kacamata) atau saat pose wajah sedikit menyimpang dari pusat kamera.                        |
| <b>Usulan</b>                                                            | ANALISIS DAN IMPLEMENTASI METODE INCREMENTAL ENROLLMENT PADA SISTEM PENGENALAN WAJAH BERBASIS EDGE COMPUTING | Diharapkan mampu menghasilkan autentikasi wajah secara lokal dengan akurasi tinggi, latensi rendah, serta mendukung penambahan pengguna baru tanpa retraining. Sistem juga menerapkan preprocessing berupa face alignment, super resolution, quality assessment, CLAHE, denoising, dan sharpening untuk meningkatkan kualitas citra. | Mengatasi keterbatasan penelitian sebelumnya dengan mengintegrasikan model lightweight, preprocessing yang efisien, dan Incremental Enrollment pada perangkat edge tanpa memerlukan cloud maupun akselerator tambahan. |

**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

## 2.1 Konsep Dasar

### 2.1.1 Edge Computing

Edge Computing merupakan arsitektur komputasi yang menempatkan proses pengolahan data lebih dekat dengan sumber data atau perangkat fisik (edge), bukan di server pusat ataupun cloud (Chen dkk., 2023). Pendekatan ini muncul untuk menjawab kebutuhan sistem modern yang menuntut respons cepat, reliabilitas tinggi, serta efisiensi komunikasi data (Liu & Wang, 2024). Pada model komputasi tradisional berbasis cloud, data yang dihasilkan dari perangkat harus dikirim ke pusat pemrosesan sebelum dapat dianalisis atau diputuskan, sehingga menimbulkan latensi dan ketergantungan jaringan yang tinggi (Sharma dkk., 2022).

### 2.1.2 Pengenalan Wajah (Face Recognition)

Pengenalan wajah merupakan metode autentikasi biometrik yang memanfaatkan karakteristik wajah manusia untuk mengidentifikasi atau memverifikasi identitas seseorang. Pengenalan wajah biasanya meliputi empat tahap utama, yaitu deteksi wajah, analisis wajah, verifikasi/identifikasi wajah, dan pencocokan (Authme, 2024). Dalam sistem smart door lock, Pengenalan wajah merupakan teknologi biometrik yang digunakan untuk mengidentifikasi atau memverifikasi identitas seseorang berdasarkan karakteristik wajah.

Secara umum, sistem pengenalan wajah terdiri dari empat tahap utama:

1. Deteksi wajah (face detection)
2. Preprocessing citra
3. Ekstraksi fitur (feature embedding)
4. Pencocokan (matching)

Tahapan tersebut menghasilkan representasi wajah dalam bentuk vektor numerik yang kemudian dibandingkan dengan data yang tersimpan di dalam database menggunakan metode similarity measurement.

**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

### 2.1.3 Edge-AI / Artificial Intelligence pada Edge

Edge-AI adalah implementasi model kecerdasan buatan pada perangkat edge untuk melakukan inferensi lokal tanpa ketergantungan cloud. Implementasi ini umumnya menggunakan model lightweight seperti MobileNet, EfficientNet-Lite, atau TensorRT agar mampu berjalan pada perangkat terbatas seperti Jetson, ARM board, dan microcontroller AI. Keunggulannya ialah respon cepat, privasi lebih baik, serta efisiensi penggunaan bandwidth (Li et al., 2022).

Keunggulan Edge AI antara lain:

1. Proses real-time dengan latensi rendah
2. Tidak bergantung pada koneksi internet
3. Privasi data lebih terjaga
4. Cocok untuk perangkat embedded system

## 2.2 Deteksi dan Pengolahan Citra

### 2.2.1 Face Detection (Haarcascade)

Face detection adalah proses untuk mendeteksi keberadaan wajah dalam sebuah citra. Pada sistem pengenalan wajah, proses ini digunakan untuk menentukan region of interest (ROI) yang akan diproses lebih lanjut pada tahap preprocessing dan ekstraksi fitur. Dalam sistem autentikasi berbasis biometrik seperti smart door lock, deteksi wajah merupakan tahap awal yang penting untuk memastikan hanya area wajah yang dianalisis oleh sistem.

Pada penelitian ini digunakan metode Haarcascade yang merupakan pendekatan berbasis Haar-like features dengan classifier berlapis (cascade classifier). Metode ini bekerja dengan mendeteksi pola sederhana pada citra seperti tepi dan perubahan intensitas untuk mengidentifikasi area yang berpotensi sebagai wajah. Haarcascade dipilih karena memiliki karakteristik yang ringan, cepat, dan tidak memerlukan komputasi tinggi sehingga sesuai untuk perangkat edge computing dengan sumber

**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

daya terbatas seperti ESP32-CAM atau Raspberry Pi. Selain itu, metode ini mampu melakukan deteksi secara real-time tanpa kebutuhan GPU.

Penggunaan Haarcascade memberikan keseimbangan antara kecepatan dan akurasi untuk aplikasi smart door lock berbasis edge computing, terutama pada kondisi perangkat dengan keterbatasan sumber daya (Viola & Jones, 2001; OpenCV Documentation, 2024).

### 2.2.2 Face Alignment

Face alignment merupakan proses normalisasi posisi wajah berdasarkan titik-titik penting (facial landmarks) seperti mata, hidung, dan mulut. Proses ini bertujuan untuk menyelaraskan orientasi wajah agar berada pada posisi standar sebelum dilakukan ekstraksi fitur. Dalam sistem pengenalan wajah, variasi pose, rotasi, dan sudut pengambilan gambar dapat mempengaruhi kualitas representasi fitur. Oleh karena itu, face alignment digunakan untuk mengurangi variasi tersebut sehingga embedding yang dihasilkan menjadi lebih konsisten dan stabil.

Proses alignment umumnya dilakukan dengan mendeteksi landmark wajah kemudian menerapkan transformasi geometris agar posisi wajah berada pada orientasi yang seragam. Teknik ini terbukti meningkatkan performa sistem *face recognition* modern berbasis deep learning (Deng et al., 2021).

### 2.2.3 Image Enhancement

Image enhancement merupakan proses peningkatan kualitas citra digital untuk memperbaiki aspek visual seperti kontras, pencahayaan, noise, dan ketajaman gambar sebelum dilakukan ekstraksi fitur. Dalam sistem pengenalan wajah, kualitas citra sangat mempengaruhi akurasi embedding. Oleh karena itu, image enhancement digunakan untuk mengatasi kondisi seperti pencahayaan rendah, blur, dan noise yang umum terjadi pada kamera dengan kualitas terbatas seperti ESP32-CAM.

**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Salah satu teknik yang umum digunakan adalah Contrast Limited Adaptive Histogram Equalization (CLAHE), yang mampu meningkatkan kontras secara adaptif pada area lokal citra sehingga lebih efektif pada kondisi pencahayaan tidak merata (Li et al., 2021). Tujuan utama image enhancement adalah meningkatkan robustness sistem terhadap variasi kondisi lingkungan dan kualitas perangkat akuisisi citra.

**a. Super Resolution (FSRCNN)**

Super Resolution merupakan teknik untuk meningkatkan resolusi citra sehingga detail wajah yang semula kurang jelas menjadi lebih tajam. Pada penelitian ini digunakan metode **Fast Super-Resolution Convolutional Neural Network (FSRCNN)** yang dirancang agar memiliki kompleksitas komputasi rendah sehingga sesuai diterapkan pada perangkat edge computing. Berbeda dengan metode interpolasi konvensional, FSRCNN mempelajari hubungan antara citra beresolusi rendah dan citra beresolusi tinggi menggunakan jaringan saraf konvolusional sehingga mampu menghasilkan detail wajah yang lebih baik. Peningkatan kualitas citra ini membantu proses ekstraksi embedding menjadi lebih akurat, terutama ketika citra diperoleh dari kamera dengan resolusi terbatas seperti ESP32-CAM (Liu et al., 2024).

**b. Quality Assessment**

Quality Assessment merupakan proses untuk mengevaluasi kualitas citra wajah sebelum dilakukan ekstraksi fitur. Evaluasi dilakukan berdasarkan beberapa parameter, seperti tingkat ketajaman, pencahayaan, dan keberadaan blur. Citra yang tidak memenuhi standar kualitas akan ditolak atau diminta untuk diambil ulang sehingga hanya citra dengan kualitas memadai yang diproses lebih lanjut. Tahapan ini bertujuan untuk mengurangi kesalahan identifikasi yang disebabkan oleh kualitas citra yang buruk serta meningkatkan konsistensi proses pengenalan wajah (Boutros et al., 2022).

**c. Contrast Limited Adaptive Histogram Equalization (CLAHE)**

**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Contrast Limited Adaptive Histogram Equalization (CLAHE) merupakan metode peningkatan kontras yang bekerja secara adaptif pada setiap wilayah kecil (tile) citra. Berbeda dengan Histogram Equalization konvensional, CLAHE membatasi peningkatan kontras agar tidak memperkuat noise secara berlebihan. Teknik ini efektif digunakan pada kondisi pencahayaan yang tidak merata karena mampu memperjelas detail wajah tanpa menghasilkan artefak yang signifikan. Dengan demikian, proses ekstraksi fitur menjadi lebih stabil pada berbagai kondisi pencahayaan (Li et al., 2021).

**d. Denoising**

Denoising merupakan proses untuk mengurangi noise atau gangguan pada citra digital yang umumnya muncul akibat kualitas sensor kamera, pencahayaan yang rendah, maupun proses kompresi citra (Kumar dkk., 2023). Noise yang berlebihan dapat menghilangkan detail penting pada wajah sehingga menurunkan akurasi pengenalan. Oleh karena itu, proses denoising diterapkan untuk mempertahankan struktur utama wajah sekaligus mengurangi gangguan visual sehingga kualitas embedding yang dihasilkan menjadi lebih konsisten.

**e. Sharpening**

Sharpening merupakan proses peningkatan ketajaman citra dengan mempertegas tepi (edge) dan detail objek pada wajah (Pratama dkk., 2024). Proses ini dilakukan setelah tahap denoising untuk mengembalikan detail yang mungkin berkurang akibat proses pengurangan noise. Penerapan sharpening secara ringan mampu meningkatkan kejelasan fitur wajah seperti mata, hidung, dan mulut tanpa memperbesar noise secara signifikan. Dengan demikian, citra yang dihasilkan menjadi lebih representatif untuk proses ekstraksi embedding pada sistem pengenalan wajah.

**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

## 2.2.4 Stepwise Recognition

Stepwise recognition (pengenalan bertahap) adalah pendekatan dalam sistem pengenalan wajah (face recognition) yang memecah keseluruhan proses pengenalan menjadi beberapa langkah atau tahap yang dilakukan secara berurutan, di mana keluaran dari satu tahap menjadi masukan bagi tahap berikutnya. Pendekatan ini lahir dari kebutuhan untuk menyederhanakan masalah pengenalan wajah yang sebenarnya sangat kompleks menyangkut deteksi wajah, penyelarasan wajah (*alignment*), dan representasi wajah, di mana hasil dari satu elemen menjadi dasar bagi elemen berikutnya untuk menghasilkan fitur identitas yang dapat dikenali. Pendekatan bertahap ini lahir dari kebutuhan untuk menyederhanakan masalah pengenalan wajah yang sebenarnya sangat kompleks menyangkut variasi pose, pencahayaan, ekspresi, dan resolusi citra menjadi beberapa sub-masalah yang lebih sederhana dan dapat diselesaikan satu per satu (Wang et al., 2022).

## 2.3 Ekstraksi Fitur Wajah

### 2.3.1 Feature Embedding (MobileFaceNet + ArcFace)

Embedding wajah merupakan representasi vektor berdimensi tinggi yang menggambarkan karakteristik biometrik unik seseorang. Pada perangkat edge, model yang digunakan harus ringan namun tetap diskriminatif. MobileFaceNet dirancang sebagai CNN ringan dengan footprint kecil sehingga mampu dijalankan pada perangkat embedded atau accelerator NPU, sementara ArcFace meningkatkan separabilitas antar identitas melalui angular margin loss. Pada smart door lock, embedding digunakan sebagai dasar perbandingan identitas antara wajah input dan wajah yang tersimpan pada sistem. Model ini menghasilkan representasi wajah dalam bentuk embedding vector berdimensi 192 yang digunakan sebagai identitas numerik wajah seseorang. Arsitektur MobileFaceNet menggunakan pendekatan lightweight convolution seperti depthwise separable convolution untuk mengurangi kompleksitas

**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

komputasi.

Keunggulan MobileFaceNet:

1. Model ringan dan cocok untuk perangkat edge
2. Inferensi cepat untuk real-time system
3. Akurasi tinggi pada tugas face verification
4. Efisiensi memori dan komputasi

Model ini banyak digunakan dalam sistem Pengenalan wajah modern berbasis edge AI (Wang et al., 2020).

### 2.3.2 Feature Embedding

Feature embedding adalah proses transformasi citra wajah menjadi representasi numerik dalam bentuk vektor multidimensi. Vektor ini menyimpan karakteristik unik dari wajah seseorang. Dalam Pengenalan wajah modern, embedding digunakan untuk menggantikan citra mentah sehingga proses perbandingan dapat dilakukan menggunakan metrik matematis seperti cosine similarity. Embedding yang baik memiliki sifat bahwa wajah dari individu yang sama memiliki jarak lebih dekat dibandingkan individu yang berbeda (Wang et al., 2021).

### 2.3.3 Non-Maximum Suppression (NMS)

Non-Maximum Suppression (NMS) adalah teknik penyaringan *bounding box* (kotak pembatas) yang merupakan tahapan pasca-pemrosesan krusial dalam algoritma deteksi objek untuk memilih deteksi dengan tingkat kepercayaan (*confidence score*) tertinggi dan membuang deteksi berlebih yang saling tumpang tindih. Dalam *pipeline face recognition* (pengenalan wajah), sebuah model deteksi sering kali memprediksi beberapa *bounding box* untuk satu wajah yang sama di area target. NMS memastikan hanya wajah yang paling relevan dan presisi yang dipertahankan untuk diteruskan ke tahap *embedding*, sehingga mempercepat komputasi dan mengurangi tingkat deteksi palsu (Wang et al., 2024).

**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

### 2.3.4 SQLite (Vector Storage Sederhana untuk Embedding)

SQLite merupakan sistem manajemen basis data relasional berbasis file yang ringan dan tidak memerlukan server terpisah. Dalam sistem pengenalan wajah berbasis edge computing, SQLite digunakan untuk menyimpan data embedding wajah serta informasi identitas pengguna secara lokal pada perangkat.

Berbeda dengan vector database khusus seperti ANN-based system, SQLite digunakan sebagai solusi penyimpanan sederhana untuk data embedding dengan pendekatan exact search atau cosine similarity manual. Hal ini membuat SQLite lebih sesuai untuk perangkat dengan sumber daya terbatas seperti Raspberry Pi.

Pada aplikasi smart door lock, SQLite berperan dalam:

1. Penyimpanan data pengguna (user profile)
2. Penyimpanan embedding wajah hasil enrollment
3. Manajemen data autentikasi secara lokal
4. Mendukung proses pencocokan identitas secara cepat tanpa koneksi server eksternal

Penggunaan SQLite memberikan keunggulan berupa ringan, cepat, serta mudah diintegrasikan dengan sistem berbasis Python pada edge device (SQLite Consortium, 2024).

### 2.3.5 Cosine Thresholding

Cosine thresholding merupakan mekanisme penentuan batas kecocokan (*matching*) dari dua vektor *embedding* wajah. Untuk mengukur seberapa mirip dua wajah, sistem menghitung **Cosine Similarity**, yaitu nilai kosinus sudut antara dua vektor dalam ruang multidimensi. Nilai ini sangat tangguh untuk *face recognition* karena tidak bergantung pada magnitudo vektor, melainkan pada orientasi atau arahnya (Öztürk, 2025). Jika nilai similarity melebihi threshold tertentu, wajah dianggap cocok dengan

**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

identitas pengguna; sebaliknya, wajah dinyatakan unknown. Penentuan threshold berpengaruh pada tingkat false acceptance rate (FAR) dan false rejection rate (FRR).

### 2.3.6 Incremental enrollment

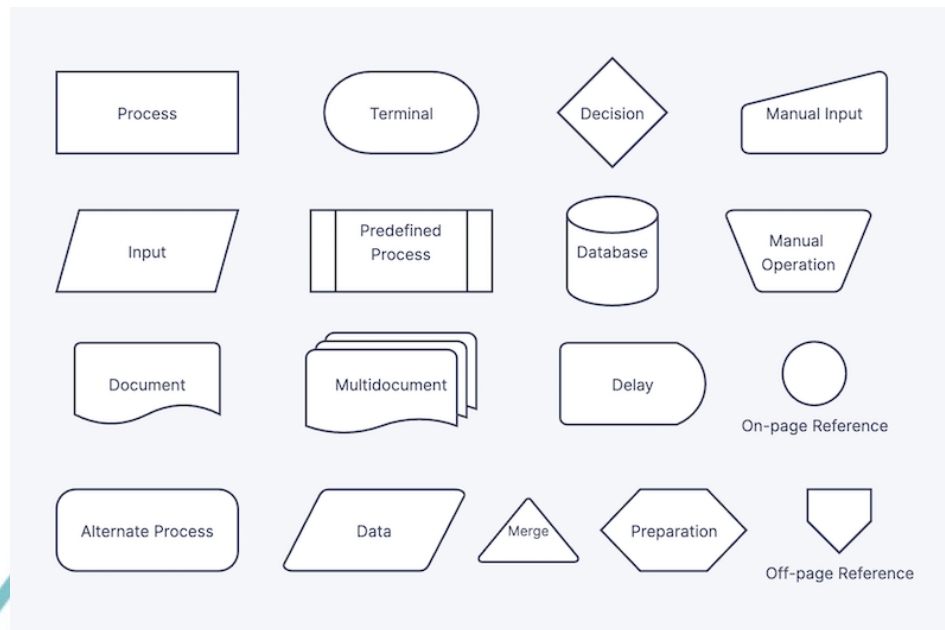
Incremental enrollment merupakan kemampuan sebuah sistem pengenalan wajah (face recognition) atau biometrik untuk menambahkan identitas pengguna baru ke dalam basis data secara bertahap tanpa perlu melakukan pelatihan ulang (retraining) pada keseluruhan model dari awal (Chen & Liu, 2024).

### 2.4 Flowchart

Flowchart adalah sebuah diagram yang secara visual menjelaskan alur proses dan logika dari suatu program atau sistem komputasi. Flowchart sangat berperan penting untuk merencanakan, mendokumentasikan, dan memutuskan setiap langkah atau fungsionalitas dari pembuatan sebuah program yang melibatkan proses bisnis suatu perusahaan. Selain itu, dengan menggunakan bagan alur proses, rancangan logika program akan terlihat lebih jelas, terstruktur, ringkas, serta mengurangi kemungkinan terjadinya salah penafsiran atau ambiguitas. Dalam ranah rekayasa perangkat lunak, penggunaan flowchart merupakan metode standar yang sangat efektif untuk menjembatani komunikasi terkait kebutuhan sistem antara tim teknis (developer) dan pihak non-teknis seperti manajemen atau pemangku kepentingan (Kusuma & Hadi, 2024).

**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



Gambar 2.1 Simbol Flowchart

(<https://www.goodnewsfromindonesia.id/2024/02/05/flowchart>)

**POLITEKNIK  
NEGERI  
JAKARTA**

**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

## BAB III

### METODE PENELITIAN

#### 3.1 Rancangan Penelitian

Penelitian ini menggunakan pendekatan kuantitatif dengan desain eksperimental (experimental research). Pendekatan kuantitatif dipilih karena penelitian ini berfokus pada pengukuran kinerja model kecerdasan buatan dalam mengukur pengaruh penerapan metode *incremental enrollment* dan *preprocessing pipeline* terhadap kinerja sistem pengenalan wajah pada perangkat edge computing. Pendekatan eksperimental dipilih karena penelitian ini menguji hubungan sebab-akibat antara perlakuan yang diberikan pada sistem (preprocessing, skema enrollment, jumlah identitas terdaftar) dan hasil yang dapat diukur secara numerik, yaitu akurasi pengenalan, latensi inferensi, serta pemanfaatan sumber daya perangkat.

Variabel yang digunakan dalam penelitian ini terdiri atas:

1. **Variabel Bebas (Independent Variable)** : Beberapa citra wajah berbeda yang ditangkap kamera ESP32-CAM
2. **Variabel Terikat (Dependent Variable)** : hasil yang diamati sebagai akibat dari perlakuan di atas: akurasi recognition, waktu proses enrollment dan recognition, serta beban CPU dan memori pada Raspberry Pi 4B selama sistem bekerja.
3. **Variabel Kontrol (Control Variable)** : agar hasil pengukuran benar-benar berasal dari perlakuan yang diuji dan tidak tercampur faktor lain, sejumlah hal dibuat tetap selama pengujian: perangkat Raspberry Pi 4B, kamera ESP32-CAM, model MobileFaceNet berbasis TensorFlow Lite, metode pencocokan cosine similarity, pipeline preprocessing (face alignment, super resolution, quality assessment, CLAHE, denoise, dan sharpening), serta kondisi pengambilan citra, yaitu jarak pengguna sekitar 50–100 cm dengan pencahayaan normal hingga cukup terang.

**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Di luar sisi eksperimen, penelitian ini juga mencakup tahap perancangan, implementasi, integrasi, dan pengujian sistem pengenalan wajah berbasis edge computing. Lewat pendekatan ini, seluruh proses dapat dipastikan berjalan langsung di Raspberry Pi 4B, mulai dari akuisisi citra, preprocessing, ekstraksi embedding, penyimpanan data melalui SQLite, hingga recognition, tanpa membutuhkan layanan cloud computing.

### 3.2 Tahapan Penelitian

Pelaksanaan penelitian ini mengacu pada tahapan sistematis yang terdiri dari sebagai berikut :

#### 1. Studi Literatur

Mengkaji teori tentang smart door lock, autentikasi biometrik, IoT, edge computing, dan model lightweight, termasuk MobileFaceNet, ArcFace, dan Incremental Enrollment, untuk menentukan metode yang relevan dipakai.

#### 2. Identifikasi Masalah dan Analisis Kebutuhan

Menentukan kebutuhan hardware dan software agar sistem berjalan di perangkat edge dengan komputasi terbatas.

1. Hardware: Raspberry Pi 4B (unit pemroses utama), ESP32-CAM (akuisisi citra), microSD/SSD (penyimpanan embedding dan log). Tidak termasuk aktuator fisik seperti relay atau kunci pintu.
2. Software: Python, OpenCV, MobileFaceNet (TFLite, pretrained, embedding 192 dimensi), SQLite untuk data identitas, PostgreSQL untuk log akses. Semua berjalan lokal di Raspberry Pi 4B tanpa cloud.

#### 3. Pengumpulan Data

Citra wajah diambil langsung dari ESP32-CAM untuk keperluan enrollment dan pengujian recognition. Karena model yang dipakai sudah pretrained, tidak ada pengumpulan data untuk melatih model dari nol.

**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

#### **4. Pra-pemrosesan Data**

Citra melewati pipeline preprocessing: face alignment, super resolution, quality check, dan image enhancement (CLAHE, denoising, sharpening), untuk memastikan kualitas citra sebelum diekstrak fiturnya.

#### **5. Pengujian dan Evaluasi Model**

Model MobileFaceNet (pretrained) diuji kemampuannya menghasilkan embedding yang akurat, diukur lewat akurasi recognition, FAR, dan FRR, dengan cosine similarity sebagai metode pencocokan.

#### **6. Implementasi ke Perangkat Edge**

Model TFLite di-deploy ke Raspberry Pi 4B. ESP32-CAM hanya mengirim citra; seluruh proses deteksi wajah (Haar Cascade), preprocessing, ekstraksi embedding, dan pencocokan berjalan di Raspberry Pi.

#### **7. Pengembangan Manajemen Data dan Logging**

Data embedding disimpan di SQLite, log akses dicatat di PostgreSQL, keduanya berjalan lokal di Raspberry Pi 4B. Tidak ada backend cloud atau aplikasi mobile.

#### **8. Integrasi dan Pengujian Sistem**

Seluruh komponen diintegrasikan dari akuisisi citra hingga pencatatan hasil. Pengujian fokus pada akurasi recognition, latensi tiap tahap, penggunaan CPU/memori, dan efektivitas Incremental Enrollment.

Kondisi pengujian: pencahayaan normal–cukup terang, jarak 50–<100 cm, tanpa kondisi ekstrem (gelap total/overexposure) dan tanpa aksesoris yang menutupi wajah. Tidak mencakup anti-spoofing, deteksi wajah 3D, deepfake detection, autentikasi multimodal, atau pengujian aktuator fisik.

**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

**9. Penyusunan Laporan**

Seluruh proses, hasil pengujian, dan analisis didokumentasikan ke dalam laporan skripsi.

**3.3 Objek Penelitian**

Objek penelitian dalam penelitian ini adalah sistem pengenalan wajah berbasis *edge computing* pada perangkat Raspberry Pi 4B dengan menggunakan model MobileFaceNet berbasis TensorFlow Lite. Sistem dirancang untuk melakukan pemrosesan secara lokal melalui tahapan akuisisi citra menggunakan ESP32-CAM. *preprocessing*, ekstraksi embedding wajah, penyimpanan embedding ke dalam database SQLite, serta proses pencocokan identitas menggunakan metode *cosine similarity*. Semua evaluasi kinerja sistem berdasarkan tingkat akurasi pengenalan, waktu proses, serta penggunaan sumber daya perangkat yang meliputi CPU dan memori pada Raspberry Pi 4B.



POLITEKNIK  
NEGERI  
JAKARTA

**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

## BAB IV

### HASIL DAN PEMBAHASAN

#### 4.1 Analisis Kebutuhan Sistem

Pada tahap ini, akan dilakukan analisis terhadap kebutuhan utama dalam membangun model klasifikasi kecerdasan buatan, yang mana mencakup kebutuhan data sebagai bahan pelatihan (training) dan kebutuhan perangkat komputasi edge untuk tahap implementasi (deployment).

##### 4.1.1 Analisis Kebutuhan Fungsional

Kebutuhan fungsional mendefinisikan kapabilitas dan layanan yang harus mampu dilakukan oleh sistem Pengenalan wajah dan Model yang dibangun. Kebutuhan fungsional pada penelitian ini meliputi:

1. Sistem harus mampu melakukan autentikasi pengguna menggunakan teknologi pengenalan wajah secara lokal pada perangkat edge computing.
2. Sistem harus mampu meningkatkan kualitas citra wajah melalui *preprocessing pipeline* sebelum proses ekstraksi embedding dilakukan agar proses pengenalan wajah menjadi lebih stabil dan akurat.
3. Sistem harus mampu menghasilkan embedding wajah menggunakan model **MobileFaceNet** berbasis **TensorFlow Lite** sehingga proses pengenalan dapat berjalan secara efisien pada perangkat dengan sumber daya terbatas.
4. Sistem harus mampu mengenali identitas pengguna dengan membandingkan embedding hasil ekstraksi terhadap embedding yang telah tersimpan pada basis data.

**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

5. Sistem harus mampu menambahkan pengguna baru melalui metode **Incremental Enrollment** tanpa memerlukan proses *retraining* model secara keseluruhan.

#### 4.1.2 Analisis Kebutuhan Non-Fungsional

Kebutuhan non-fungsional menitikberatkan pada kualitas, performa, dan batasan operasional sistem yang dibangun. Analisis kebutuhan non-fungsional pada penelitian ini meliputi:

1. **Kinerja Sistem (Performance)**

Sistem harus memiliki tingkat akurasi pengenalan wajah yang tinggi dengan nilai *precision*, *recall*, dan *F1-score* yang baik serta menghasilkan nilai *False Acceptance Rate* (FAR) dan *False Rejection Rate* (FRR) yang rendah.

2. **Kecepatan Proses (Real-time Performance)**

Sistem diharapkan mampu melakukan proses inferensi dan autentikasi dalam waktu yang singkat.

3. **Efisiensi Sumber Daya (Resource Efficiency)**

Sistem diharapkan mampu berjalan pada perangkat edge computing dengan penggunaan CPU, memori, dan kapasitas penyimpanan yang efisien.

4. **Keandalan (Reliability)**

Sistem harus mampu melakukan autentikasi secara lokal tanpa bergantung pada layanan cloud.

5. **Skalabilitas (Scalability)**

Sistem harus mampu mendukung penambahan pengguna baru melalui metode Incremental Enrollment tanpa memerlukan pelatihan ulang model

#### 4.1.3 Analisis Kebutuhan Perangkat Keras

Perangkat keras yang digunakan pada penelitian ini mencakup:

**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

1. **Edge Processing Unit — Raspberry Pi 4B**

Berfungsi sebagai platform eksekusi model face recognition serta proses incremental enrollment secara lokal.

2. **ESP32-CAM** sebagai perangkat akuisisi citra wajah, ekstraksi fitur, dan autentikasi pengguna.

3. **Media Penyimpanan Lokal — microSD / SSD**

Digunakan sebagai penyimpanan embedding wajah, logging proses inferensi, serta dataset incremental.

Spesifikasi perangkat keras dipilih berdasarkan pertimbangan efisiensi daya, kinerja inferensi, serta kemampuan operasional dalam kondisi lingkungan rumah maupun perkantoran.

**4.1.4 Analisis Kebutuhan Perangkat Lunak**

Perangkat lunak merupakan komponen utama dalam perancangan software architecture pada penelitian ini, meliputi:

1. **Python** sebagai bahasa pemrograman utama
2. **OpenCV** untuk pemrosesan citra
3. **MobileFaceNet+Arcface pretrain (TFLite)** untuk ekstraksi embedding wajah
4. **SQLite** untuk penyimpanan data identitas dan embedding
5. **PostgreSQL** untuk pencatatan log akses sistem

Kedua kebutuhan sistem tersebut merupakan fondasi implementatif yang diperlukan agar sistem dapat beroperasi secara fungsional dan andal pada perangkat dengan keterbatasan komputasi.

**4.2 Perancangan Sistem**

Perancangan sistem di fokuskan pada tahapan pengolahan model AI menjadi *light weight model* yang sesuai untuk di jalankan pada perangkat Edge. Sistem dirancang

### Hak Cipta :

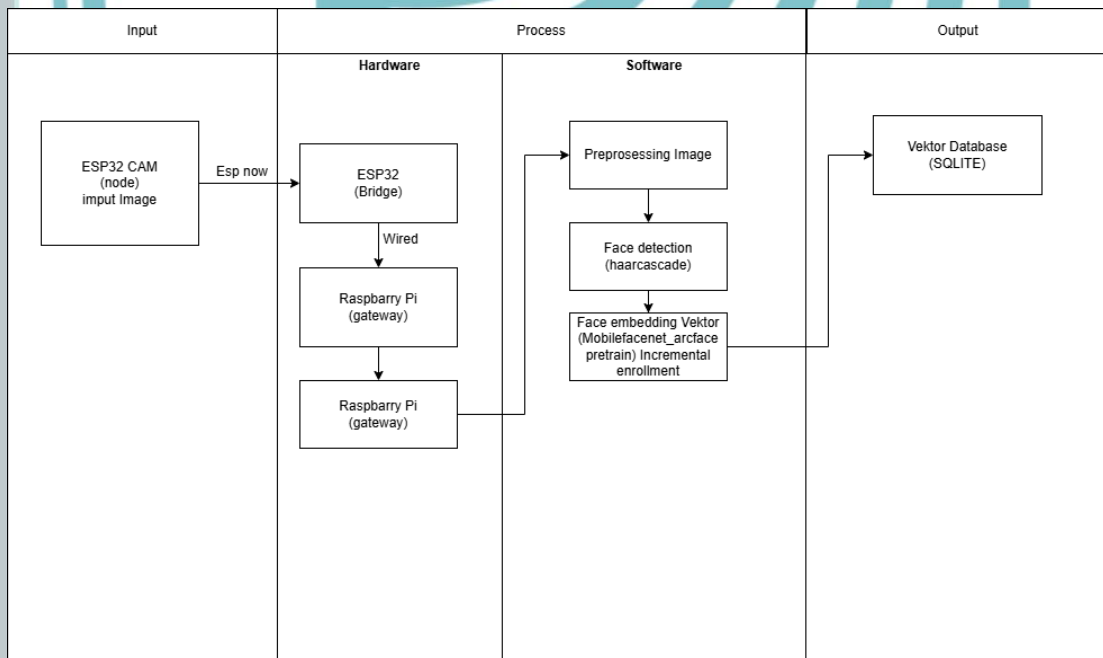
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

agar seluruh proses pengolahan data dilakukan secara lokal. Selain itu, sistem menerapkan metode pengolahan data gambar yang memberikan beberapa jenis tipe Vektor embedding.

#### 4.2.1 Blok Diagram

Blok diagram ini merupakan representasi atau gambaran sistem yang disederhanakan untuk mempermudah analisis. Membantu menggambarkan bagaimana sistem bekerja secara garis besar.

##### A. Blok Diagram Sistem Incremental Enrollment



Gambar 4.1 Blok Diagram Sistem Incremental Enrollment

Pada Gambar 4.1 menunjukkan blok diagram sistem yang menggambarkan alur kerja proses penyimpanan data vektor embedding secara lokal pada perangkat edge. Adapun fungsi masing-masing komponen pada blok diagram adalah sebagai berikut:

**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

1. ESP32 Camera (Akuisisi Citra)  
Kamera berfungsi untuk menangkap citra wajah pengguna
2. ESP32 (Bridge)  
ESP32 Bridge menerima gambar dari ESP32 Camera menggunakan komunikasi ESPNOW. ESP32 ini digunakan sebagai penghubung antara ESP32 Camera dengan Raspberry Pi.
3. *Raspberry Pi*  
Raspberry Pi menerima gambar yang nantinya akan digunakan untuk Sistem Pengenalan wajah.
4. *Pre-Processing Image*  
*Pre-Processing* dilakukan sebelum gambar digunakan untuk proses deteksi wajah agar ragam Vektor embedding dimiliki lebih banyak dan bervariasi.
5. *Face Detection*  
Dimana proses ini memungkinkan sistem untuk mencari dan mendeteksi wajah sehingga dapat memilah gambar yang wajah nya terlihat jelas atau tidak.
6. Face Embedding Vektor  
Face Embedding Vektor ini menggunakan model Mobile Facenet dengan *pretrain* Arcface yang dapat menyimpan Vektor wajah yang ada yang nantinya digunakan untuk Enrollment maupun recognition.
7. Vektor Database  
Proses Incremental enrollment yang mana Vektor database digunakan untuk menyimpan seluruh identitas maupun vektor yang didapat sehingga proses pengenalan wajah nantinya dapat digunakan secara lokal.

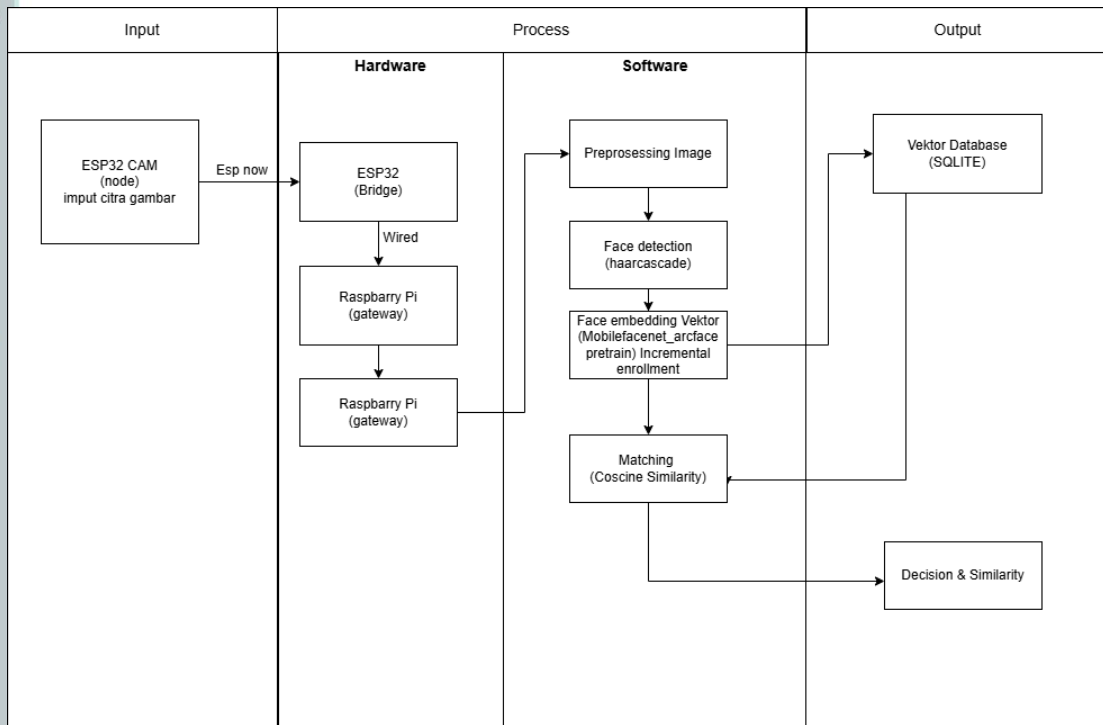
Mekanisme incremental enrollment memungkinkan penambahan embedding baru tanpa retraining model global. Vektor embedding yang tersimpan digunakan sebagai referensi verifikasi identitas.

**B. Blok Diagram Sistem Pengenalan Wajah**

### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Rancang Bangun Sistem Smart Door Lock pengenalan wajah berbasis Edge Computing dapat dilihat pada gambar berikut ini.



Gambar 4.2 Blok Diagram Sistem Pengenalan Wajah

Pada Gambar 4.2 menunjukkan blok diagram sistem yang menggambarkan alur kerja proses autentikasi berbasis pengenalan wajah yang dijalankan secara lokal pada perangkat edge. Adapun fungsi masing-masing komponen pada blok diagram adalah sebagai berikut:

1. *ESP32 Camera (Akuisisi Citra)*  
Kamera berfungsi untuk menangkap citra wajah pengguna
2. *ESP32 (Bridge)*  
ESP32 Bridge menerima gambar dari ESP32 Camera menggunakan komunikasi ESPNOW. ESP32 ini digunakan sebagai penghubung antara ESP32 Camera dengan Raspberry Pi.
3. *Raspberry Pi*

**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Raspberry Pi menerima gambar yang nantinya akan digunakan untuk Sistem Pengenalan wajah.

4. *Pre-Processing Image*

*Pre-Processing* dilakukan sebelum gambar digunakan untuk proses deteksi wajah agar ragam Vektor embedding dimiliki lebih banyak dan bervariasi.

5. *Face Detection*

Dimana proses ini memungkinkan sistem untuk mencari dan mendeteksi wajah sehingga dapat memilah gambar yang wajah nya terlihat jelas atau tidak.

6. Face Embedding Vektor

Face Embedding Vektor ini menggunakan model Mobile Facenet dengan *pretrain* Arcface yang dapat menyimpan Vektor wajah yang ada yang nantinya digunakan untuk Enrollment maupun recognition.

7. Vektor Database

Proses Incremental enrollment yang mana database digunakan untuk menyimpan seluruh identitas maupun vektor yang didapat sehingga proses pengenalan wajah nantinya dapat digunakan secara lokal.

8. Matching (Similarity Computation)

Embedding hasil inferensi dibandingkan dengan embedding yang tersimpan pada basis data menggunakan similarity search cosine similarity threshold. Proses pencocokan ini menentukan nilai kemiripan antara embedding input dan identitas pengguna yang telah ada. Jika nilai kemiripan melewati ambang batas (threshold), sistem mengidentifikasi bahwa pengguna terverifikasi. Keputusan hasil verifikasi digunakan untuk menentukan apakah akses diberikan atau ditolak.

9. Decision & Similarity

Hasil dari pengenalan wajah yang menghasilkan persentase kemiripan dan juga hasil dari matching yang memberikan nama pengguna paling mirip atau Unknown.

**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Secara keseluruhan, blok diagram menunjukkan proses dari pemerolehan citra gambar sampai hasil proses pengenalan wajah.

#### 4.2.2 Flowchart Sistem

Selain blok diagram, diperlukan juga pemodelan alur sistem untuk mendeskripsikan secara spesifik bagaimana siklus perancangan alur penggunaan dari awal hingga akhir. Perancangan ini bertujuan untuk mempermudah implementasi agar sesuai dengan alur yang diinginkan.

##### A. Flowchart Sistem Incremental Enrollment

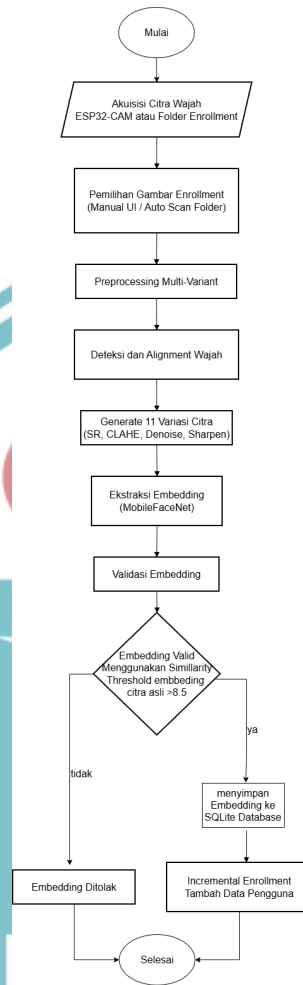
Flowchart sistem pengenalan wajah menggambarkan alur proses identifikasi pengguna yang telah terdaftar pada database. Seluruh proses dilakukan secara lokal pada perangkat edge computing untuk menjaga keamanan data dan mengurangi latensi sistem. Berikut merupakan gambar dari Flowchart Sistem Incremental Enrollment dan untuk gambar lebih jelas dapat di lihat pada lampiran 1 :



**POLITEKNIK  
NEGERI  
JAKARTA**

### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



Gambar 4.3 Flowchart Sistem Incremental enrollment

Alur proses Incremental Enrollment yang ditunjukkan pada Gambar 4.3 diawali dengan tahap capture gambar wajah, yaitu pengambilan citra menggunakan ESP32-CAM atau membaca citra yang telah tersedia pada folder enrollment. Selanjutnya, citra memasuki tahap pemilihan gambar enrollment, di mana pengguna dapat memilih gambar secara manual melalui antarmuka

Setelah gambar dipilih, sistem menjalankan preprocessing multi-variant untuk meningkatkan kualitas citra sebelum dilakukan ekstraksi fitur. Pada tahap ini dilakukan proses deteksi dan alignment wajah untuk memastikan area wajah berada pada posisi yang konsisten. Selanjutnya sistem menghasilkan beberapa variasi citra

**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

menggunakan teknik enhancement seperti Super Resolution (FSRCNN), Contrast Limited Adaptive Histogram Equalization (CLAHE), denoise, dan sharpen. Hasil dari proses ini berupa sebelas variasi citra wajah yang digunakan untuk meningkatkan robustness terhadap perbedaan kualitas gambar.

Setiap variasi citra kemudian diproses menggunakan model MobileFaceNet untuk menghasilkan embedding wajah berdimensi 192. Embedding yang dihasilkan merupakan representasi numerik dari karakteristik unik wajah pengguna.

Sebelum disimpan ke basis data, embedding yang dihasilkan terlebih dahulu melalui proses validasi embedding dimana vektor yang akan disimpan dibandingkan dengan vektor dari gambar asli sebelum dilakukan *preprocessing*. Sistem melakukan pemeriksaan terhadap kualitas citra, konsistensi embedding, serta parameter validasi lainnya.

Apabila embedding tidak memenuhi kriteria yang ditentukan, embedding tersebut akan ditolak dan tidak disimpan ke dalam basis data. Sebaliknya, apabila embedding dinyatakan valid, sistem akan menyimpan embedding tersebut ke dalam database SQLite bersama informasi identitas pengguna.

Setelah proses penyimpanan selesai, identitas baru berhasil ditambahkan ke dalam sistem melalui mekanisme Incremental Enrollment tanpa perlu melakukan retraining model. Dengan pendekatan ini, sistem dapat memperluas jumlah pengguna secara dinamis dengan biaya komputasi yang rendah sehingga sesuai untuk diterapkan pada perangkat edge computing.

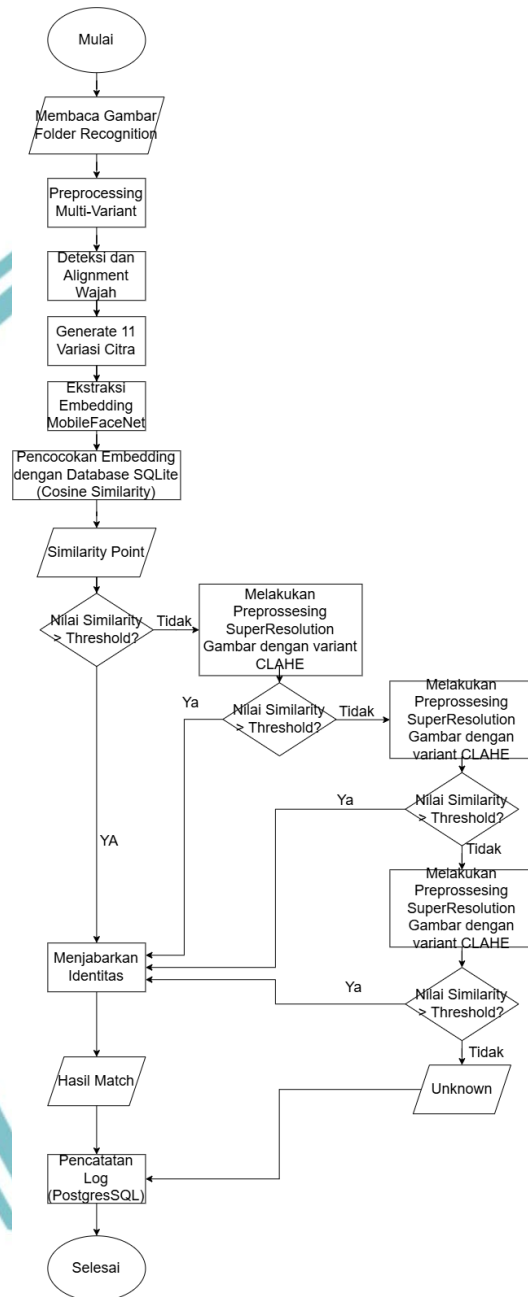
**B. Flowchart Sistem Pengenalan Wajah**

Flowchart Incremental Enrollment menggambarkan alur proses penambahan identitas pengguna baru ke dalam sistem pengenalan wajah tanpa melakukan pelatihan ulang (retraining) model. Proses ini merupakan inti dari metode Incremental Enrollment

### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

yang digunakan dalam penelitian. Berikut merupakan gambar dari Flowchart Sistem pengenalan wajah dan untuk gambar lebih jelas dapat di lihat pada lampiran 2 :



Gambar 4.4 Flowchart Sistem Pengenalan Wajan

Alur sistem pengenalan wajah yang ditunjukkan pada Gambar 4.4 diawali dengan pembacaan citra wajah yang berada pada folder recognition. Selanjutnya, citra

**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

diproses menggunakan preprocessing multi-variant yang sama dengan proses enrollment untuk memastikan konsistensi kualitas data masukan sebelum memasuki tahap ekstraksi fitur wajah.

Pada tahap preprocessing, sistem melakukan deteksi dan alignment wajah untuk memperoleh area wajah yang siap diproses. Selanjutnya sistem menghasilkan sebelas variasi citra melalui kombinasi beberapa teknik enhancement seperti Super Resolution (FSRCNN), CLAHE, denoise, dan sharpen. Penggunaan multi-variant bertujuan meningkatkan peluang keberhasilan pengenalan wajah pada berbagai kondisi pencahayaan, kualitas kamera, maupun tingkat noise yang berbeda.

Setiap variasi citra diproses menggunakan model MobileFaceNet untuk menghasilkan embedding wajah berdimensi 192. Embedding yang diperoleh kemudian dibandingkan dengan seluruh embedding yang tersimpan pada database SQLite menggunakan metode Cosine Similarity.

Hasil perhitungan similarity dibandingkan dengan nilai threshold yang telah ditentukan. Jika nilai similarity tertinggi melebihi threshold, sistem menyatakan bahwa identitas pengguna berhasil ditemukan dan proses pengenalan wajah dinyatakan berhasil. Sebaliknya, apabila nilai similarity belum memenuhi threshold, sistem akan mencoba melakukan pencocokan menggunakan variasi citra berikutnya.

Proses ini berlangsung secara bertahap (stepwise matching) hingga seluruh variasi citra selesai diperiksa. Jika tidak ditemukan satupun embedding yang memenuhi nilai threshold, sistem akan memberikan hasil Unknown, yang menunjukkan bahwa wajah pengguna tidak ditemukan dalam database.

Setelah keputusan akhir diperoleh, baik berupa identitas pengguna yang berhasil dikenali maupun status Unknown, sistem akan mencatat seluruh hasil proses ke dalam database PostgreSQL sebagai log aktivitas. Informasi yang dicatat meliputi

**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

waktu akses, hasil pencocokan, nilai similarity, serta informasi kualitas citra yang digunakan selama proses pengenalan wajah. Melalui mekanisme tersebut, sistem mampu melakukan proses identifikasi secara otomatis, cepat, dan efisien pada perangkat edge computing tanpa ketergantungan terhadap layanan cloud atau server eksternal.

#### 4.3 Implementasi Sistem Pengenalan Wajah

Pada tahap ini, akan dijelaskan proses implementasi sistem pengenalan wajah berbasis edge computing yang telah dirancang pada bab sebelumnya. Implementasi dilakukan pada perangkat Raspberry Pi 4B sebagai edge device dengan ESP32-CAM sebagai sumber akuisisi citra. Seluruh proses pengolahan data, mulai dari preprocessing citra, ekstraksi fitur wajah, penyimpanan embedding, hingga proses identifikasi dilakukan secara lokal tanpa memanfaatkan layanan cloud.

Implementasi sistem difokuskan pada penerapan metode Incremental Enrollment yang memungkinkan penambahan identitas baru ke dalam sistem tanpa melakukan pelatihan ulang model pengenalan wajah. Dengan pendekatan tersebut, sistem dapat beradaptasi terhadap penambahan pengguna baru secara dinamis dengan tetap mempertahankan performa pengenalan wajah.

Secara umum implementasi sistem terdiri atas beberapa komponen utama, yaitu:

1. Implementasi model pengenalan wajah.
2. Evaluasi model Ai
3. Implementasi preprocessing multi-variant.
4. Implementasi ekstraksi embedding menggunakan MobileFaceNet+Arcface (pretrain model).
5. Implementasi penyimpanan embedding pada SQLite.
6. Implementasi proses recognition.
7. Implementasi pencatatan log sistem.

### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

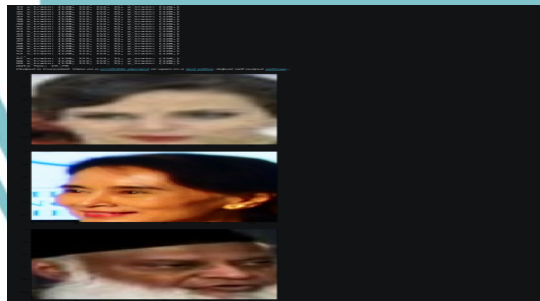
## 4.3.1 Implementasi Model Pengenalan Wajah

Pada penelitian ini, sistem pengenalan wajah menggunakan model MobileFaceNet yang telah dilatih menggunakan pendekatan ArcFace sebagai fungsi loss. Pemilihan MobileFaceNet didasarkan pada karakteristiknya yang ringan sehingga sesuai untuk implementasi pada perangkat edge computing seperti Raspberry Pi 4B. Model yang telah dilatih kemudian dikonversi ke format TensorFlow Lite (TFLite) agar dapat digunakan pada proses inferensi dengan konsumsi sumber daya yang lebih rendah.

Tahap implementasi model meliputi persiapan dataset, proses pelatihan menggunakan MobileFaceNet dan ArcFace, evaluasi performa model, serta konversi model ke format TensorFlow Lite untuk digunakan pada sistem pengenalan wajah yang dikembangkan.

### 4.3.1.1 Dataset Pelatihan

Proses pelatihan model dilakukan menggunakan dataset wajah MS1M-Refine-V2 ([faces\\_emore.zip](https://github.com/stephenliu/e2f500ee) [免费高速下载|百度网盘-分享无限制](https://www.baidu.com/s?wd=faces_emore.zip)) yang merupakan salah satu dataset face recognition berskala besar yang banyak digunakan dalam penelitian pengenalan wajah.



Gambar 4.5 Dataset MS1M-Refine-V2

Berdasarkan Gambar 4.5, dataset yang digunakan pada proses pelatihan model adalah MS1M-Refine-V2. Dataset ini terdiri atas 5.822.653 citra wajah yang berasal dari

**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

85.742 identitas berbeda. Seluruh citra telah melalui proses data cleaning sehingga kualitas label identitas yang digunakan relatif lebih baik dibandingkan dengan dataset generasi sebelumnya.

Pada penelitian ini, dataset MS1M-Refine-V2 telah dikonversi ke dalam format TFRecord sebelum digunakan pada proses pelatihan. Penggunaan format TFRecord bertujuan untuk meningkatkan efisiensi pembacaan data selama proses pelatihan model sehingga waktu akses data menjadi lebih cepat dan proses training dapat berjalan dengan lebih optimal.

Tahapan preprocessing yang diterapkan pada dataset pelatihan, Setiap citra wajah dinormalisasi ke ukuran 112×112 piksel sesuai kebutuhan MobileFaceNet serta dilakukan augmentasi sederhana berupa horizontal flipping.

Kode Program 4.1 Proses Parsing Dataset TFRecord

```
def parse_function(example_proto, n_classes=85742):
    features = {'image_raw': tf.io.FixedLenFeature([], tf.string),
                'label': tf.io.FixedLenFeature([], tf.int64)}
    features = tf.io.parse_single_example(example_proto, features)
    img = tf.image.decode_jpeg(features['image_raw'])
    img = tf.reshape(img, shape=(112, 112, 3))

    img = tf.cast(img, dtype=tf.float32)
    img = tf.subtract(img, 128)
    img = tf.multiply(img, 0.0078125)
    img = tf.image.random_flip_left_right(img)
    label = tf.cast(features['label'], tf.int64)
    label = tf.one_hot(label, n_classes)
    return (img, label), label

file_dataset =
tf.data.TFRecordDataset.list_files("/kaggle/input/faces-ms1m-refine-v2-112x112-tf
record/faces_ms1m_refine_v2_112x112-*.tfrecord")

train_dataset =
tf.data.TFRecordDataset(file_dataset,num_parallel_reads=tf.data.AUTOTUNE)
```

**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```
train_dataset =  
train_dataset.map(parse_function,num_parallel_calls=tf.data.AUTOTUNE)  
train_dataset = train_dataset.batch(GLOBAL_BATCH_SIZE)  
train_dataset = train_dataset.prefetch(tf.data.AUTOTUNE)
```

Kode Program 4.1 menunjukkan proses pembacaan dataset TFRecord yang terdiri dari citra wajah dan label identitas. Pada tahap ini dilakukan normalisasi nilai piksel serta augmentasi data menggunakan horizontal flip untuk meningkatkan kemampuan generalisasi model terhadap variasi posisi wajah.

#### 4.3.2 Implementasi Arsitektur MobileFaceNet dan ArcFace

Pada penelitian ini, model pengenalan wajah dibangun menggunakan arsitektur MobileFaceNet yang dikombinasikan dengan fungsi loss ArcFace. Pemilihan MobileFaceNet didasarkan pada kemampuannya menghasilkan representasi wajah (embedding) yang akurat dengan jumlah parameter yang relatif kecil, sehingga sesuai untuk diimplementasikan pada perangkat edge seperti Raspberry Pi 4. Sementara itu, ArcFace digunakan selama proses pelatihan untuk meningkatkan kemampuan model dalam membedakan identitas wajah yang memiliki karakteristik serupa.

Kode Program 4.2 Implementasi ArcFace Loss

```
def call(self, inputs):  
    embedding, labels = inputs  
  
    # L2 Normalization  
    embedding = tf.divide(  
        embedding,  
        tf.norm(embedding, axis=1, keepdims=True)  
    )  
  
    weights = tf.divide(  
        self.W,  
        tf.norm(self.W, axis=0, keepdims=True)  
    )
```

**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```
# Cosine Similarity
cos_t = tf.matmul(embedding, weights)

# Angular Margin
cos_mt = self.s * (
    tf.multiply(cos_t, math.cos(self.m))
    - tf.multiply(tf.sqrt(1.0 - tf.square(cos_t)),
        math.sin(self.m))
)

# Softmax Output
logit = tf.where(labels > 0, cos_mt, self.s * cos_t)
return tf.nn.softmax(logit)
```

Pada kode program 4.2 terdapat beberapa informasi yang hilang dan dapat diperjelas dengan melihat lampiran 3, yang menjelaskan kelas ArcFace dibangun sebagai custom layer TensorFlow yang menerima dua masukan, yaitu embedding hasil ekstraksi fitur dan label identitas. Sebelum proses klasifikasi dilakukan, embedding dan bobot kelas dinormalisasi menggunakan L2 Normalization sehingga seluruh vektor memiliki panjang yang sama.

Selanjutnya dilakukan perhitungan nilai cosine similarity antara embedding dan bobot kelas. ArcFace kemudian menambahkan nilai margin sudut sebesar 0,5 radian ( $m = 0.50$ ) untuk memperbesar separasi antar kelas. Hasil perhitungan tersebut digunakan untuk menghasilkan nilai probabilitas klasifikasi melalui fungsi Softmax. Dengan pendekatan ini, model dapat menghasilkan embedding wajah yang lebih diskriminatif sehingga meningkatkan performa proses verifikasi dan identifikasi wajah.

Untuk menghasilkan model yang ringan namun tetap mampu mengekstraksi fitur wajah secara efektif, MobileFaceNet memanfaatkan struktur Inverted Residual Block yang merupakan pengembangan dari MobileNetV2.

Kode Program 4.3 Implementasi Inverted Residual Block

```
def correct_pad(inputs, kernel_size):
    input_size = tf.keras.backend.int_shape(inputs)[1:3]
    adjust = (1 - input_size[0] % 2, 1 - input_size[1] % 2)
    correct = (kernel_size // 2, kernel_size // 2)
    return ((correct[0] - adjust[0], correct[0]),
            (correct[1] - adjust[1], correct[1]))

def inverted_res_block(inputs, expansion, stride, filters, block_id):

    # Expand
    x = Conv2D(expansion * inputs.shape[-1], 1, use_bias=False)(inputs)
    x = BatchNormalization()(x)

    # Depthwise Convolution
    x = DepthwiseConv2D(
        3,
        strides=stride,
        padding='same' if stride == 1 else 'valid'
    )(x)
    x = BatchNormalization()(x)
    x = ReLU(6.0)(x)

    # Project
    x = Conv2D(filters, 1, use_bias=False)(x)
    x = BatchNormalization()(x)

    # Residual Connection
    if stride == 1 and inputs.shape[-1] == filters:
        x = Add()(inputs, x)

    return x
```

Fungsi `correct_pad()` digunakan untuk menghitung kebutuhan padding secara otomatis ketika ukuran citra berubah akibat proses konvolusi. Perhitungan ini bertujuan agar ukuran feature map tetap sesuai dengan konfigurasi kernel dan stride yang digunakan.

**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Pada Kode Program 4.3, fungsi `inverted_res_block()` membangun blok utama MobileFaceNet yang terdiri atas tiga tahapan, yaitu `expand`, `depthwise convolution`, dan `project`. Tahap `expand` menggunakan konvolusi  $1 \times 1$  untuk memperluas jumlah kanal fitur, kemudian `depthwise convolution` melakukan ekstraksi fitur pada setiap kanal secara terpisah sehingga kompleksitas komputasi menjadi lebih rendah dibandingkan konvolusi standar. Selanjutnya, tahap `project` menggunakan konvolusi  $1 \times 1$  untuk memproyeksikan kembali jumlah kanal sesuai dengan kebutuhan arsitektur.

Selain itu, pada kondisi tertentu diterapkan `residual connection` yang menghubungkan input dengan output blok apabila jumlah kanal sama dan nilai `stride` bernilai 1. Koneksi residual membantu mempertahankan informasi penting selama proses propagasi sekaligus mengurangi risiko degradasi performa pada jaringan yang lebih dalam. Dengan struktur tersebut, MobileFaceNet mampu menghasilkan model dengan jumlah parameter yang relatif kecil sehingga sesuai untuk diimplementasikan pada perangkat `edge` dengan sumber daya komputasi terbatas.

Kode Program 4.3 yang ditampilkan pada bagian ini merupakan cuplikan implementasi yang memuat tahapan utama Inverted Residual Block. Implementasi secara lengkap beserta seluruh parameter dan konfigurasi program disajikan pada Lampiran 4.

Setelah fungsi `ArcFace` dan Inverted Residual Block dibangun, tahap berikutnya adalah menyusun keseluruhan arsitektur MobileFaceNet yang digunakan pada penelitian ini.

**Kode Program 4.4 Implementasi MobileFaceNet**

```
def mobilefacenet_arcface(n_classes=85742):  
  
    # Input
```

**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```
input = Input(shape=(112, 112, 3), name='input')
label = Input(shape=(n_classes), name='label')

# Feature Extraction
x = Conv2D(64, (3, 3), strides=2, padding='same')(input)
x = DepthwiseConv2D((3, 3), padding='same')(x)

# Inverted Residual Blocks
x = inverted_res_block(x, 2, 2, 64, 0)
x = inverted_res_block(x, 4, 2, 128, 1)
x = inverted_res_block(x, 2, 1, 128, 2)

# Global Feature Representation
x = DepthwiseConv2D((7, 7), padding='valid')(x)
x = Conv2D(128, (1, 1), padding='same')(x)

# Embedding
x = Flatten()(x)
embedding = Lambda(
    lambda k: tf.keras.backend.l2_normalize(k, axis=-1),
    name="embedding"
)(x)

# ArcFace Loss
output = ArcFace(n_classes=n_classes)([embedding, label])

return Model([input, label], output)
```

Kode Program 4.4 yang ditampilkan pada bagian ini merupakan cuplikan implementasi arsitektur MobileFaceNet yang memuat tahapan utama proses ekstraksi fitur dan pembentukan embedding. Implementasi secara lengkap beserta konfigurasi setiap lapisan jaringan disajikan pada Lampiran 5.

Arsitektur MobileFaceNet menerima citra wajah berukuran  $112 \times 112$  piksel sebagai masukan. Citra tersebut diproses melalui lapisan Convolution, Depthwise Convolution, dan beberapa Inverted Residual Block untuk mengekstraksi karakteristik wajah secara bertahap. Pada bagian akhir jaringan digunakan Depthwise Convolution berukuran  $7 \times 7$  dan Convolution  $1 \times 1$  untuk menghasilkan representasi fitur yang lebih ringkas tanpa menghilangkan informasi penting.

**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Selanjutnya, hasil ekstraksi fitur diratakan (flatten) dan dinormalisasi menggunakan L2 Normalization sehingga menghasilkan embedding wajah yang memiliki panjang vektor yang seragam. Embedding ini digunakan sebagai representasi numerik identitas wajah selama proses pelatihan model.

Pada tahap pelatihan, embedding diteruskan ke lapisan ArcFace sebagai fungsi loss untuk meningkatkan kemampuan model dalam menghasilkan representasi wajah yang lebih diskriminatif. Setelah proses pelatihan selesai, lapisan ArcFace tidak digunakan lagi karena hanya berperan pada tahap optimasi model. Model yang diekspor ke format TensorFlow Lite hanya menghasilkan embedding wajah yang kemudian dibandingkan dengan embedding yang tersimpan pada basis data SQLite menggunakan metode cosine similarity selama proses pengenalan wajah.

Setelah proses pelatihan selesai dilakukan menggunakan dataset MS1M-Refine-V2 dan metode ArcFace sebagai fungsi loss, diperoleh model MobileFaceNet yang mampu menghasilkan representasi wajah (face embedding) dengan tingkat diskriminasi yang tinggi. Proses pelatihan dilakukan hingga mencapai **326.000 global step** dengan nilai learning rate akhir sebesar **0,0001**.

Berdasarkan log pelatihan, nilai loss pada tahap akhir berada pada rentang **3,26–3,33**. Nilai tersebut menunjukkan bahwa proses optimasi telah mencapai kondisi yang relatif stabil dan model mampu mempelajari karakteristik wajah dari dataset pelatihan secara efektif.

### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

|                                                                                     |             |                     |           |                     |                        |
|-------------------------------------------------------------------------------------|-------------|---------------------|-----------|---------------------|------------------------|
| Training: 2026-04-12 00:45:47,717-Speed 9528.25 samples/sec                         | Loss 3.3268 | LearningRate 0.0001 | Epoch: 19 | Global Step: 325980 | Fp16 Grad Scale: 32768 |
| Training: 2026-04-12 00:45:48,801-Speed 9451.37 samples/sec                         | Loss 3.2652 | LearningRate 0.0001 | Epoch: 19 | Global Step: 325990 | Fp16 Grad Scale: 32768 |
| Training: 2026-04-12 00:45:49,871-Speed 9570.21 samples/sec                         | Loss 3.3027 | LearningRate 0.0001 | Epoch: 19 | Global Step: 326000 | Fp16 Grad Scale: 32768 |
| Training: 2026-04-12 00:46:11,846-[lfw][326000]XNorm: 6.548740                      |             |                     |           |                     |                        |
| Training: 2026-04-12 00:46:11,847-[lfw][326000]Accuracy-Flip: 0.99650+-0.00263      |             |                     |           |                     |                        |
| Training: 2026-04-12 00:46:11,847-[lfw][326000]Accuracy-Highest: 0.99750            |             |                     |           |                     |                        |
| Training: 2026-04-12 00:46:37,252-[cfp_fp][326000]XNorm: 5.721344                   |             |                     |           |                     |                        |
| Training: 2026-04-12 00:46:37,253-[cfp_fp][326000]Accuracy-Flip: 0.97186+-0.00892   |             |                     |           |                     |                        |
| Training: 2026-04-12 00:46:37,253-[cfp_fp][326000]Accuracy-Highest: 0.97543         |             |                     |           |                     |                        |
| Training: 2026-04-12 00:46:59,122-[agedb_30][326000]XNorm: 6.378732                 |             |                     |           |                     |                        |
| Training: 2026-04-12 00:46:59,122-[agedb_30][326000]Accuracy-Flip: 0.97217+-0.00803 |             |                     |           |                     |                        |
| Training: 2026-04-12 00:46:59,122-[agedb_30][326000]Accuracy-Highest: 0.97417       |             |                     |           |                     |                        |

Gambar 4.6 Training output model

Berdasarkan Gambar 4.6, hasil pelatihan menunjukkan bahwa model MobileFaceNet yang dilatih menggunakan ArcFace mampu mencapai performa yang baik pada beberapa dataset evaluasi. Salah satu hasil evaluasi ditunjukkan pada dataset CFP-FP, di mana model memperoleh akurasi sebesar 97,54%. Dataset ini dikenal lebih menantang karena terdiri atas pasangan citra wajah dengan perbedaan pose frontal dan profile. Hasil tersebut menunjukkan bahwa MobileFaceNet yang dilatih menggunakan ArcFace tetap mampu mempertahankan performa pengenalan wajah yang tinggi meskipun terjadi perubahan sudut pandang wajah.

Sementara itu, pada dataset AgeDB-30, model memperoleh akurasi sebesar 97,42%. Nilai ini lebih baik dari model mobilefacenet biasa yang hanya mendapat performa Confusion Matrix, dengan hasil akurasi sebesar 91,38%.(Gilang Nugraha,2025)dengan ini menunjukkan bahwa model memiliki kemampuan yang baik dalam mengenali identitas wajah meskipun terdapat perbedaan usia pada citra yang dibandingkan.

Selain akurasi, log pelatihan juga menampilkan nilai XNorm yang berkisar antara 5,72 hingga 6,55. Nilai tersebut menunjukkan bahwa embedding yang dihasilkan memiliki distribusi yang stabil setelah proses normalisasi L2. Stabilitas embedding sangat penting karena akan digunakan pada tahap pencocokan identitas menggunakan metode cosine similarity pada sistem pengenalan wajah yang dikembangkan.

Secara keseluruhan, hasil pelatihan menunjukkan bahwa model MobileFaceNet yang dilatih menggunakan ArcFace berhasil menghasilkan representasi wajah yang kompak dan diskriminatif dengan tingkat akurasi yang tinggi.

**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

### 4.3.3 Evaluasi model

Evaluasi model dilakukan untuk mengukur kemampuan MobileFaceNet dalam menghasilkan representasi fitur wajah (embedding) yang dapat membedakan identitas setiap pengguna. Pengujian menggunakan Dataset yang digunakan dalam pengujian ini adalah LFW (Labeled Faces in the Wild), yaitu kumpulan citra wajah yang dihimpun dari internet dan banyak digunakan sebagai standar benchmark pada penelitian verifikasi wajah (face verification). Varian yang digunakan adalah LFW-funneled, yaitu versi LFW yang citranya telah melalui proses penyelarasan awal (pre-alignment) menggunakan metode funneling.

Selain citra wajah, digunakan pula berkas pairs.txt yang menjadi acuan skema pengujian. Berkas ini memuat daftar pasangan citra yang akan dibandingkan, yang terbagi menjadi dua kategori, yaitu:

1. Pasangan genuine, yaitu dua citra wajah yang berasal dari identitas (orang) yang sama
2. Pasangan impostor, yaitu dua citra wajah yang berasal dari dua identitas yang berbeda

Setiap pasangan genuine diberi label kelas 1, sedangkan setiap pasangan impostor diberi label kelas 0. Label inilah yang nantinya digunakan sebagai ground truth dalam penghitungan matriks evaluasi.

Pada evaluasi ini dilakukan pengujian antara 2 model mobilefacenet sebelum pretrain dan sesudah pretrain. Yang nantinya akan menghasilkan FAR, FRR, dan EER.

Equal Error Rate (EER) adalah titik pada kurva evaluasi di mana nilai FAR sama dengan nilai FRR. EER umum digunakan sebagai satu nilai tunggal untuk merepresentasikan performa sistem biometrik secara keseluruhan; semakin kecil nilai EER, semakin baik performa sistem.

Berikut hasil dari Evaluasi kedua model tersebut :

### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

**Tabel 4.1 Tabel Pengujian Penggunaan Sumber Daya**

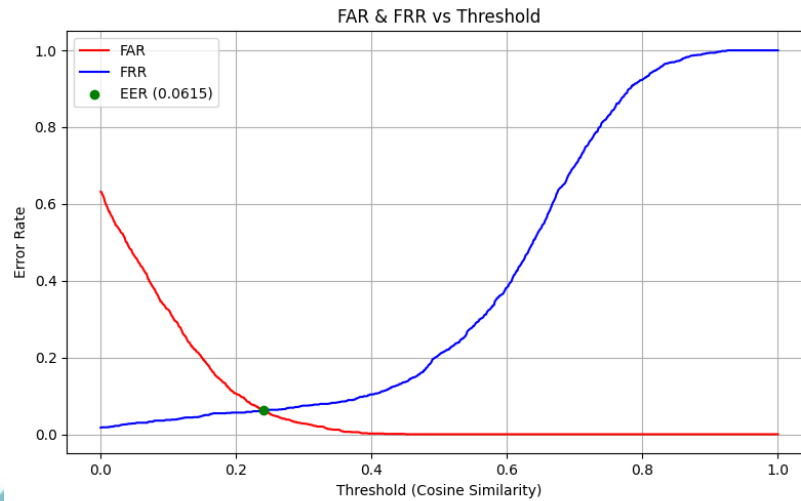
| Parameter                    | Pengujian 1        | Pengujian 2                      | Selisih             |
|------------------------------|--------------------|----------------------------------|---------------------|
| Equal Error Rate (EER)       | <b>0.0615</b>      | <b>0.0600</b>                    | ↓ 0.0015            |
| Threshold Saat EER           | 0.2403             | 0.2450                           | ↑ 0.0047            |
| Threshold Terbaik (Accuracy) | 0.359              | 0.403                            | ↑ 0.044             |
| Accuracy Maksimum            | 0.9541<br>(95.41%) | <b>0.9668</b><br><b>(96.68%)</b> | ↑ 0.0127<br>(1.27%) |
| FAR Minimum                  | 0.0000             | 0.0000                           | Tetap               |
| FRR Minimum                  | <b>0.0173</b>      | 0.0236                           | ↑ 0.0063            |

Berdasarkan Tabel 4.1, MobileFaceNet yang telah dipretrain menggunakan ArcFace Loss menunjukkan peningkatan performa dibandingkan MobileFaceNet tanpa pretraining. Peningkatan tersebut ditunjukkan oleh nilai accuracy maksimum yang meningkat dari 95,41% menjadi 96,68%, serta penurunan nilai Equal Error Rate (EER) dari 0,0615 menjadi 0,0600. Selain itu, nilai threshold terbaik berdasarkan accuracy juga meningkat dari 0,359 menjadi 0,403, sedangkan False Acceptance Rate (FAR) minimum tetap bernilai 0,0000 pada kedua model.

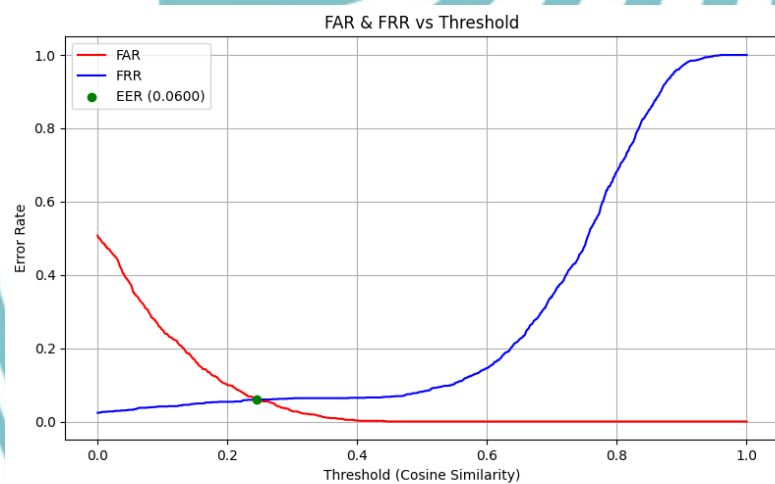
Untuk memberikan gambaran yang lebih jelas mengenai perubahan nilai FAR, FRR, dan EER, hasil evaluasi divisualisasikan dalam bentuk kurva sebagaimana ditunjukkan pada Gambar 4.7 dan Gambar 4.8.

### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



Gambar 4.7 Grafik FAR,FRR, dan ERR sebelum Pre-train



Gambar 4.8 Grafik FAR,FRR, dan ERR sesudah Pre-train

Berdasarkan Gambar 4.7 dan Gambar 4.8, kurva hasil evaluasi menunjukkan bahwa model yang telah dipretrain menggunakan ArcFace menghasilkan titik Equal Error Rate (EER) yang lebih rendah dibandingkan model tanpa pretraining. Penurunan nilai EER menunjukkan peningkatan kemampuan sistem dalam membedakan wajah yang berasal dari identitas yang sama maupun identitas yang berbeda.

**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Meskipun nilai False Rejection Rate (FRR) minimum meningkat sedikit dari 0,0173 menjadi 0,0236, kedua model sama-sama menghasilkan False Acceptance Rate (FAR) minimum sebesar 0,0000, sehingga tidak terjadi kesalahan penerimaan pengguna yang tidak terdaftar pada nilai threshold tertentu. Secara keseluruhan, hasil evaluasi menunjukkan bahwa penerapan ArcFace Loss pada proses pretraining mampu meningkatkan kualitas representasi fitur wajah (face embedding) sehingga memberikan performa verifikasi wajah yang lebih baik dibandingkan MobileFaceNet tanpa pretraining.

Setelah proses pelatihan selesai dilakukan dan model menunjukkan performa yang baik berdasarkan hasil evaluasi, tahap berikutnya adalah mengonversi model ke dalam format TensorFlow Lite (TFLite) agar dapat dijalankan secara efisien pada perangkat edge computing, seperti Raspberry Pi 4B. Format TFLite dipilih karena memiliki ukuran model yang lebih kecil serta mendukung proses inferensi dengan kebutuhan memori dan komputasi yang lebih rendah dibandingkan model TensorFlow standar.

Berikut cara konversi model menjadi TFlite :

**Kode Program 4.5 Konversi model ke TFlite**

```
tf.keras.mixed_precision.set_global_policy("float32")
f32_model = mobilefacenet_arcface()
f32_model.set_weights(model.get_weights())
outputModel = tf.keras.Model(f32_model.get_layer('input').input,
f32_model.get_layer('embedding').output, trainable=False)

outputModel.save('/kaggle/working/output_model')

converter = tf.lite.TFLiteConverter.from_keras_model(outputModel)
converter.optimizations = [tf.lite.Optimize.DEFAULT]
tflite_model = converter.convert()
with open('/kaggle/working/output_model.tflite', 'wb') as f:
    f.write(tflite_model)
```

**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Pada Kode Program 4.5, model yang telah selesai dilatih terlebih dahulu diubah menjadi model inference dengan menjadikan lapisan embedding sebagai keluaran model. Tahap ini dilakukan agar model TensorFlow Lite hanya menghasilkan embedding wajah, sedangkan lapisan ArcFace tidak lagi digunakan karena hanya berfungsi selama proses pelatihan.

Selanjutnya, proses konversi dilakukan menggunakan `TFLiteConverter.from_keras_model()`, kemudian diterapkan optimasi melalui `tf.lite.Optimize.DEFAULT` untuk menghasilkan model dengan ukuran yang lebih kecil dan lebih efisien. Hasil konversi berupa berkas `.tflite` yang selanjutnya digunakan pada proses enrollment dan recognition di perangkat Raspberry Pi 4B.

#### 4.4 Implementasi Incremental Enrollment

Incremental Enrollment merupakan mekanisme yang digunakan untuk menambahkan identitas pengguna baru ke dalam sistem pengenalan wajah tanpa memerlukan proses pelatihan ulang model MobileFaceNet. Pada sistem yang dikembangkan, proses enrollment dilakukan dengan cara mengekstraksi embedding wajah dari citra pengguna, kemudian menyimpan hasil embedding tersebut ke dalam basis data SQLite sebagai representasi identitas pengguna.

##### 4.4.1 Proses Akuisisi Data Wajah

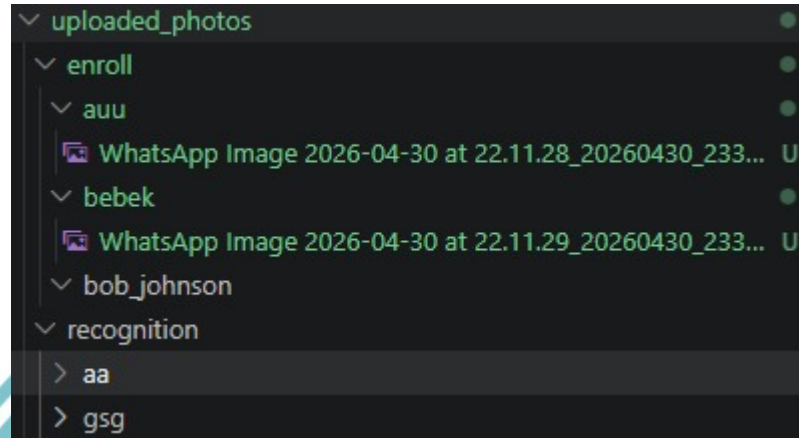
Proses akuisisi data wajah pada sistem yang dikembangkan dilakukan menggunakan pendekatan berbasis folder (folder-based acquisition). Pada metode ini, sistem tidak mengambil citra secara langsung dari kamera pada saat proses enrollment maupun recognition, melainkan membaca gambar yang telah tersedia pada direktori input sistem.

Struktur direktori yang digunakan terdiri atas dua folder utama, yaitu folder enroll yang digunakan untuk proses pendaftaran identitas baru dan folder recognition yang

### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

digunakan untuk proses pengenalan wajah. Kedua folder tersebut berada di dalam direktori `uploaded_photos` yang secara berkala dipantau oleh aplikasi utama.



Gambar 4.9 Struktur Folder Akuisisi Data Wajah

Berdasarkan Gambar 4.9, sistem menggunakan struktur folder sebagai media akuisisi data wajah. Seluruh citra yang akan diproses ditempatkan pada folder `uploaded_photos`, yang selanjutnya dipindai secara otomatis oleh modul utama saat aplikasi dijalankan.

Sistem kemudian memeriksa keberadaan gambar pada setiap subfolder untuk menentukan jenis proses yang akan dilakukan. Citra yang berada pada folder enrollment diproses sebagai data pendaftaran pengguna baru, sedangkan citra pada folder recognition digunakan untuk proses pengenalan wajah terhadap data embedding yang telah tersimpan di database.

Dengan mekanisme tersebut, proses akuisisi data dapat dilakukan secara otomatis berdasarkan lokasi penyimpanan citra tanpa memerlukan pemilihan mode secara manual oleh pengguna. Pendekatan ini juga mempermudah pengelolaan data selama proses enrollment maupun recognition.

**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

#### 4.4.2 Preprocessing Citra Wajah

Setelah citra wajah berhasil diperoleh dari proses akuisisi data, sistem melakukan tahap preprocessing untuk meningkatkan kualitas citra sebelum memasuki proses deteksi wajah dan ekstraksi embedding. Tahap ini sangat penting karena kualitas citra yang rendah dapat menyebabkan kegagalan deteksi wajah maupun menghasilkan embedding yang kurang representatif. Kondisi tersebut umum terjadi pada citra yang diperoleh dari ESP32-CAM karena keterbatasan sensor kamera, kondisi pencahayaan yang tidak merata, noise, serta rendahnya ketajaman gambar.

Pada penelitian ini, preprocessing dilakukan menggunakan pendekatan multi-stage image enhancement. Tahap pertama adalah normalisasi ukuran citra dan konversi format gambar agar sesuai dengan kebutuhan sistem. Selanjutnya dilakukan peningkatan kualitas citra menggunakan beberapa teknik pengolahan citra yang bertujuan memperbaiki detail wajah serta meningkatkan kualitas visual sebelum proses deteksi wajah dilakukan.

Proses preprocessing diawali dengan penerapan metode Super Resolution menggunakan FSRCNN (Fast Super-Resolution Convolutional Neural Network).

Metode ini digunakan untuk meningkatkan resolusi citra sehingga detail wajah yang sebelumnya kurang terlihat dapat diperjelas. Peningkatan resolusi ini sangat membantu ketika gambar yang diperoleh memiliki ukuran kecil atau mengalami penurunan kualitas akibat proses kompresi.

**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



Gambar 4.10 FSRCNN (Fast Super-Resolution Convolutional Neural Network)

Berdasarkan Gambar 4.10, penelitian ini menggunakan model FSRCNN dengan faktor pembesaran (scale factor)  $\times 2$ . Pemilihan skala  $\times 2$  dilakukan sebagai kompromi antara peningkatan kualitas citra dan kebutuhan komputasi pada perangkat edge. Dibandingkan faktor pembesaran yang lebih tinggi seperti  $\times 3$  atau  $\times 4$ , skala  $\times 2$  mampu meningkatkan ketajaman detail wajah tanpa menghasilkan artefak (distorsi hasil rekonstruksi) yang berlebihan serta tetap mempertahankan waktu pemrosesan yang lebih efisien. Dengan demikian, penggunaan FSRCNN  $\times 2$  dinilai lebih sesuai untuk mendukung proses ekstraksi fitur menggunakan MobileFaceNet pada Raspberry Pi.



Gambar 4.11 Sebelum Dan Sesudah FSRCNN

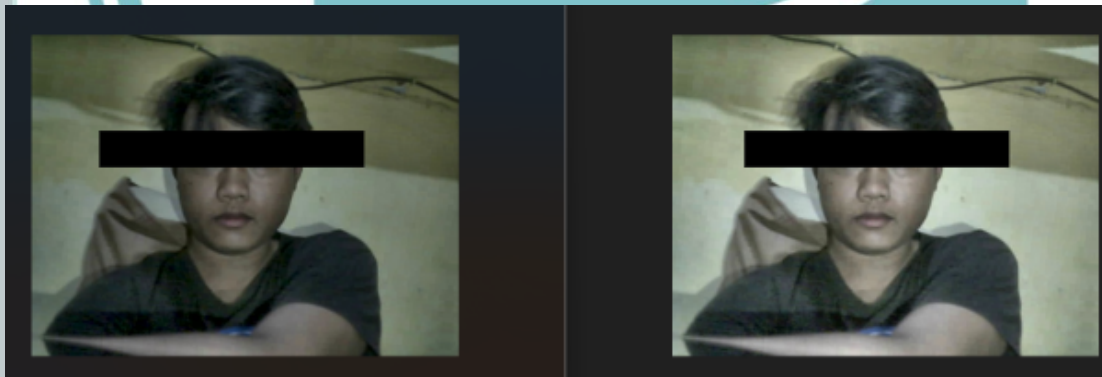
Berdasarkan Gambar 4.11, penerapan FSRCNN meningkatkan resolusi citra dari  $554 \times 540$  piksel menjadi  $1108 \times 1080$  piksel dengan faktor pembesaran  $\times 2$ . Peningkatan resolusi tersebut membuat detail wajah tampak lebih jelas dibandingkan citra asli, sehingga informasi visual yang digunakan pada proses ekstraksi fitur menjadi lebih baik.

**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Meskipun ukuran citra dan ukuran berkas meningkat setelah proses Super Resolution, metode FSRCNN  $\times 2$  tetap dipilih karena mampu meningkatkan kualitas visual tanpa menghasilkan artefak yang berlebihan. Hal ini penting untuk menjaga keseimbangan antara kualitas citra dan kebutuhan komputasi pada perangkat edge. Oleh karena itu, hasil keluaran FSRCNN selanjutnya digunakan sebagai masukan pada tahapan preprocessing berikutnya sebelum dilakukan proses ekstraksi embedding menggunakan MobileFaceNet.

Setelah itu dilakukan peningkatan kontras menggunakan metode Contrast Limited Adaptive Histogram Equalization (CLAHE). Teknik ini bekerja dengan memperbaiki distribusi intensitas cahaya pada area-area lokal gambar sehingga detail wajah tetap dapat terlihat meskipun kondisi pencahayaan tidak merata. Penggunaan CLAHE juga membantu meningkatkan visibilitas fitur-fitur penting seperti mata, hidung, dan kontur wajah.



Gambar 4.12 CLAHE Gelap dan Terang

Berdasarkan Gambar 4.12, penerapan CLAHE mampu meningkatkan kontras citra pada kondisi pencahayaan gelap maupun terang. Pada citra dengan pencahayaan rendah, detail wajah menjadi lebih jelas karena area yang sebelumnya gelap mengalami peningkatan intensitas secara adaptif. Sementara itu, pada citra dengan pencahayaan terang, CLAHE membantu mempertahankan detail wajah tanpa

### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

menghasilkan kontras yang berlebihan. Dengan demikian, fitur-fitur penting seperti mata, hidung, dan kontur wajah menjadi lebih mudah dikenali pada tahapan selanjutnya.

Tahap berikutnya adalah proses denoising untuk mengurangi noise atau gangguan visual yang muncul pada citra hasil tangkapan kamera. Noise yang berlebihan dapat memengaruhi proses deteksi wajah maupun ekstraksi fitur sehingga kualitas embedding yang dihasilkan menjadi kurang optimal.



Gambar 4.13 Denoising

Berdasarkan Gambar 4.13, proses denoising menghasilkan citra yang lebih bersih dengan mengurangi gangguan visual tanpa menghilangkan detail penting pada wajah. Proses ini membantu meningkatkan kualitas citra sehingga informasi yang digunakan pada tahap deteksi wajah dan ekstraksi fitur menjadi lebih stabil.

Setelah proses denoising selesai dilakukan, sistem menerapkan proses sharpening untuk meningkatkan ketajaman citra. Proses ini bertujuan memperjelas tepi objek dan detail wajah sehingga karakteristik wajah dapat direpresentasikan dengan lebih baik.

### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



Gambar 4.14 Sharpen

Berdasarkan Gambar 4.14, proses sharpening menghasilkan citra dengan detail tepi yang lebih jelas dibandingkan citra sebelum dilakukan penajaman. Peningkatan ketajaman tersebut membantu memperjelas kontur wajah dan bagian-bagian penting seperti mata, hidung, serta mulut sehingga kualitas citra masukan menjadi lebih baik sebelum dilakukan deteksi wajah.

Seluruh tahapan preprocessing tersebut dilakukan secara otomatis oleh sistem sebelum proses deteksi wajah dijalankan. Kombinasi Super Resolution, CLAHE, denoising, dan sharpening bertujuan meningkatkan kualitas citra sehingga proses deteksi wajah menggunakan Haar Cascade dapat berjalan dengan lebih optimal.

Pada tahap berikutnya, sistem menggunakan Haar Cascade untuk mendeteksi lokasi wajah pada citra. proses face alignment untuk menyelaraskan orientasi wajah

### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

sehingga posisi wajah menjadi seragam sebelum dilakukan ekstraksi fitur.



Gambar 4.15 Haar Cascade dan Alignment

Berdasarkan Gambar 4.15, Haar Cascade berhasil mendeteksi area wajah pada citra, kemudian sistem melakukan face alignment untuk menyesuaikan orientasi wajah berdasarkan posisi fitur-fitur utama, seperti mata. Proses alignment bertujuan mengurangi variasi akibat kemiringan kepala, rotasi wajah, maupun perbedaan sudut pengambilan gambar sehingga wajah yang dihasilkan memiliki posisi yang lebih konsisten. Dengan orientasi wajah yang seragam, proses ekstraksi embedding menggunakan MobileFaceNet dapat menghasilkan representasi fitur yang lebih stabil dan akurat.

Setelah wajah berhasil dideteksi dan disejajarkan, sistem menghasilkan beberapa variasi citra (multi-variant preprocessing) yang akan digunakan pada proses Incremental Enrollment. Variasi citra tersebut bertujuan meningkatkan keberagaman data wajah sehingga sistem menjadi lebih robust terhadap perubahan kualitas citra maupun kondisi lingkungan saat proses recognition berlangsung.

Variasi citra yang dihasilkan meliputi:

1. Original
2. CLAHE
3. Denoise

### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

4. Sharpen
5. Super Resolution (FSRCNN)
6. Kombinasi beberapa teknik enhancement



Gambar 4.16 Variasi citra

Berdasarkan Gambar 4.16, setiap citra wajah diproses menjadi beberapa variasi hasil preprocessing yang terdiri atas Original, CLAHE, Denoise, Sharpen, Super Resolution (FSRCNN), serta beberapa variasi yang merupakan kombinasi dari teknik-teknik enhancement tersebut. Masing-masing variasi memiliki karakteristik yang berbeda, seperti peningkatan kontras, pengurangan noise, peningkatan ketajaman, maupun peningkatan resolusi citra.

#### 4.4.3 Ekstraksi dan Validasi Embedding

Setelah proses preprocessing multi-variant selesai dilakukan, setiap citra wajah diproses menggunakan model MobileFaceNet untuk menghasilkan embedding wajah berdimensi 192. Embedding merupakan representasi numerik yang menyimpan karakteristik unik dari wajah pengguna dan digunakan sebagai dasar dalam proses enrollment maupun recognition. Pada penelitian ini, tidak semua embedding yang dihasilkan langsung disimpan ke database. Sistem terlebih dahulu melakukan proses validasi menggunakan fungsi `validate_embedding()` untuk memastikan bahwa embedding memiliki kualitas yang baik dan layak digunakan.

#### Kode Program 4.6 Implementasi Fungsi Validasi Embedding

```
def validate_embedding(self, embedding):

    # Check 1: Embedding tidak boleh kosong
```

**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```
if embedding is None or embedding.size == 0:
    return False, report

embedding = np.array(embedding).flatten()

# Check 2-3: Tidak mengandung NaN atau Inf
if np.isnan(embedding).any() or np.isinf(embedding).any():
    return False, report

# Check 4-5: Validasi L2 Norm
norm = np.linalg.norm(embedding)
if norm < 1e-6:
    return False, report

report['norm_healthy'] = (
    self.HEALTHY_EMBEDDING_NORM[0] <= norm <=
    self.HEALTHY_EMBEDDING_NORM[1]
)

# Check 6: Dimensi embedding
report['dimension_ok'] = (embedding.shape[0] == 192)

return report['valid'], report
```

Kode Program 4.6 yang ditampilkan pada bagian ini merupakan cuplikan implementasi fungsi `validate_embedding()` yang memuat tahapan utama proses validasi embedding. Implementasi lengkap beserta seluruh mekanisme pemeriksaan, pencatatan statistik, dan penanganan kesalahan disajikan pada Lampiran 6.

fungsi `validate_embedding()` menerima input berupa embedding hasil ekstraksi MobileFaceNet dan menghasilkan status validasi beserta laporan pemeriksaan. Tahap pertama yang dilakukan adalah memeriksa apakah embedding bernilai None atau kosong. Jika kondisi tersebut ditemukan, embedding langsung dinyatakan tidak valid dan proses validasi dihentikan.

Selanjutnya embedding dikonversi ke dalam bentuk `numpy.ndarray` dan diubah menjadi vektor satu dimensi menggunakan metode `flatten()`. Setelah itu dilakukan pemeriksaan terhadap nilai NaN (Not a Number) dan Inf (Infinity). Embedding yang

**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

mengandung salah satu nilai tersebut dianggap rusak sehingga tidak dapat digunakan pada proses pengenalan wajah.

Kemudian sistem menghitung nilai L2 Norm untuk mendeteksi kemungkinan terjadinya zero vector. Jika nilai norm lebih kecil dari batas minimum yang ditentukan, embedding dianggap tidak memiliki informasi fitur yang cukup dan langsung ditolak. Selain itu, nilai norm juga digunakan untuk mengevaluasi kesehatan embedding dengan rentang nilai 0,95 hingga 1,05 yang sesuai dengan karakteristik embedding hasil L2 Normalization.

Tahap berikutnya adalah pemeriksaan dimensi embedding. Sistem mengharapkan embedding memiliki panjang 192 dimensi sesuai dengan konfigurasi MobileFaceNet yang digunakan pada penelitian ini. Jika dimensi tidak sesuai, sistem memberikan peringatan sebagai indikasi kemungkinan kesalahan pada proses inferensi.

Hasil pemeriksaan kemudian disimpan dalam variabel report dan statistik validasi dicatat pada self.stats. Hanya embedding yang lolos validasi yang akan disimpan ke database SQLite dan digunakan pada proses recognition.

#### 4.4.4 Penyimpanan Embedding ke Database SQLite

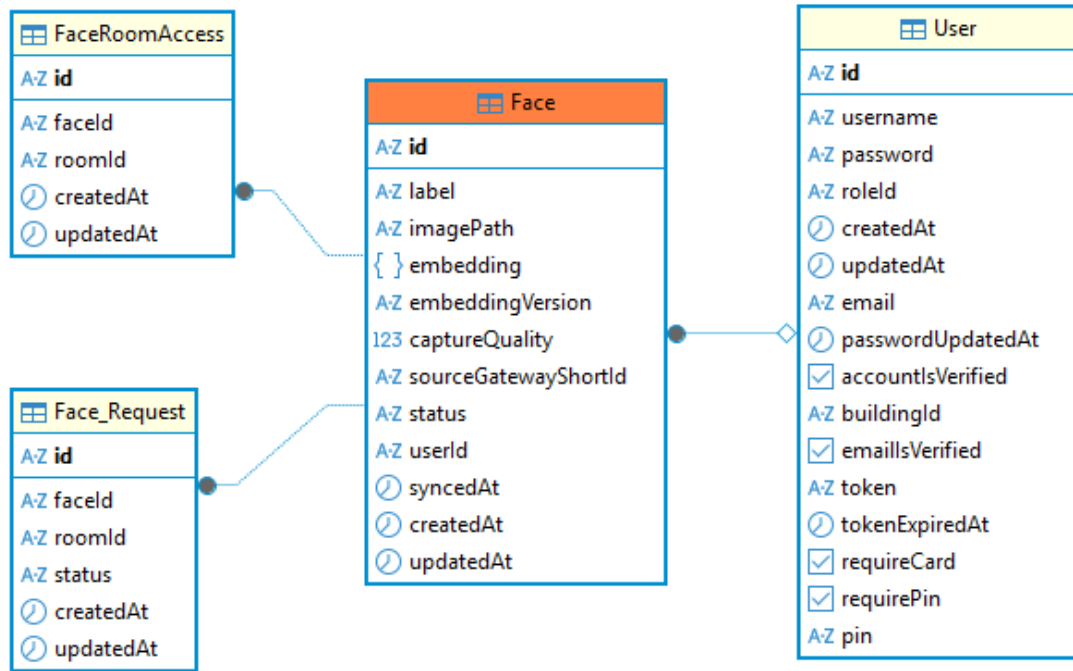
Embedding yang telah dinyatakan valid kemudian disimpan ke dalam database SQLite bersama informasi identitas pengguna. Setiap pengguna dapat memiliki beberapa embedding yang berasal dari variasi citra hasil proses Incremental Enrollment. Pendekatan ini memungkinkan sistem memiliki representasi wajah yang lebih beragam sehingga proses recognition menjadi lebih robust terhadap variasi pose maupun pencahayaan.

Untuk mendukung proses penyimpanan tersebut, penelitian ini menggunakan rancangan basis data sebagaimana ditunjukkan pada Gambar 4.17.

## © Hak Cipta milik Politeknik Negeri Jakarta

### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



Gambar 4.17 Entity Relationship Diagram (ERD) Penyimpanan Embedding ke Database SQLite

Berdasarkan Gambar 4.17, penyimpanan data hasil Incremental Enrollment menggunakan database SQLite yang terdiri atas tabel User, Face, FaceRoomAccess, dan Face\_Request. Tabel Face merupakan tabel utama yang digunakan untuk menyimpan data wajah hasil proses enrollment, sedangkan tabel lainnya berfungsi sebagai pendukung dalam pengelolaan pengguna, hak akses ruangan, dan pencatatan permintaan akses.

Hubungan antara tabel User dan Face ditunjukkan oleh simbol diamond kosong pada sisi tabel User, yang menunjukkan non-identifying relationship. Hal ini berarti tabel Face memiliki foreign key userId yang mengacu pada id pada tabel User, namun primary key pada tabel Face tetap menggunakan atribut id miliknya sendiri. Dengan demikian, satu pengguna dapat memiliki satu atau lebih data wajah (embedding), sedangkan setiap data wajah hanya dimiliki oleh satu pengguna.

### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Selain itu, tabel FaceRoomAccess digunakan untuk menyimpan hubungan antara data wajah dengan ruang (room) yang dapat diakses oleh pengguna, sedangkan tabel Face\_Request digunakan untuk menyimpan informasi permintaan akses beserta status prosesnya. Kedua tabel tersebut berelasi dengan tabel Face melalui atribut faceId sehingga setiap data akses dan permintaan dapat dikaitkan dengan identitas wajah yang telah terdaftar.

Pada penelitian ini, proses recognition hanya memanfaatkan data embedding yang tersimpan pada tabel Face sebagai dasar pencocokan identitas menggunakan metode cosine similarity, sedangkan tabel lainnya berfungsi sebagai pendukung pengelolaan data pengguna dan pencatatan aktivitas sistem.

| Column Name              | #  | Data type            | Identity | Collation | Not Null | Default                 | Comment |
|--------------------------|----|----------------------|----------|-----------|----------|-------------------------|---------|
| A:Z id                   | 1  | text                 |          | default   | [v]      |                         |         |
| A:Z label                | 2  | text                 |          | default   | [ ]      |                         |         |
| A:Z imagePath            | 3  | text                 |          | default   | [v]      |                         |         |
| { } embedding            | 4  | jsonb                |          |           | [ ]      |                         |         |
| A:Z embeddingVersion     | 5  | text                 |          | default   | [ ]      |                         |         |
| 123 captureQuality       | 6  | float8               |          |           | [ ]      |                         |         |
| A:Z sourceGatewayShortId | 7  | text                 |          | default   | [ ]      |                         |         |
| A:Z status               | 8  | public."FACE_STATUS" |          |           | [v]      | 'PENDING';"FACE_STATUS" |         |
| A:Z userId               | 9  | text                 |          | default   | [ ]      |                         |         |
| syncedAt                 | 10 | timestamp(3)         |          |           | [ ]      |                         |         |
| createdAt                | 11 | timestamp(3)         |          |           | [v]      | CURRENT_TIMESTAMP       |         |
| updatedAt                | 12 | timestamp(3)         |          |           | [v]      |                         |         |

Gambar 4.18 Tabel Penyimpanan Embedding ke Database SQLite

Struktur tabel Face yang digunakan untuk menyimpan hasil Incremental Enrollment ditunjukkan pada Gambar 4.18. Setiap data wajah yang telah lolos proses validasi akan disimpan sebagai satu record sehingga dapat langsung digunakan pada proses recognition tanpa memerlukan proses retraining model MobileFaceNet.

Tabel Face memiliki atribut id sebagai primary key yang digunakan untuk mengidentifikasi setiap data wajah secara unik. Informasi identitas wajah disimpan pada atribut label, sedangkan lokasi penyimpanan citra dicatat pada atribut imagePath. Hasil ekstraksi fitur menggunakan MobileFaceNet disimpan pada atribut embedding bertipe BLOB, sehingga representasi numerik wajah dapat disimpan secara efisien di dalam database SQLite.

#### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Selain itu, atribut `embeddingVersion` digunakan untuk menyimpan informasi versi `embedding`, `captureQuality` menyimpan nilai kualitas citra hasil preprocessing, sedangkan status menunjukkan status data wajah yang tersimpan. Atribut `userId` berfungsi sebagai foreign key yang menghubungkan data wajah dengan pengguna pada tabel `User` sebagaimana ditunjukkan pada Gambar 4.17. Sementara itu, atribut `createdAt`, `updatedAt`, dan `syncedAt` digunakan untuk mencatat waktu pembuatan, pembaruan, dan sinkronisasi data.

Melalui struktur tabel tersebut, sistem dapat menyimpan beberapa `embedding` untuk setiap pengguna sebagai hasil dari berbagai variasi citra pada proses `Incremental Enrollment`. Dengan mekanisme ini, pengguna baru dapat ditambahkan ke dalam sistem tanpa melakukan retraining model, dan `embedding` yang telah tersimpan dapat langsung digunakan pada proses `recognition`.

#### 4.4.5 Implementasi Recognition Embedding

Setelah proses `enrollment` selesai dan `embedding` pengguna tersimpan pada database `SQLite`, sistem dapat melakukan proses pengenalan wajah (`recognition`). Tujuan dari tahap ini adalah mengidentifikasi identitas pengguna berdasarkan kemiripan antara `embedding` wajah yang baru diekstraksi dengan `embedding` yang telah tersimpan pada database.

Proses pencocokan dilakukan menggunakan metode **Cosine Similarity**.

##### Kode Program 4.7 Kode Cosine Similarity

```
def _cosine_similarity(emb1: np.ndarray, emb2: np.ndarray) -> float:
    """Calculate cosine similarity"""
    emb1 = np.asarray(emb1, dtype=np.float32).flatten()
    emb2 = np.asarray(emb2, dtype=np.float32).flatten()

    norm1 = np.linalg.norm(emb1)
```

**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```
norm2 = np.linalg.norm(emb2)

if norm1 < 1e-6 or norm2 < 1e-6:
    return 0.0

return float(np.dot(emb1, emb2) / (norm1 * norm2))
```

Pada Kode Program 4.7, Sistem membandingkan embedding wajah query dengan seluruh embedding yang tersimpan pada database SQLite. Nilai similarity yang dihasilkan menunjukkan tingkat kemiripan antara dua embedding. Semakin tinggi nilai similarity, semakin besar kemungkinan kedua embedding berasal dari orang yang sama.

Pada implementasi ini digunakan mekanisme stepwise recognition, yaitu proses pencocokan wajah dilakukan secara bertahap terhadap setiap variasi citra hasil preprocessing. Proses diawali dengan pemeriksaan validitas citra masukan, kemudian sistem menghasilkan beberapa variasi citra menggunakan fungsi `generate_enrollment_frames()`.

Setiap variasi citra selanjutnya diproses melalui tahapan deteksi wajah, ekstraksi embedding menggunakan MobileFaceNet, serta validasi embedding untuk memastikan bahwa hasil ekstraksi memenuhi kriteria yang telah ditentukan. Embedding yang valid kemudian dibandingkan dengan seluruh embedding yang tersimpan pada database SQLite menggunakan metode cosine similarity.

Apabila pada salah satu variasi citra ditemukan nilai similarity yang melebihi nilai threshold yang telah ditentukan, proses recognition langsung dihentikan (early stopping) dan identitas pengguna dikembalikan sebagai hasil pengenalan. Sebaliknya, apabila seluruh variasi citra telah diproses namun tidak ditemukan nilai similarity yang memenuhi threshold, sistem mengembalikan status Unknown sebagai indikasi bahwa wajah tidak dikenali.

#### Kode Program 4.8 StepWise Recognition

```
def _recognize_stepwise(self, image, database):  
  
    if image is None or image.size == 0:  
        return UNKNOWN_FACE  
  
    variant_frames = self.pipeline.generate_enrollment_frames(image)  
  
    for frame, variant_name in variant_frames:  
  
        face_crop, _ = self.pipeline.detect_and_crop_face(frame)  
  
        embedding = self.embedder.extract_embedding(face_crop)  
  
        is_valid, _ = self.validator.validate_embedding(embedding)  
        if not is_valid:  
            continue  
  
        result = self.recognizer.match(embedding, database)  
  
        # Berhenti jika similarity melebihi threshold  
        if result.get("matched"):  
            return result  
  
    # Seluruh varian gagal dikenali  
    return UNKNOWN_FACE
```

Kode Program 4.8 yang ditampilkan pada bagian ini merupakan cuplikan implementasi mekanisme stepwise recognition yang memuat alur utama proses pengenalan wajah. Implementasi lengkap beserta seluruh proses penanganan hasil recognition, pelaporan kualitas citra, pencatatan logging, dan pengelolaan data disajikan pada Lampiran 7.

Seluruh hasil recognition kemudian dicatat ke dalam database PostgreSQL melalui modul logging sistem. Informasi yang disimpan meliputi identitas pengguna, nilai similarity, status akses, laporan kualitas citra, dan waktu pengenalan. Setelah proses selesai dilakukan, folder recognition beserta seluruh gambar di dalamnya akan dihapus secara otomatis untuk menjaga penggunaan ruang penyimpanan tetap efisien.

### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Dengan mekanisme tersebut, sistem mampu melakukan pengenalan wajah secara otomatis menggunakan embedding hasil enrollment tanpa memerlukan proses pelatihan ulang model. Pendekatan ini memungkinkan penambahan pengguna baru secara dinamis melalui Incremental Enrollment serta tetap mempertahankan performa recognition pada perangkat edge computing seperti Raspberry Pi 4. Seperti yang dapat dilihat dari contoh hasil preview output dari proses pengenalan wajah pada Gambar 4.19 dibawah :



Gambar 4.19 Hasil Preview output Recognition

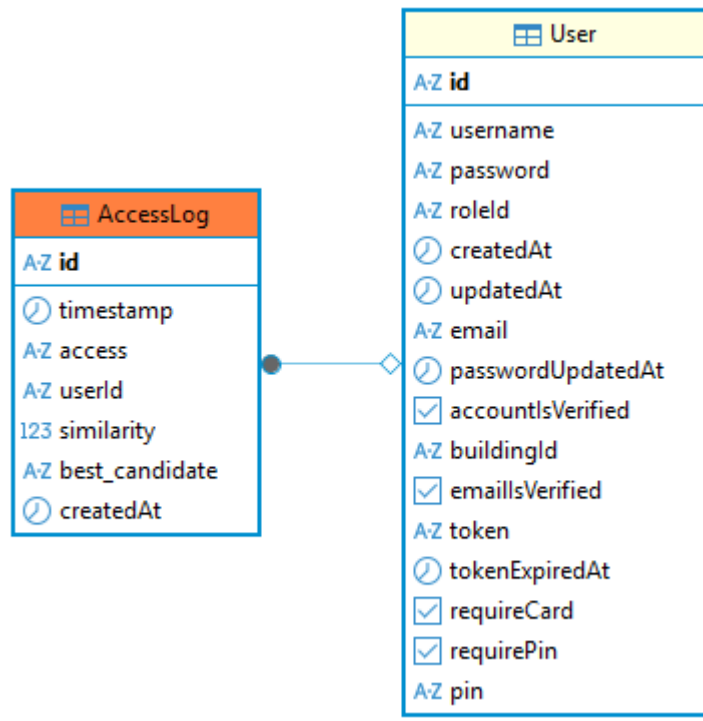
#### 4.4.6 Implementasi Pencatatan Log Sistem

Setelah proses pengenalan wajah selesai dilakukan, sistem akan mencatat hasil recognition ke dalam basis data PostgreSQL. Informasi yang disimpan meliputi identitas pengguna yang dikenali, nilai similarity, status hasil pencocokan (access), waktu proses pengenalan (timestamp), serta kandidat dengan nilai similarity tertinggi (best\_candidate). Pencatatan log ini bertujuan untuk menyediakan riwayat aktivitas sistem sehingga setiap proses autentikasi dapat ditelusuri kembali apabila diperlukan.

Struktur basis data yang digunakan untuk pencatatan log sistem ditunjukkan pada Gambar 4.20.

### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



Gambar 4.20 ERD Penyimpanan Log Sistem

Berdasarkan Gambar 4.20, tabel AccessLog merupakan tabel utama yang digunakan untuk menyimpan seluruh aktivitas hasil proses recognition. Tabel ini menyimpan informasi berupa timestamp sebagai waktu terjadinya proses autentikasi, access sebagai status hasil pengenalan, similarity sebagai nilai kemiripan antara wajah masukan dengan data yang tersimpan, serta best\_candidate sebagai kandidat dengan nilai similarity tertinggi apabila proses pengenalan tidak menghasilkan kecocokan yang memenuhi threshold. Selain itu, atribut userId digunakan sebagai foreign key yang menghubungkan data log dengan tabel User, sehingga setiap aktivitas autentikasi dapat dikaitkan dengan pengguna yang sesuai.

Sementara itu, tabel User menyimpan informasi akun pengguna yang digunakan dalam sistem autentikasi. Relasi antara tabel AccessLog dan User memungkinkan sistem menyimpan riwayat akses setiap pengguna secara terstruktur serta memudahkan proses pelacakan aktivitas autentikasi.

### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Melalui mekanisme tersebut, seluruh aktivitas pengenalan wajah dapat terdokumentasi secara otomatis tanpa mempengaruhi proses recognition. Data log yang tersimpan dapat dimanfaatkan untuk kebutuhan pemantauan sistem, analisis performa, maupun audit riwayat akses pengguna.

| Column Name         | # | Data type    | Identity | Collation | Not Null | Default | Comment |
|---------------------|---|--------------|----------|-----------|----------|---------|---------|
| AZ id               | 1 | text         |          | default   | [v]      |         |         |
| timestamp           | 2 | timestamp(3) |          |           | [v]      |         |         |
| AZ access           | 3 | varchar(20)  |          | default   | [v]      |         |         |
| AZ userId           | 4 | text         |          | default   | [ ]      |         |         |
| 123 similarity      | 5 | float8       |          |           | [ ]      |         |         |
| AZ best_candidate   | 6 | text         |          | default   | [ ]      |         |         |
| timestamp createdAt | 7 | timestamp(3) |          |           | [v]      | now()   |         |

Gambar 4.21 Penyimpanan Log Sistem

Struktur tabel AccessLog yang digunakan untuk menyimpan hasil proses recognition ditunjukkan pada Gambar 4.21. Setiap proses autentikasi yang dilakukan oleh sistem akan menghasilkan satu data log yang disimpan pada tabel ini.

Tabel AccessLog memiliki atribut id sebagai primary key yang digunakan untuk mengidentifikasi setiap data log secara unik. Atribut timestamp menyimpan waktu terjadinya proses autentikasi, sedangkan atribut access menyimpan status hasil pengenalan wajah, seperti granted, denied, atau status lain sesuai implementasi sistem.

Selain itu, atribut similarity menyimpan nilai cosine similarity antara wajah masukan dengan data embedding yang tersimpan di database. Atribut best\_candidate digunakan untuk menyimpan identitas pengguna dengan nilai similarity tertinggi apabila proses recognition tidak menghasilkan kecocokan yang memenuhi nilai threshold. Sementara itu, atribut userId berfungsi sebagai foreign key yang menghubungkan data log dengan tabel User, sebagaimana ditunjukkan pada Gambar 4.20. Adapun atribut createdAt digunakan untuk mencatat waktu penyimpanan data log ke dalam database PostgreSQL.

**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Melalui struktur tabel tersebut, seluruh aktivitas pengenalan wajah dapat terdokumentasi secara otomatis tanpa memengaruhi proses recognition. Data log yang tersimpan dapat dimanfaatkan untuk kebutuhan pemantauan sistem, analisis performa, maupun audit riwayat akses pengguna.

#### 4.5 Pengujian Sistem

Pengujian sistem dilakukan untuk mengevaluasi performa metode Incremental Enrollment yang diterapkan pada sistem pengenalan wajah berbasis edge computing. Pengujian bertujuan untuk mengetahui pengaruh penambahan identitas baru tanpa proses retraining model terhadap akurasi autentikasi, waktu inferensi, serta penggunaan sumber daya pada perangkat Raspberry Pi 4B.

Pengujian dilakukan menggunakan model MobileFaceNet yang telah dikonversi ke format TensorFlow Lite, database SQLite sebagai penyimpanan embedding, dan PostgreSQL sebagai media pencatatan aktivitas sistem. Seluruh pengujian dilakukan pada lingkungan implementasi yang sama untuk memastikan konsistensi hasil.

##### 4.5.1 Ruang Lingkup Pengujian

Ruang lingkup pengujian pada penelitian ini difokuskan pada evaluasi performa sistem pengenalan wajah setelah menerapkan metode Incremental Enrollment. Pengujian dilakukan terhadap proses recognition menggunakan embedding yang tersimpan pada database tanpa melakukan pelatihan ulang model.

Parameter yang diuji meliputi:

1. Akurasi autentikasi wajah.
2. Pengaruh Pre-Processing Terhadap Akurasi
3. Waktu inferensi sistem.
4. Penggunaan sumber daya perangkat.

### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

#### 5. Blackbox testing pengujian fungsional sistem facerecognition

Pengujian tidak mencakup implementasi aktuator fisik seperti relay, smart lock, maupun mekanisme kontrol akses perangkat keras lainnya. Fokus penelitian hanya pada performa sistem pengenalan wajah dan metode Incremental Enrollment yang diimplementasikan.

#### 4.5.2 Pengujian Akurasi Pengenalan Wajah

Pengujian akurasi dilakukan untuk mengetahui kemampuan sistem dalam mengenali identitas pengguna yang telah tersimpan pada database embedding. Disiapkan 100 citra gambar dengan 5 pengguna berbeda, Setiap citra uji diproses melalui tahapan preprocessing, ekstraksi embedding, validasi embedding, dan pencocokan menggunakan metode cosine similarity.

Hasil pengenalan kemudian dibandingkan dengan identitas sebenarnya untuk menentukan apakah recognition berhasil atau tidak. Nilai akurasi dihitung berdasarkan jumlah pengenalan yang benar dibandingkan dengan jumlah seluruh pengujian yang dilakukan.

$$Accuracy = \frac{Jumlah\ Pengenalan\ Benar}{Total\ Pengujian} \times 100\%$$

Pengujian ini digunakan untuk mengetahui apakah penambahan pengguna baru melalui Incremental Enrollment mempengaruhi kemampuan sistem dalam mengenali pengguna yang telah terdaftar sebelumnya.

**Tabel 4.2 Tabel Pengujian Akurasi Pengenalan Wajah**

| No | Nama Pengguna | Jumlah Data Uji | Dikenali | Tidak Dikenali | Rata-rata Cosine Similarity |
|----|---------------|-----------------|----------|----------------|-----------------------------|
|    |               |                 |          |                |                             |

## © Hak Cipta milik Politeknik Negeri Jakarta

### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

|              |                   |            |           |          |             |
|--------------|-------------------|------------|-----------|----------|-------------|
| 1            | Beryl             | 20         | 20        | 0        | 0,90        |
| 2            | Haikal            | 20         | 19        | 1        | 0,89        |
| 3            | Rafi              | 20         | 19        | 1        | 0,86        |
| 4            | Varo              | 20         | 20        | 0        | 0,98        |
| 5            | Wibi              | 20         | 20        | 0        | 0,98        |
| <b>Total</b> | <b>5 Pengguna</b> | <b>100</b> | <b>98</b> | <b>2</b> | <b>0,92</b> |

Berdasarkan Tabel 4.2, pengujian akurasi recognition dilakukan untuk mengevaluasi kemampuan sistem dalam mengenali identitas pengguna yang telah tersimpan pada basis data embedding. Data pengujian diperoleh dari hasil proses recognition yang tersimpan pada database PostgreSQL. Setiap pengguna diuji menggunakan 20 citra wajah dengan variasi kondisi yang berbeda untuk mensimulasikan penggunaan pada lingkungan nyata.

Komposisi data uji untuk setiap pengguna terdiri dari 10 citra wajah menghadap lurus ke kamera, 2 citra wajah menghadap ke kanan, 2 citra wajah menghadap ke kiri, 3 citra pada kondisi pencahayaan gelap, dan 3 citra pada kondisi pencahayaan terang. Dengan demikian, setiap pengguna memiliki total 20 data uji dan keseluruhan pengujian menggunakan 100 citra wajah dari 5 pengguna yang berbeda.

### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Berdasarkan hasil pengujian pada Tabel di atas, sistem mampu mengenali seluruh citra milik Beryl, Varo, dan Wibi dengan tingkat keberhasilan 100%. Sementara itu, pengguna Haikal dan Rafi masing-masing memiliki satu citra yang gagal dikenali sehingga tingkat keberhasilan recognition untuk kedua pengguna tersebut mencapai 95%. Nilai rata-rata cosine similarity yang diperoleh berada pada rentang 0,86 hingga 0,98, yang menunjukkan bahwa embedding hasil ekstraksi memiliki tingkat kemiripan yang tinggi terhadap data referensi yang tersimpan pada database.

| AZ user_id | AZ best_candidate | l23 similarity | 01 image_data            | AZ image_filename                              | created_at              |
|------------|-------------------|----------------|--------------------------|------------------------------------------------|-------------------------|
| varo8      | [NULL]            | 0.9158318043   | y0y0 JFIF y0 ICC_PROF... | WhatsApp Image 2026-05-11 at 16.57.49 (2).jpeg | 2026-05-24 16:57:21.030 |
| haikal8    | [NULL]            | 0.8632291555   | y0y0 JFIF y0 C ... [27K] | captured_130.jpg                               | 2026-05-24 17:11:55.496 |
| haikal8    | [NULL]            | 0.9082720876   | y0y0 JFIF y0 ICC_PROF... | WhatsApp Image 2026-05-11 at 16.36.01 (1).jpeg | 2026-05-24 17:11:55.686 |
| haikal8    | [NULL]            | 0.8632291555   | y0y0 JFIF y0 C ... [27K] | captured_130.jpg                               | 2026-05-24 17:12:18.248 |
| haikal8    | [NULL]            | 0.8632291555   | y0y0 JFIF y0 C ... [27K] | captured_130.jpg                               | 2026-05-24 17:16:20.372 |
| haikal8    | [NULL]            | 0.9082720876   | y0y0 JFIF y0 ICC_PROF... | WhatsApp Image 2026-05-11 at 16.36.01 (1).jpeg | 2026-05-24 17:16:20.540 |

Gambar 4.22 Database Recognition Log

Berdasarkan Gambar 4.22, setiap proses recognition menghasilkan informasi berupa identitas pengguna yang dikenali, nilai cosine similarity, status hasil pengenalan, waktu proses (timestamp), serta informasi pendukung lainnya yang secara otomatis dicatat ke dalam database PostgreSQL. Data log tersebut digunakan sebagai dasar dalam proses evaluasi performa sistem sekaligus sebagai dokumentasi aktivitas autentikasi pengguna.

Secara keseluruhan, dari 100 data uji yang dilakukan, sebanyak 98 citra berhasil dikenali dengan benar dan hanya 2 citra tidak dikenali, sehingga sistem memperoleh akurasi pengenalan sebesar 98%. Hasil ini menunjukkan bahwa metode Incremental Enrollment mampu menambahkan data pengguna baru tanpa menurunkan kemampuan sistem dalam mengenali pengguna yang telah terdaftar sebelumnya.

#### 4.3.2 Pengujian Pengaruh Pre-Prosesing Terhadap Akurasi

Pada pengujian ini dilakukan di peruntukan untuk mengukur efektivitas dari pre-processing, Analisis dilakukan secara bertahap dengan membandingkan hasil pengenalan wajah menggunakan beberapa skenario preprocessing.

### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Seluruh pengujian menggunakan dataset yang sudah disiapkan sebelumnya yang digunakan untuk pengujian akurasi pengenalan wajah yang mana di miliki total 100 citra gambar dari 5 pengguna berbeda. Lalu menghitung rata-rata similarity akurasi dalam pengenalan wajah tiap skenario preprocessing.

Berikut hasil dari pengujian nya:

| SUMMARY TABLE                 |             |
|-------------------------------|-------------|
| Scenario                      | Accuracy(%) |
| Alignment                     | 75.0000     |
| Alignment + SR                | 87.5000     |
| Alignment + CLAHE Bright      | 81.2500     |
| Alignment + CLAHE Dark        | 81.2500     |
| Alignment + SR + CLAHE Bright | 93.7500     |
| Alignment + SR + CLAHE Dark   | 93.7500     |
| Alignment + Denoise           | 75.0000     |
| Alignment + Denoise + SR      | 75.0000     |
| Alignment + Sharpen           | 75.0000     |
| Alignment + Sharpen + SR      | 81.2500     |

Gambar 4.23 Pengujian Pengaruh Pre-Processing Terhadap Akurasi

Berdasarkan Gambar 4.23, skenario Alignmenent sebagai baseline menghasilkan akurasi sebesar 75,00%. Penambahan Super Resolution meningkatkan akurasi menjadi 87,50%, sedangkan penerapan CLAHE Bright maupun CLAHE Dark menghasilkan akurasi 81,25%. Kombinasi Super Resolution dengan CLAHE Bright maupun CLAHE Dark memberikan performa terbaik dengan akurasi sebesar 93,75%, sehingga menunjukkan bahwa peningkatan resolusi yang disertai perbaikan kontras mampu meningkatkan kualitas citra wajah untuk proses pengenalan.

Sebaliknya, penggunaan Denoising maupun Sharpening tidak memberikan peningkatan akurasi yang signifikan. Skenario Alignment + Denoising, Alignment + Denoising + Super Resolution, dan Alignment + Sharpening tetap menghasilkan akurasi 75,00%, sedangkan Alignment + Sharpening + Super Resolution hanya mencapai 81,25%. Berdasarkan hasil tersebut, dapat disimpulkan bahwa kombinasi

**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Alignment + Super Resolution + CLAHE merupakan preprocessing yang paling efektif dalam meningkatkan performa.

#### 4.5.3 Pengujian Waktu Inferensi

Pengujian waktu inferensi dilakukan untuk mengukur latensi proses pengenalan wajah pada sistem autentikasi berbasis MobileFaceNet yang dijalankan pada perangkat edge computing. Pengukuran dilakukan dengan mencatat waktu eksekusi setiap tahapan dalam pipeline enrollment, mulai dari citra masukan hingga sistem menghasilkan keputusan identifikasi pengguna.

Tahapan yang diukur meliputi:

1. Preprocessing citra wajah.
2. Deteksi wajah menggunakan Haar Cascade.
3. Ekstraksi fitur (embedding) menggunakan MobileFaceNet.
4. Perhitungan cosine similarity terhadap data wajah yang tersimpan.
5. Pengambilan keputusan identifikasi.

Pengujian dilakukan menggunakan 100 sampel citra wajah. Waktu eksekusi setiap tahapan dicatat kemudian dihitung nilai rata-ratanya untuk memperoleh waktu inferensi sistem.

##### Kode Program 4.9 Pencatatan Waktu Inferensi Enrollment

```
def _format_preprocess_detail(preprocess_timings, preprocess_total=0.0):  
    stage_names = [  
        'alignment',  
        'super_resolution',  
        'clahe_brighten',  
        'clahe_darken',  
        'sr_clahe_bright',  
        'sr_clahe_dark',  
        'denoise_aligned',  
        'denoise_sr',
```

**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```
'sharpen_aligned',  
'sharpen_sr',  
]  
  
if isinstance(preprocess_timings, dict):  
    values = {}  
    for key in stage_names:  
        values[key] = float(preprocess_timings.get(key, 0.0) or 0.0)  
    if any(v > 0.0 for v in values.values()):  
        return ' | '.join(f'{key}: {values[key]*1000:.1f} ms' for key in stage_names)  
  
if preprocess_total > 0.0:  
    return f'total: {preprocess_total*1000:.1f} ms'  
return 'no_stage_timing_available'
```

Berikut merupakan hasil pengujian dari Pengukuran dilakukan dengan mencatat waktu eksekusi setiap tahapan dalam pipeline enrollment, mulai dari citra masukan hingga sistem menghasilkan keputusan identifikasi pengguna :

**POLITEKNIK  
NEGERI  
JAKARTA**

### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```
=====
Samples processed: 100
Avg preprocess: 1625.7 ms
Avg embedding : 174.9 ms
Per-process avg:
- alignment           72.2 ms
- super_resolution    1364.1 ms
- clahe_brighten       6.0 ms
- clahe_darken         8.7 ms
- sr_clahe_bright      23.7 ms
- sr_clahe_dark        45.8 ms
- denoise_aligned      16.2 ms
- denoise_sr           62.7 ms
- sharpen_aligned       4.2 ms
- sharpen_sr           22.1 ms
=====
```

Gambar 4.24 Pengujian Waktu Inferensi Enrollment

```
=====
RECOGNITION TIMING SUMMARY
=====
Samples measured : 100
Avg preprocess : 1593.8 ms
Avg detection : 22.3 ms
Avg embedding : 16.6 ms
Avg similarity : 8.1 ms
Avg total : 1645.3 ms
Per-process avg:
- alignment           71.6 ms
- super_resolution    1333.9 ms
- clahe_brighten       5.7 ms
- clahe_darken         8.6 ms
- sr_clahe_bright      23.5 ms
- sr_clahe_dark        45.2 ms
- denoise_aligned      16.3 ms
- denoise_sr           62.7 ms
- sharpen_aligned       4.2 ms
- sharpen_sr           22.0 ms
=====
```

Gambar 4.25 Pengujian Waktu Inferensi Recognition

Berdasarkan hasil pengujian yang ditunjukkan pada Gambar 4.24, proses enrollment memerlukan rata-rata waktu 1800,6 ms untuk memproses satu citra wajah. Waktu tersebut terdiri atas proses preprocessing sebesar 1625,7 ms dan ekstraksi embedding menggunakan MobileFaceNet sebesar 174,9 ms. Sebagai perbandingan, ketika tahap Super Resolution tidak digunakan, total waktu enrollment hanya sekitar 436,5 ms. Hal ini menunjukkan bahwa penerapan Super Resolution meningkatkan waktu

**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

komputasi secara signifikan, namun bertujuan menghasilkan citra dengan kualitas yang lebih baik sehingga proses ekstraksi fitur menjadi lebih optimal.

Sementara itu, berdasarkan hasil pengujian pada Gambar 4.25, proses recognition memerlukan rata-rata waktu 1645,3 ms atau sekitar 1,65 detik untuk mengenali satu citra wajah. Waktu tersebut terdiri atas preprocessing sebesar 1593,8 ms, deteksi wajah sebesar 22,3 ms, ekstraksi embedding menggunakan MobileFaceNet sebesar 16,6 ms, serta perhitungan cosine similarity sebesar 8,1 ms. Apabila Super Resolution tidak diterapkan, waktu recognition berkurang secara signifikan menjadi sekitar 311,2 ms, sehingga proses identifikasi dapat dilakukan dalam waktu kurang dari setengah detik.

Berdasarkan data pada Gambar 4.24 dan Gambar 4.25, dapat diketahui bahwa tahapan preprocessing merupakan komponen utama yang menyebabkan tingginya latensi sistem. Pada proses enrollment, preprocessing menyumbang sekitar 90,3% dari total waktu komputasi, sedangkan pada proses recognition kontribusinya mencapai sekitar 96,9%. Sebaliknya, proses inti pengenalan wajah memiliki latensi yang relatif rendah. MobileFaceNet hanya membutuhkan waktu rata-rata 174,9 ms untuk menghasilkan embedding pada tahap enrollment dan 16,6 ms pada tahap recognition. Selain itu, proses deteksi wajah dan perhitungan cosine similarity masing-masing hanya memerlukan 22,3 ms dan 8,1 ms, sehingga kontribusinya terhadap total latensi sistem relatif kecil.

Analisis lebih lanjut terhadap rincian waktu pada Gambar 4.24 dan Gambar 4.25 menunjukkan bahwa proses Super Resolution merupakan tahapan yang paling dominan terhadap waktu komputasi. Rata-rata waktu yang dibutuhkan mencapai 1364,1 ms pada proses enrollment dan 1333,9 ms pada proses recognition. Sebaliknya, tahapan preprocessing lainnya, seperti face alignment, CLAHE Bright, CLAHE Dark, dan sharpening, hanya memerlukan waktu beberapa milidetik sehingga kontribusinya terhadap total latensi relatif kecil. Oleh karena itu, dapat

**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

disimpulkan bahwa peningkatan waktu pemrosesan sistem terutama disebabkan oleh penggunaan Super Resolution. Dengan demikian, penggunaan Super Resolution dapat disesuaikan dengan kebutuhan implementasi. Apabila sistem memprioritaskan akurasi pengenalan, maka Super Resolution dapat diaktifkan meskipun meningkatkan latensi. Sebaliknya, apabila sistem lebih mengutamakan kecepatan respons, tahapan tersebut dapat dinonaktifkan sehingga proses enrollment dan recognition berlangsung lebih cepat dengan konsekuensi penurunan akurasi.

#### 4.5.4 Pengujian Penggunaan Sumber Daya

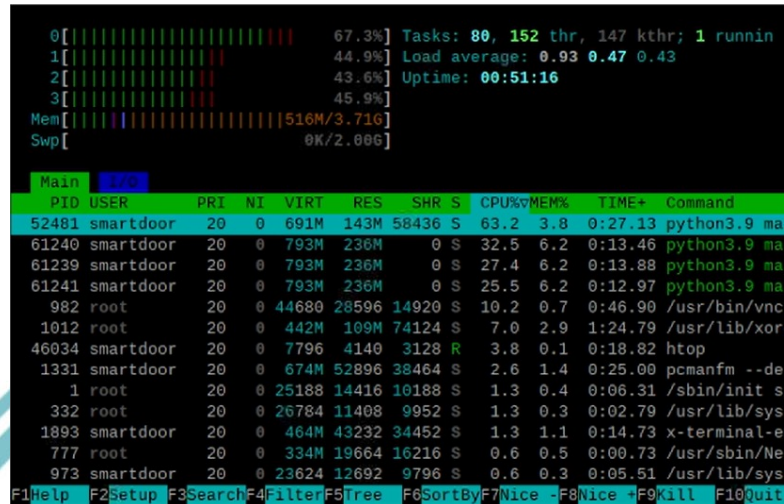
Pengujian penggunaan sumber daya dilakukan untuk mengetahui tingkat konsumsi CPU dan memori selama proses enrollment maupun recognition berlangsung. Pengukuran dilakukan pada perangkat Raspberry Pi 4B saat sistem menjalankan seluruh pipeline pengenalan wajah. Data yang diperoleh digunakan untuk mengevaluasi efisiensi implementasi MobileFaceNet dan metode Incremental Enrollment pada perangkat dengan sumber daya terbatas. Parameter yang diamati meliputi:

1. Persentase penggunaan CPU.
2. Penggunaan memori (RAM).
3. Stabilitas sistem selama proses recognition.

Pengukuran dilakukan pada tiga kondisi pengujian, yaitu kondisi idle (sistem aktif tanpa menjalankan proses pengenalan wajah), proses enrollment, dan proses recognition. Monitoring penggunaan sumber daya dilakukan menggunakan utilitas htop pada sistem operasi Raspberry Pi OS. Tampilan hasil monitoring untuk masing-masing kondisi disajikan pada Lampiran 7 (Idle), Lampiran 8 (Enrollment), dan Lampiran 9 (Recognition).

### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



**Gambar 4.26 Penggunaan Sumber Daya (Idle,Enrollment,Recognition)**

Berdasarkan hasil monitoring pada Lampiran 7, Lampiran 8, dan Lampiran 9, terlihat bahwa penggunaan CPU dan memori mengalami peningkatan ketika sistem melakukan proses enrollment dan recognition dibandingkan dengan kondisi idle. Peningkatan tersebut menunjukkan bahwa proses pengolahan citra dan ekstraksi embedding merupakan komponen yang paling banyak memanfaatkan sumber daya komputasi pada Raspberry Pi 4B.

**Tabel 4.3 Tabel Pengujian Penggunaan Sumber Daya**

| Parameter                  | Idle   | Enrollment | Recognition |
|----------------------------|--------|------------|-------------|
| Penggunaan CPU Tertinggi   | 3,2%   | 94,8%      | 67,3%       |
| Penggunaan RAM             | 454 MB | 595 MB     | 516 MB      |
| Persentase RAM             | 12,2%  | 16,0%      | 13,9%       |
| Tambahan RAM terhadap Idle | -      | 141 MB     | 62 MB       |
| Penggunaan Swap            | 0 KB   | 0 KB       | 0 KB        |
| Load Average               | 0,26   | 0,28       | 0,93        |
| Jumlah Proses Python Aktif | 0      | 4          | 4           |

Berdasarkan Tabel 4.3, kondisi idle menunjukkan penggunaan memori sebesar 454 MB atau sekitar 12,2% dari total kapasitas RAM Raspberry Pi 4B sebesar 3,71 GB. Ketika proses enrollment dijalankan, penggunaan memori meningkat menjadi 595 MB atau sekitar 16,0% dari total RAM. Sementara itu, pada proses recognition, penggunaan memori tercatat sebesar 516 MB atau sekitar 13,9% dari total kapasitas RAM.

**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

Dari sisi penggunaan CPU, kondisi idle menunjukkan utilisasi prosesor yang rendah dengan nilai tertinggi sebesar 3,2%. Pada proses enrollment, penggunaan CPU meningkat hingga mencapai 94,8%, menunjukkan bahwa proses pembentukan dan penyimpanan embedding membutuhkan sumber daya komputasi yang cukup tinggi. Pada proses recognition, penggunaan CPU mencapai 67,3%, yang masih lebih rendah dibandingkan proses enrollment karena sistem hanya melakukan deteksi wajah, ekstraksi embedding, dan pencocokan dengan data yang tersimpan.

Stabilitas sistem dievaluasi berdasarkan nilai load average dan penggunaan swap memory. Hasil pengujian menunjukkan bahwa nilai load average tertinggi selama proses recognition adalah 0,93, yang masih jauh di bawah jumlah inti prosesor Raspberry Pi 4B yang berjumlah empat inti. Selain itu, tidak ditemukan penggunaan swap memory pada seluruh kondisi pengujian. Hasil tersebut menunjukkan bahwa sistem mampu berjalan secara stabil tanpa mengalami kelebihan beban maupun kekurangan memori selama proses pengenalan wajah berlangsung.

Berdasarkan hasil pengujian penggunaan sumber daya, sistem pengenalan wajah yang diimplementasikan pada Raspberry Pi 4B menunjukkan efisiensi yang baik untuk lingkungan edge computing. Penggunaan memori selama proses enrollment dan recognition hanya berada pada rentang 13,9% hingga 16,0% dari total kapasitas RAM yang tersedia. Meskipun penggunaan CPU meningkat cukup signifikan selama proses pengolahan wajah, sistem tetap dapat beroperasi secara stabil tanpa penggunaan swap memory. Nilai load average yang rendah juga menunjukkan bahwa perangkat masih memiliki sumber daya yang cukup untuk menjalankan seluruh proses pengenalan wajah secara lokal. Dengan demikian, implementasi MobileFaceNet dan Incremental Enrollment dinilai sesuai untuk diterapkan pada perangkat edge computing dengan keterbatasan sumber daya komputasi.

### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

#### 4.5.6 Pengujian Fungsional Sistem (Black Box Testing)

Pengujian Black Box dilakukan untuk memverifikasi bahwa setiap fungsi utama pada sistem pengenalan wajah dapat berjalan sesuai dengan kebutuhan yang telah dirancang. Pengujian dilakukan dengan memberikan masukan (input) pada sistem dan mengamati keluaran (output) yang dihasilkan tanpa memperhatikan proses internal program.

Fungsi yang diuji meliputi proses enrollment, recognition, pengelolaan data pengguna, dan penyimpanan data embedding ke dalam database.

**Tabel 4.4 Pengujian Fungsional Sistem (Black Box Testing)**

| No | Skenario Pengujian                             | Input                                   | Hasil yang Diharapkan                                            | Hasil Pengujian                    | Status |
|----|------------------------------------------------|-----------------------------------------|------------------------------------------------------------------|------------------------------------|--------|
| 1  | Penambahan pengguna baru ( <i>Enrollment</i> ) | Wajah pengguna baru terdeteksi kamera   | Sistem berhasil melakukan enrollment dan menyimpan data pengguna | Data pengguna berhasil ditambahkan | Valid  |
| 2  | Enrollment dengan wajah tidak terdeteksi       | Kamera tidak menemukan wajah pada citra | Sistem menolak proses enrollment dan menampilkan pesan kesalahan | Enrollment ditolak                 | Valid  |

© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

|   |                                      |                                                   |                                                                   |                                      |       |
|---|--------------------------------------|---------------------------------------------------|-------------------------------------------------------------------|--------------------------------------|-------|
| 3 | Enrollment dengan wajah blur         | Citra wajah memiliki kualitas rendah atau blur    | Sistem menolak proses enrollment                                  | Enrollment ditolak                   | Valid |
| 4 | Recognition pengguna terdaftar       | Wajah pengguna yang telah tersimpan pada database | Sistem berhasil mengenali identitas pengguna                      | Identitas pengguna berhasil dikenali | Valid |
| 5 | Recognition pengguna tidak terdaftar | Wajah tidak terdapat pada database embedding      | Sistem menampilkan status <i>Unknown</i>                          | Status <i>Unknown</i> ditampilkan    | Valid |
| 6 | Recognition dengan wajah blur        | Citra wajah berkualitas rendah                    | Sistem tidak melakukan recognition atau menolak proses pengenalan | Recognition ditolak                  | Valid |
| 7 | Recognition tanpa wajah              | Tidak terdapat wajah pada frame kamera            | Sistem tidak melakukan proses recognition                         | Tidak ada proses recognition         | Valid |

### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

|   |                                          |                                                    |                                              |                              |       |
|---|------------------------------------------|----------------------------------------------------|----------------------------------------------|------------------------------|-------|
| 8 | Penghapusan data pengguna                | Memilih data pengguna yang tersimpan pada database | Data pengguna dan embedding berhasil dihapus | Data berhasil dihapus        | Valid |
| 9 | Penyimpanan vektor embedding ke database | Proses enrollment berhasil dilakukan               | Embedding tersimpan pada database SQLite     | Embedding berhasil tersimpan | Valid |

Berdasarkan hasil pengujian pada Tabel 4.4 Black Box testing, seluruh fungsi utama sistem berhasil dijalankan sesuai dengan kebutuhan yang telah ditetapkan. Proses enrollment mampu menambahkan pengguna baru serta menolak citra yang tidak memenuhi syarat, seperti wajah yang tidak terdeteksi atau citra yang blur. Pada proses recognition, sistem dapat mengenali pengguna yang telah terdaftar, menampilkan status Unknown untuk pengguna yang tidak terdaftar, serta tidak memproses citra yang tidak mengandung wajah atau memiliki kualitas yang rendah.

Selain itu, fungsi penghapusan data pengguna dan penyimpanan vektor embedding ke dalam database juga berjalan dengan baik. Hal ini menunjukkan bahwa mekanisme pengelolaan data pengguna dan database embedding telah berfungsi sesuai dengan rancangan sistem.

Berdasarkan hasil Black Box Testing, seluruh fungsi yang diuji berhasil berjalan sesuai dengan spesifikasi kebutuhan sistem. Seluruh skenario pengujian memperoleh hasil yang sesuai dengan keluaran yang diharapkan sehingga tingkat keberhasilan pengujian mencapai 100%. Dengan demikian, sistem pengenalan wajah berbasis



## © Hak Cipta milik Politeknik Negeri Jakarta

MobileFaceNet dan Incremental Enrollment telah memenuhi kebutuhan fungsional dan siap digunakan pada lingkungan implementasi yang telah dirancang.



### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta