

**LAPORAN TEKNIS : PENGEMBANGAN SISTEM PENGENALAN WAJAH DAN DETEKSI  
SUKU MENGGUNAKAN COMPUTER VISION**

*Proposal ini dibuat untuk memenuhi tugas mata kuliah Pengolahan Citra Digital*

Dosen : Rizky



Dibuat Oleh :

- Moch Rizky Taufiqurrahman : 231511018
- Prima Aji Akbar : 231511025
- Rafka Imanda Putra : 231511026

**JURUSAN TEKNIK KOMPUTER DAN INFORMATIKA**

**PROGRAM STUDI D3 TEKNIK INFORMATIKA**

**POLITEKNIK NEGERI BANDUNG**

**2025**

## DAFTAR ISI

<b>DAFTAR ISI.....</b>	<b>i</b>
<b>BAB 1 PENDAHULUAN .....</b>	<b>1</b>
1.1 Latar Belakang .....	1
1.2 Rumusan Masalah .....	1
1.3 Tujuan .....	1
1.4 Manfaat Penelitian .....	1
<b>BAB II TINJAUAN PUSTAKA .....</b>	<b>2</b>
2.1 Teknologi Pengenalan Wajah .....	2
2.2 Deteksi dan Klasifikasi Etnis .....	2
2.2.1 Pendekatan Antropometrik .....	2
2.2.2 Metode Machine Learning .....	2
2.3 Dataset Wajah dan Keterwakilan .....	2
2.4 Algoritma Deteksi Wajah .....	2
2.4.1 Haar Cascade Classifier .....	3
2.4.2 Multi-task Cascaded Convolutional Networks (MTCNN) .....	3
2.4.3 RetinaFace.....	3
<b>BAB III METODOLOGI .....</b>	<b>4</b>
3.1 Perancangan dan Pengumpulan Dataset .....	4
3.1.1 Kriteria Pengumpulan Data.....	4
3.1.2 Prosedur Pengumpulan .....	4
3.2 Pipeline Preprocessing .....	4
3.2.1 Deteksi Wajah.....	4
3.2.2 Normalisasi Wajah.....	5
3.2.3 Augmentasi Wajah.....	5
3.3 Implementasi Deteksi Wajah .....	6
3.3.1 MTCNN .....	6
3.4 Ekstraksi Fitur Wajah .....	6
3.4.1 Preprocessing untuk Ekstraksi Fitur.....	6
3.4.2 Face Embedding.....	6
3.4.3 Evaluasi Kualitas Embedding .....	6
3.5 Klasifikasi Teknis .....	6
<b>BAB IV IMPLEMENTASI .....</b>	<b>8</b>
4.1 Alat dan Library .....	8
4.1.1 Bahasa Pemrograman dan Framework.....	8

4.1.2 Library Computer Vision .....	8
4.1.3 Library Machine Learning .....	8
4.1.3 Tools Pengembangan .....	8
4.2 Arsitektur Sistem .....	8
4.2.1 Input Processing Module .....	9
4.2.2 Face Detection Module .....	9
4.2.3 Feature Extraction Module.....	9
4.2.4 Face Similarity Module.....	9
4.2.5 Ethnicity Classification .....	9
4.2.6 Visualization Module .....	9
4.2.7 Sytem Integration.....	9
4.3 Hypeparametes.....	9
4.3.1 Deteksi Wajah (MTCNN).....	9
4.3.2 Face Similarity (FaceNet) .....	9
4.3.3 Deteksi Etnis (ResNet18).....	10
4.3.4 Preprocessing Dataset .....	10
<b>BAB V HASIL EKSPERIMEN DAN ANALISIS.....</b>	<b>11</b>
5.1 Statistik Dataset .....	11
5.2 Performa Preprocessing .....	11
5.2.1 Deteksi Wajah.....	11
5.2.2 Distribusi Dataset.....	12
5.3 Performa Face Similarity .....	14
5.3.1 Evaluasi Face Embedding .....	14
5.3.2 Analisis ROC Curve.....	14
5.4 Performa Deteksi Etnis .....	16
5.4.1 Metrik Pelatihan Model .....	16
5.4.2 Performa Test Set.....	18
5.5 Analisis Sistem Integrasi.....	20
5.5.1 Waktu Pemrosesan.....	21
5.5.2 Integrasi Antarmuka.....	21
5.6 Limitasi dan Tantangan.....	21
<b>BAB VI KESIMPULAN DAN PENGEMBANGAN MASA DEPAN.....</b>	<b>22</b>
6.1 Kesimpulan .....	22
6.2 Limitasi .....	22
6.3 Saran Pengembangan .....	22

<b>REFERENSI.....</b>	<b>24</b>
-----------------------	-----------

# **BAB 1**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Teknologi pengenalan wajah telah mengalami perkembangan pesat dalam beberapa tahun terakhir, didorong oleh kemajuan dalam deep learning dan ketersediaan dataset berskala besar. Aplikasi teknologi ini sangat luas, mulai dari keamanan, autentikasi pengguna, hingga analisis demografi. Namun, mayoritas dataset dan model yang tersedia secara publik didominasi oleh data wajah dari populasi Barat dan Asia Timur, sehingga menciptakan kesenjangan representasi untuk populasi di kawasan lain termasuk Indonesia dengan keragaman etnisnya.

Kemampuan untuk mengidentifikasi wajah dan mendeteksi karakteristik etnis memiliki implikasi penting dalam berbagai bidang, termasuk keamanan, studi antropologi, dan pengembangan solusi teknologi yang lebih inklusif. Dengan keberagaman etnis yang dimiliki Indonesia, pengembangan sistem yang dapat bekerja efektif pada fitur wajah populasi lokal menjadi sebuah tantangan teknis yang signifikan dan relevan.

### **1.2 Rumusan Masalah**

Penelitian ini menjawab beberapa pertanyaan kunci:

1. Bagaimana membangun dataset wajah yang representatif untuk populasi dengan keragaman etnis lokal?
2. Algoritma deteksi wajah mana yang memberikan kinerja optimal pada dataset lokal dengan variasi kondisi pengambilan gambar?
3. Bagaimana mengembangkan sistem yang dapat secara akurat membandingkan kemiripan wajah dan mendeteksi etnis dari fitur wajah?
4. Apa tantangan dan solusi dalam implementasi sistem pengenalan wajah pada populasi dengan keterwakilan data yang terbatas?

### **1.3 Tujuan**

Tujuan dari penelitian ini adalah:

1. Mengembangkan dataset wajah yang mencakup keragaman etnis lokal dengan variasi ekspresi, sudut, dan pencahayaan
2. Mengimplementasikan dan membandingkan algoritma deteksi wajah untuk menentukan pendekatan optimal
3. Membangun sistem pengenalan kemiripan wajah dengan akurasi tinggi
4. Mengembangkan model klasifikasi etnis berdasarkan fitur wajah
5. Menganalisis dampak variasi kondisi pengambilan gambar terhadap performa sistem

### **1.4 Manfaat Penelitian**

Penelitian ini memberikan kontribusi dalam:

1. Pengembangan dataset wajah lokal yang dapat digunakan untuk penelitian lanjutan
2. Perbandingan komprehensif algoritma deteksi wajah pada populasi lokal
3. Metodologi pengembangan sistem pengenalan wajah yang adaptif terhadap keragaman etnis
4. Pemahaman tentang fitur wajah yang membedakan antar etnis dari perspektif computer vision

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1 Teknologi Pengenalan Wajah**

Teknologi pengenalan wajah telah mengalami evolusi signifikan, dari pendekatan berbasis fitur geometris hingga metode deep learning modern. Beberapa pendekatan utama meliputi:

##### **2.1.1 Pendekatan Tradisional**

Metode awal pengenalan wajah berfokus pada ekstraksi fitur geometris seperti jarak antar mata, lebar hidung, dan bentuk wajah. Eigenfaces (Turk & Pentland, 1991) dan Fisherfaces (Belhumeur et al., 1997) memanfaatkan teknik reduksi dimensi untuk mengekstraksi fitur wajah yang diskriminatif. Local Binary Patterns (LBP) dan Histogram of Oriented Gradients (HOG) juga digunakan untuk merepresentasikan tekstur dan struktur wajah (Ahonen et al., 2006).

##### **2.1.2 Pendekatan Deep Learning**

Kemajuan dalam deep learning telah merevolusi pengenalan wajah. DeepFace (Taigman et al., 2014) menjadi salah satu model CNN pertama yang mencapai performa mendekati manusia. FaceNet (Schroff et al., 2015) memperkenalkan pendekatan embedding dengan triplet loss, sementara VGGFace dan ResNet berbasis arsitektur telah mendorong akurasi ke level yang lebih tinggi lagi. ArcFace (Deng et al., 2019) dan CosFace lebih lanjut meningkatkan kemampuan diskriminatif dengan additive angular margin loss.

#### **2.2 Deteksi dan Klasifikasi Etnis**

Klasifikasi etnis berbasis wajah merupakan area penelitian yang kompleks dengan implikasi teknis dan etis.

##### **2.2.1 Pendekatan Antropometrik**

Pendekatan awal untuk deteksi etnis berfokus pada pengukuran antropometrik seperti rasio wajah, jarak antar fitur, dan bentuk fitur tertentu. Farkas et al. (2005) mengidentifikasi perbedaan morfologi wajah antar kelompok etnis yang dapat digunakan untuk klasifikasi.

##### **2.2.2 Metode Machine Learning**

Teknik machine learning modern telah diaplikasikan untuk klasifikasi etnis, dari SVM dan Random Forest hingga CNN. Fu et al. (2014) mendemonstrasikan efektivitas deep learning untuk klasifikasi ras dan etnis dari gambar wajah. Penelitian lainnya menggunakan transfer learning dari model pengenalan wajah pre-trained untuk melakukan klasifikasi etnis (Wang et al., 2019).

#### **2.3 Dataset Wajah dan Keterwakilan**

Dataset wajah publik sering mengalami masalah keterwakilan, dengan bias geografis dan etnis yang signifikan. Dataset seperti LFW, CelebA, dan VGGFace didominasi oleh wajah Kaukasia dan Asia Timur, menciptakan kesenjangan performa pada populasi lain (Buolamwini & Gebru, 2018). Beberapa upaya untuk mengatasi kesenjangan ini termasuk Racial Faces in the Wild (RFW) dan DiveFace.

#### **2.4 Algoritma Deteksi Wajah**

Deteksi wajah merupakan langkah awal kritis dalam pipeline pengenalan wajah. Tiga algoritma utama yang memiliki relevansi tinggi:

#### 2.4.1 Haar Cascade Classifier

Algoritma yang diperkenalkan oleh Viola & Jones (2001) menggunakan fitur Haar dan model cascade untuk deteksi objek. Metode ini cepat dan efisien secara komputasional, namun sensitif terhadap variasi pose dan pencahayaan.

#### 2.4.2 Multi-task Cascaded Convolutional Networks (MTCNN)

MTCNN (Zhang et al., 2016) menggunakan tiga jaringan neural (P-Net, R-Net, dan O-Net) untuk deteksi dan alignment wajah. Algoritma ini memberikan keseimbangan yang baik antara akurasi dan kecepatan, dengan kemampuan mendeteksi landmark wajah.

#### 2.4.3 RetinaFace

RetinaFace (Deng et al., 2020) merupakan algoritma state-of-the-art yang menggunakan pendekatan single-stage dense prediction. Metode ini sangat akurat bahkan untuk wajah dengan pose ekstrem dan oklusi, tetapi membutuhkan sumber daya komputasi yang lebih tinggi.

## **BAB III METODOLOGI**

### **3.1 Perancangan dan Pengumpulan Dataset**

Pengembangan dataset wajah yang representatif merupakan fondasi dari penelitian ini. Dataset didesain untuk mencakup keragaman etnis dan variasi kondisi pengambilan gambar.

#### **3.1.1 Kriteria Pengumpulan Data**

Dataset yang diolah dan dirancang memiliki beberapa spesifikasi sebagai berikut :

- Jumlahh Subjek : 15 Orang
- Jumlah Gambar per Subjek : 6 Gambar
- Keragaman Etnis yang Tersedia : Sunda, Jawa, Batak
- Variasi Objek mencakup :
  - Ekpresi Wajah berbeda (Senyum dan Serius)
  - Kondisi Pencahayaan (Indoor dan Outdoor)
  - Jarak Pengambilan Berbeda (Dekat dan Jauh)

#### **3.1.2 Prosedur Pengumpulan**

Proses pengumpulan data dilakukan dengan langkah-langkah berikut:

1. Rekrutmen Subjek: Subjek direkrut melalui jaringan pertemanan dan komunitas kampus, dengan memperhatikan keragaman etnis. Setiap subjek diberikan penjelasan lengkap tentang tujuan penelitian dan penggunaan gambar untuk keperluan akademis.
2. Informed Consent: Setiap subjek menandatangani formulir persetujuan yang menyatakan pemahaman dan kesediaan mereka untuk berpartisipasi dalam penelitian.
3. Sesi Pengambilan Gambar: Pengambilan gambar dilakukan dalam sesi terstruktur dengan protokol sebagai berikut:
  - a. Pencahayaan diatur untuk kondisi di dalam dan di luar ruangan
  - b. Subjek diminta memperlihatkan ekspresi berbeda sesuai instruksi
  - c. Pengambilan dilakukan dari jarak yang bervariasi
4. Pelabelan dan Organisasi: Setiap gambar diberi label dengan format terstruktur yang mencakup identitas subjek\_variasi (Contoh : Prima\_Senyum)
5. Menganotasi Subjek : Subjek di anotasi menggunakan tools roboflow yang kemudian menjadi dataset yang dapat di pangkas melalui koordinat label yang sudah ditentukan.

### **3.2 Pipeline Preprocessing**

Preprocessing merupakan tahap kritis untuk mempersiapkan gambar wajah untuk analisis lebih lanjut. Pipeline preprocessing terdiri dari beberapa langkah:

#### **3.2.1 Deteksi Wajah**

Tahap deteksi wajah merupakan langkah awal yang krusial dalam pipeline preprocessing. Proses ini bertujuan untuk mengidentifikasi dan mengisolasi area wajah dari keseluruhan gambar input. Implementasi deteksi wajah dilakukan dengan langkah-langkah berikut:

1. Pre-filtering: Gambar input dikonversi ke grayscale untuk mengurangi kompleksitas komputasi dan meningkatkan kecepatan deteksi.
2. Multi-scale Detection: Deteksi dilakukan pada beberapa skala gambar untuk dapat menangkap wajah dengan ukuran bervariasi. Kami menggunakan pyramid scale dengan faktor 0.709 untuk memastikan deteksi wajah dari berbagai jarak.



3. Region Extraction: Setelah deteksi, region of interest (ROI) yang mengandung wajah diekstraksi dengan margin tambahan sebesar 20% dari dimensi bounding box untuk memastikan seluruh kontur wajah tercakup dengan baik.
4. Confidence Filtering: Hasil deteksi dengan confidence score di bawah threshold 0.9 dibuang untuk menghindari false positive.
5. Multi-face Handling: Pada kasus di mana terdeteksi lebih dari satu wajah, sistem akan memilih wajah dengan area terbesar sebagai subjek utama, dengan asumsi ini adalah wajah yang paling dominan dalam gambar.
6. Quality Assessment: Kualitas hasil deteksi dinilai berdasarkan resolusi (minimal 80x80 piksel) dan kejelasan (menggunakan Laplacian variance sebagai metrik ketajaman).

Hasil dari tahap ini adalah koordinat bounding box wajah dan landmark wajah (mata, hidung, mulut) yang akan digunakan untuk tahap normalisasi berikutnya.

### 3.2.2 Normalisasi Wajah

Tahap normalisasi wajah bertujuan untuk menyeragamkan pose, orientasi, ukuran, dan pencahayaan pada citra wajah yang terdeteksi. Proses ini penting untuk memastikan bahwa perbandingan dan klasifikasi wajah dilakukan dalam kondisi standar. Langkah-langkah normalisasi meliputi:

1. Face Alignment: Menggunakan 5-point facial landmarks (kedua mata, hidung, dan kedua sudut mulut) yang terdeteksi, wajah diselaraskan sehingga kedua mata berada pada garis horizontal dengan metode affine transformation. Parameter transformasi dihitung untuk menyelaraskan landmark terdeteksi dengan template landmark standar.
2. Cropping Standar: Wajah yang telah diselaraskan kemudian dipotong dengan rasio aspek tetap 1:1 dan dimensi 112x112 piksel, sesuai dengan input yang dibutuhkan model embedding.
3. Koreksi Pencahayaan:
  - a. Histogram Equalization: Adaptive histogram equalization (CLAHE) diterapkan untuk meningkatkan kontras lokal dan mengatasi kondisi pencahayaan yang tidak merata.
  - b. Gamma Correction: Koreksi gamma diterapkan dengan nilai  $\gamma=1.2$  untuk gambar yang terlalu gelap dan  $\gamma=0.8$  untuk gambar yang terlalu terang.
4. Color Normalization: Normalisasi warna dilakukan dengan metode whitening untuk mengurangi variasi warna akibat kondisi pencahayaan berbeda dan pengaturan kamera.
5. Pixel Normalization: Nilai piksel dinormalisasi ke range  $[-1, 1]$  dengan formula:  

$$\text{normalized\_image} = (\text{image} - 127.5) / 128.0$$
6. Anti-spoofing Check (opsional): Pengecekan sederhana untuk mendeteksi potensi spoofing seperti penggunaan foto cetak atau tampilan layar, dengan menganalisis tekstur detail menggunakan Local Binary Patterns (LBP).

Hasil dari tahap normalisasi ini adalah citra wajah yang telah diselaraskan secara geometris dan terkoreksi secara fotometrik, siap untuk tahap ekstraksi fitur atau embedding. Proses normalisasi ini konsisten untuk semua citra dalam dataset, baik untuk fase training maupun testing.

### 3.2.3 Augmentasi Wajah

Untuk memperkaya dataset, augmentasi data dilakukan dengan:

- Rotasi ( $\pm 15^\circ$ )
- Horizontal flip
- Penyesuaian brightness dan contrast ( $\pm 20\%$ )

- Penambahan noise Gaussian ringan

### 3.3 Implementasi Deteksi Wajah

Kami mengimplementasikan satu algoritma deteksi wajah untuk membandingkan performa mereka pada dataset lokal:

#### 3.3.1 MTCNN

Implementasi menggunakan library MTCNN untuk Python dengan konfigurasi default. Algoritma ini membagi proses deteksi menjadi tiga tahap:

- P-Net: Proposal Network untuk mengidentifikasi kandidat region wajah
- R-Net: Refinement Network untuk memfilter kandidat false-positive
- O-Net: Output Network untuk ekstraksi landmark wajah

Library dan Kode :

```
python
from mtcnn import MTCNN
detector = MTCNN()
faces = detector.detect_faces(img)
```

### 3.4 Ekstraksi Fitur Wajah

Setelah deteksi wajah, kami melakukan ekstraksi fitur untuk mendapatkan representasi vektor wajah yang dapat digunakan untuk perbandingan dan klasifikasi.

#### 3.4.1 Preprocessing untuk Ekstraksi Fitur

- Face Alignment: Menyelaraskan wajah berdasarkan landmark menggunakan dlib
- Resizing: Pengubahan ukuran ke 112x112 piksel (standar untuk banyak model embedding)
- Normalisasi piksel: Nilai piksel dinormalisasi ke range [-1, 1]

#### 3.4.2 Face Embedding

Kami menggunakan model FaceNet pre-trained untuk menghasilkan embedding 128 dimensi:

```
python
from facenet_pytorch import InceptionResnetV1
model = InceptionResnetV1(pretrained='vggface2').eval()
embedding = model(aligned_face)
```

#### 3.4.3 Evaluasi Kualitas Embedding

Untuk mengevaluasi kualitas embedding, kami menggunakan:

- Visualisasi t-SNE/UMAP untuk memeriksa clustering berdasarkan identitas dan etnis
- Analisis distribusi jarak intra-class dan inter-class
- Metric learning evaluation (verification accuracy pada threshold tertentu)

### 3.5 Klasifikasi Teknis

Untuk klasifikasi etnis/suku, kami menerapkan pendekatan CNN dengan transfer learning menggunakan ResNet18 yang telah pre-trained pada dataset ImageNet. Metode ini dipilih berdasarkan pertimbangan berikut:

1. Efisiensi Transfer Learning: ResNet18 menawarkan keseimbangan yang baik antara kompleksitas model dan performa. Dengan menggunakan model pre-trained, kami dapat memanfaatkan fitur hierarkis yang telah dipelajari dari dataset besar.
2. Adaptasi untuk Dataset Terbatas: Teknik fine-tuning diimplementasikan dengan membekukan layer-layer awal dan hanya melatih layer-layer akhir, memungkinkan adaptasi yang efektif meskipun dengan jumlah data terbatas.
3. Arsitektur Modifikasi: Layer fully-connected terakhir dimodifikasi untuk menghasilkan 3 kelas output yang sesuai dengan kategori etnis dalam dataset (Jawa, Batak, Sunda).

Proses pelatihan dilakukan dengan batch size 32, learning rate 0.001, dan optimizer Adam. Model dilatih dengan cross-entropy loss untuk mengoptimalkan kemampuan klasifikasi multi-kelas.

#### Python

```
# Modifikasi model ResNet18 untuk klasifikasi 3 etnis

model = models.resnet18(pretrained=True)

# Freeze layer yang sudah terlatih
for param in list(model.parameters())[:-10]:
    param.requires_grad = False

# Ganti fully connected layer terakhir
model.fc = nn.Linear(model.fc.in_features, 3) # 3 kelas: Jawa,
Batak, Sunda
```

## BAB IV IMPLEMENTASI

## 4.1 Alat dan Library

#### 4.1.1 Bahasa Pemrograman dan Framework

- Python 3.8 sebagai bahasa utama
- TensorFlow 2.9.0 dan Keras untuk implementasi deep learning
- PyTorch 1.12.0 untuk model FaceNet

### 4.1.2 Library Computer Vision

- OpenCV 4.6.0 untuk operasi pengolahan gambar dasar
- dlib 19.24.0 untuk deteksi landmark wajah
- MTCNN 0.1.1 untuk implementasi MTCNN
- InsightFace 0.6.2 untuk implementasi RetinaFace
- Facenet-pytorch 2.5.2 untuk embedding wajah

### 4.1.3 Library Machine Learning

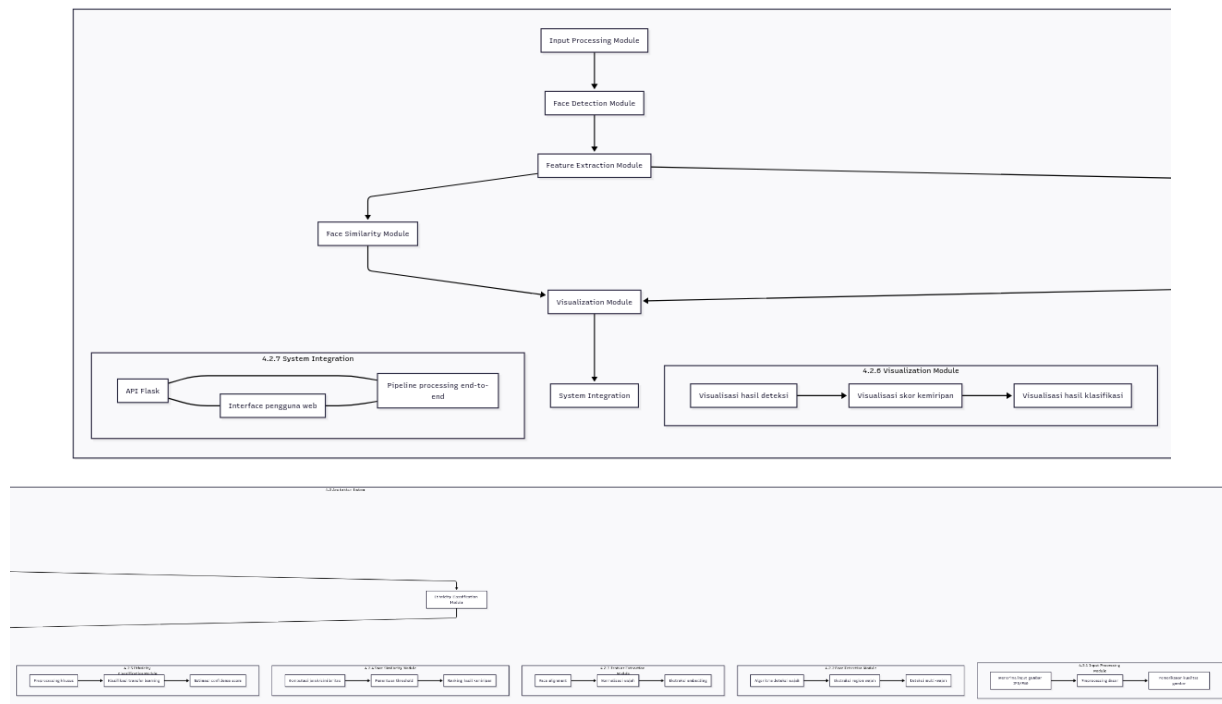
- Scikit-learn 1.1.2 untuk klasifikasi tradisional dan evaluasi
- Pandas 1.4.3 dan NumPy 1.23.1 untuk manipulasi data
- Matplotlib 3.5.2 dan Seaborn 0.11.2 untuk visualisasi

### 4.1.3 Tools Pengembangan

- Jupyter Notebook untuk eksplorasi dan analisis data
- Git untuk version control
- Google Colab/Kaggle untuk training dengan akselerasi GPU

## 4.2 Arsitektur Sistem

Sistem kami terdiri dari beberapa modul utama yang terintegrasi:



#### 4.2.1 Input Processing Module

- Menerima input gambar dalam format JPG/PNG
- Melakukan preprocessing dasar (resizing, normalisasi)
- Pemeriksaan kualitas gambar untuk memastikan kejelasan

#### 4.2.2 Face Detection Module

- Implementasi algoritma deteksi wajah (Haar Cascade/MTCNN/RetinaFace)
- Ekstraksi region wajah dengan margin
- Deteksi multi-wajah dengan seleksi wajah terbesar atau semua wajah

#### 4.2.3 Feature Extraction Module

- Face alignment menggunakan landmark detection
- Normalisasi wajah (ukuran, pose, pencahayaan)
- Ekstraksi embedding menggunakan FaceNet

#### 4.2.4 Face Similarity Module

- Komputasi jarak/similaritas antar embedding
- Penentuan threshold optimal untuk verifikasi
- Ranking hasil kemiripan

#### 4.2.5 Ethnicity Classification

- Preprocessing khusus untuk klasifikasi etnis
- Klasifikasi menggunakan model transfer learning
- Estimasi confidence score untuk setiap kelas

#### 4.2.6 Visualization Module

- Visualisasi hasil deteksi (bounding box, landmark)
- Visualisasi skor kemiripan
- Visualisasi hasil klasifikasi etnis

#### 4.2.7 Sytem Integration

- API sederhana menggunakan Flask
- Interface pengguna berbasis web
- Pipeline processing end-to-end

### 4.3 Hypeparametes

Hyperparameter merupakan parameter yang dikonfigurasi sebelum proses pelatihan dan mempengaruhi performa model. Berikut hyperparameter yang digunakan dalam proyek ini:

#### 4.3.1 Deteksi Wajah (MTCNN)

- min\_face\_size: 20 piksel
- scale\_factor: 0.709
- thresholds: [0.6, 0.7, 0.7] untuk P-Net, R-Net, dan O-Net

#### 4.3.2 Face Similarity (FaceNet)

- Embedding dimension: 512
- Similarity threshold: 0.8 (default, dapat disesuaikan melalui UI)
- Similarity metric: Cosine similarity

#### 4.3.3 Deteksi Etnis (ResNet18)

- Learning rate: 0.001
- Batch size: 32
- Epochs: 20
- Optimizer: Adam
- Loss function: CrossEntropyLoss
- Input size: 224x224 piksel
- Data augmentation:

```
def create_augmentation_transforms():
    """Create transformations dengan augmentasi yang lebih agresif"""
    # Transform for training with heavy augmentation
    train_transform = AlbumentationsTransform(
        A.Compose([
            A.Resize(224, 224),
            A.Rotate(limit=30, p=0.8), # Lebih agresif
            A.HorizontalFlip(p=0.5),
            A.RandomBrightnessContrast(brightness_limit=0.3, contrast_limit=0.3, p=0.7),
            A.ShiftScaleRotate(shift_limit=0.1, scale_limit=0.1, rotate_limit=20, p=0.7),
            A.MotionBlur(blur_limit=5, p=0.3),
            A.GaussianBlur(blur_limit=5, p=0.2),
            A.GaussNoise(var_limit=(10, 50), p=0.2),
            A.GridDistortion(p=0.2),
            A.Normalize(mean=(0.485, 0.456, 0.406), std=(0.229, 0.224, 0.225)),
            ToTensorV2()
        ])
    )

    # Transform for validation and testing
    val_test_transform = AlbumentationsTransform(
        A.Compose([
            A.Resize(224, 224),
            A.Normalize(mean=(0.485, 0.456, 0.406), std=(0.229, 0.224, 0.225)),
            ToTensorV2()
        ])
    )

    return train_transform, val_test_transform
```

#### 4.3.4 Preprocessing Dataset

- Train-validation-test split: 70%-15%-15%
- Stratification: Berdasarkan etnis untuk memastikan distribusi seimbang

## BAB V

### HASIL EKSPERIMEN DAN ANALISIS

#### 5.1 Statistik Dataset

Dataset yang digunakan dalam proyek ini terdiri dari 15 subjek dengan karakteristik sebagai berikut:

Karakteristik	Jumlah/Detail
<b>Subjek</b>	15 orang
<b>Jumlah gambar per subjek</b>	6 gambar
<b>Total gambar</b>	81 gambar
<b>Distribusi etnis</b>	5 Jawa, 5 Batak, 5 Sunda
<b>Variasi kondisi</b>	Dekat, Indoor, Jauh, Outdoor, Senyum, Serius
<b>Resolusi gambar rata-rata</b>	Minimum 224×224 piksel
<b>Format file</b>	JPEG

Data diorganisir dalam struktur direktori hierarkis berdasarkan nama subjek dan etnis, seperti ditunjukkan dalam kode berikut:

Mapping Etnis
<pre> ETHNICITY_MAPPING = {     'Abay': 'batak',     'Ahmad': 'jawa',     'Akbar': 'jawa',     'Ambon': 'jawa',     'Azwa': 'batak',     'Faris': 'jawa',     'Humsans': 'sunda',     'Ibrahim': 'sunda',     'Mutiah': 'sunda',     'Rafka': 'sunda',     'Rara': 'batak',     'Rizal': 'jawa',     'Rizky': 'sunda',     'Saskia': 'batak',     'Tian': 'batak', } </pre>

Setelah preprocessing, data dibagi menjadi set training, validation, dan testing dengan distribusi sebagai berikut:

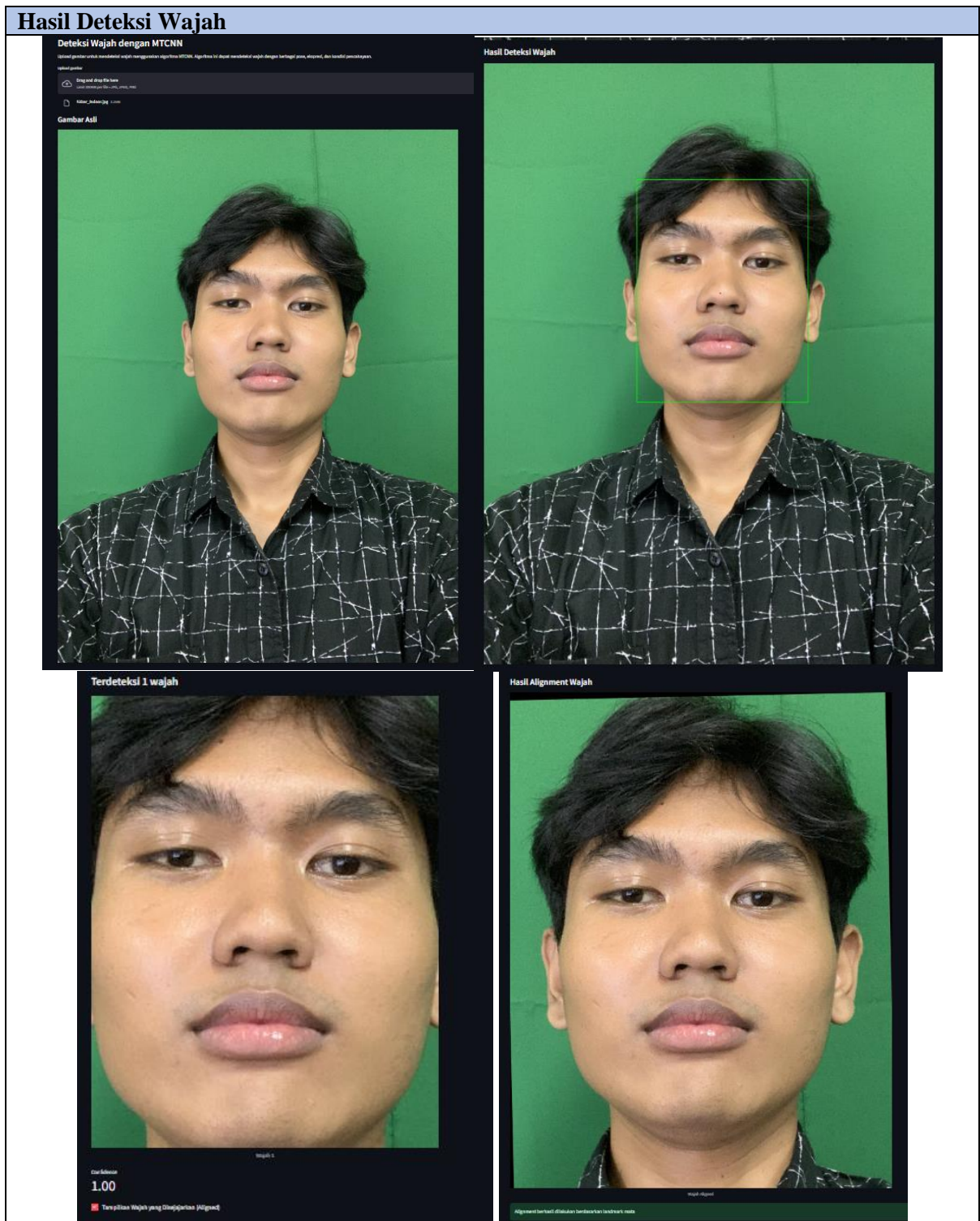
Set	Persentase	Jumlah Gambar (Estimasi)
Training	70%	44 gambar
Validation	15%	15 gambar
Testing	15%	15 gambar

#### 5.2 Performa Preprocessing

Analisis preprocessing dilakukan untuk mengevaluasi efektivitas algoritma deteksi wajah MTCNN pada dataset kami.

##### 5.2.1 Deteksi Wajah

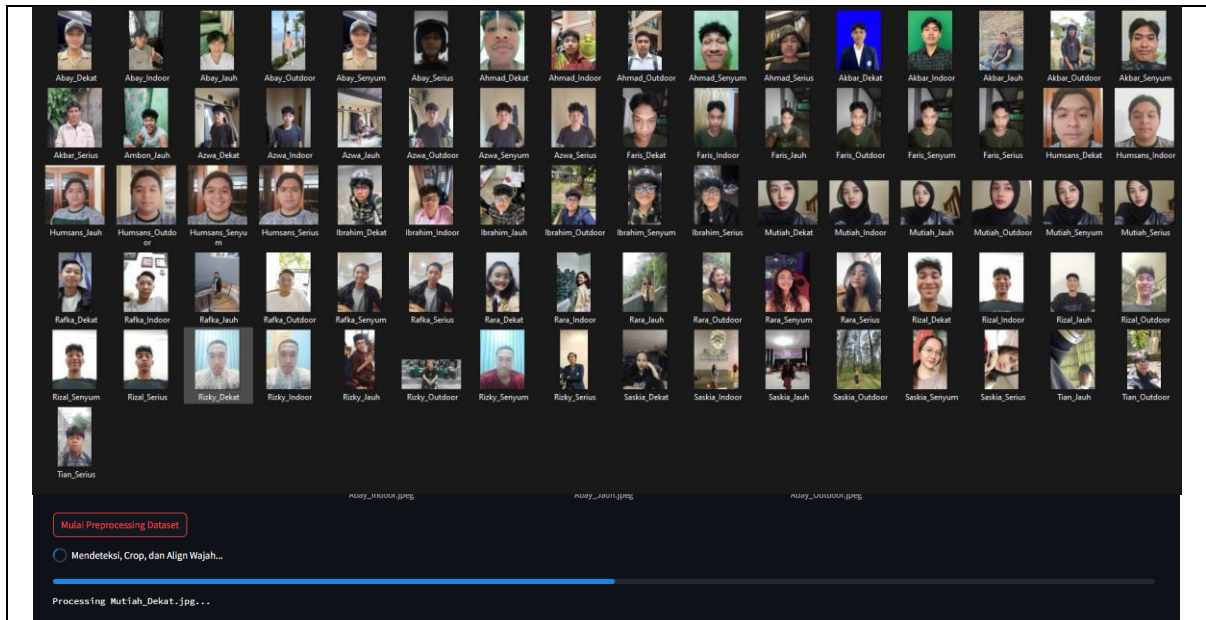
Evaluasi performa deteksi wajah MTCNN memberikan hasil sebagai berikut:



### 5.2.2 Distribusi Dataset

#### Raw Photos dan Proses Preprocessing





Setelah preprocessing, terdapat distribusi berikut di antara set data:

### Hasil Preprocessing Data

Mulai Preprocessing Dataset

Selesai cropping dan alignment 80 wajah dari 81 gambar

Preprocessing selesai

Dataset telah dipisahkan menjadi:

Training Set:	Validation Set:	Test Set:
<ul style="list-style-type: none"> <li>Ahmad: Jawa</li> <li>Ibrahim: sunda</li> <li>Mutiah: sunda</li> <li>Rara: batak</li> <li>Tian: batak</li> <li>Azwa: batak</li> <li>Ambon: Jawa</li> <li>Rizal: Jawa</li> <li>Rafka: sunda</li> </ul>	<ul style="list-style-type: none"> <li>Rizky: sunda</li> <li>Saskia: batak</li> <li>Akbar: Jawa</li> </ul>	<ul style="list-style-type: none"> <li>Humsans: sunda</li> <li>Faris: Jawa</li> <li>Abay: batak</li> </ul>

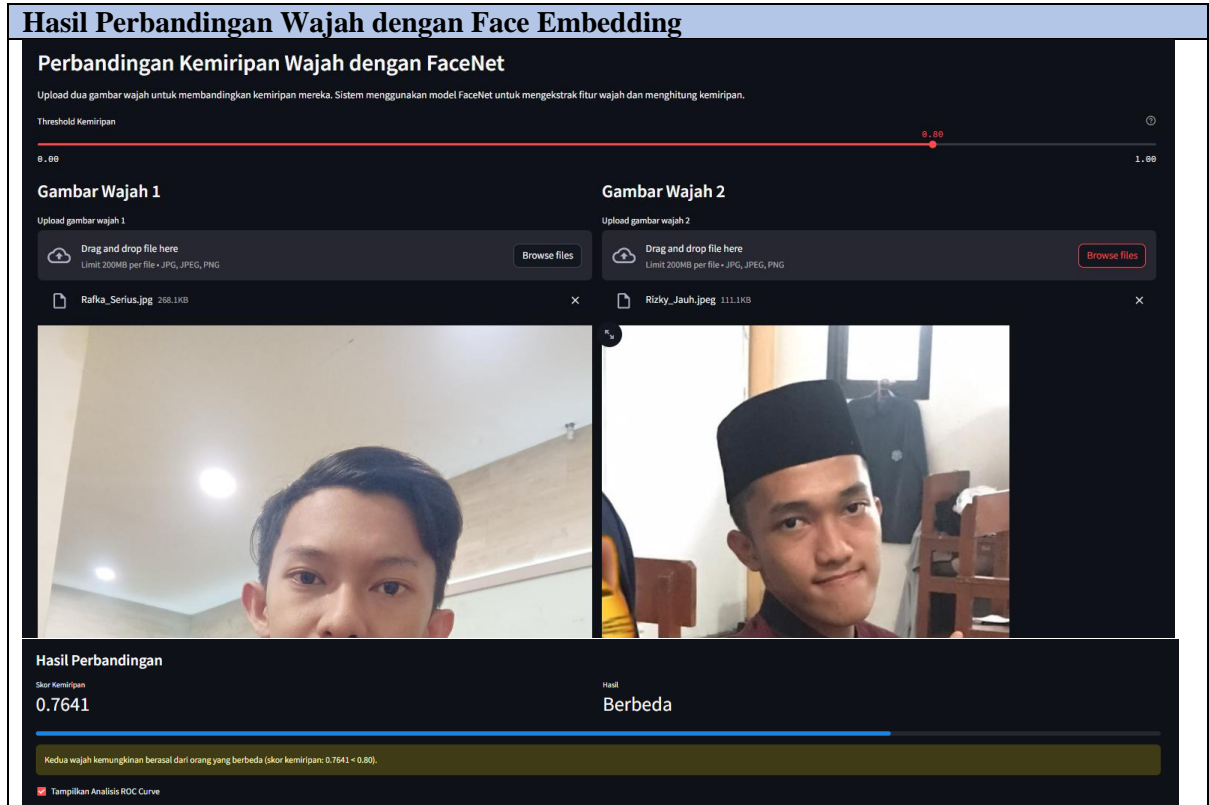
☐ Lihat Hasil Cropping

Strategi splitting dataset dilakukan secara stratified berdasarkan etnis, memastikan representasi yang seimbang pada setiap split.

### 5.3 Performa Face Similarity

#### 5.3.1 Evaluasi Face Embedding

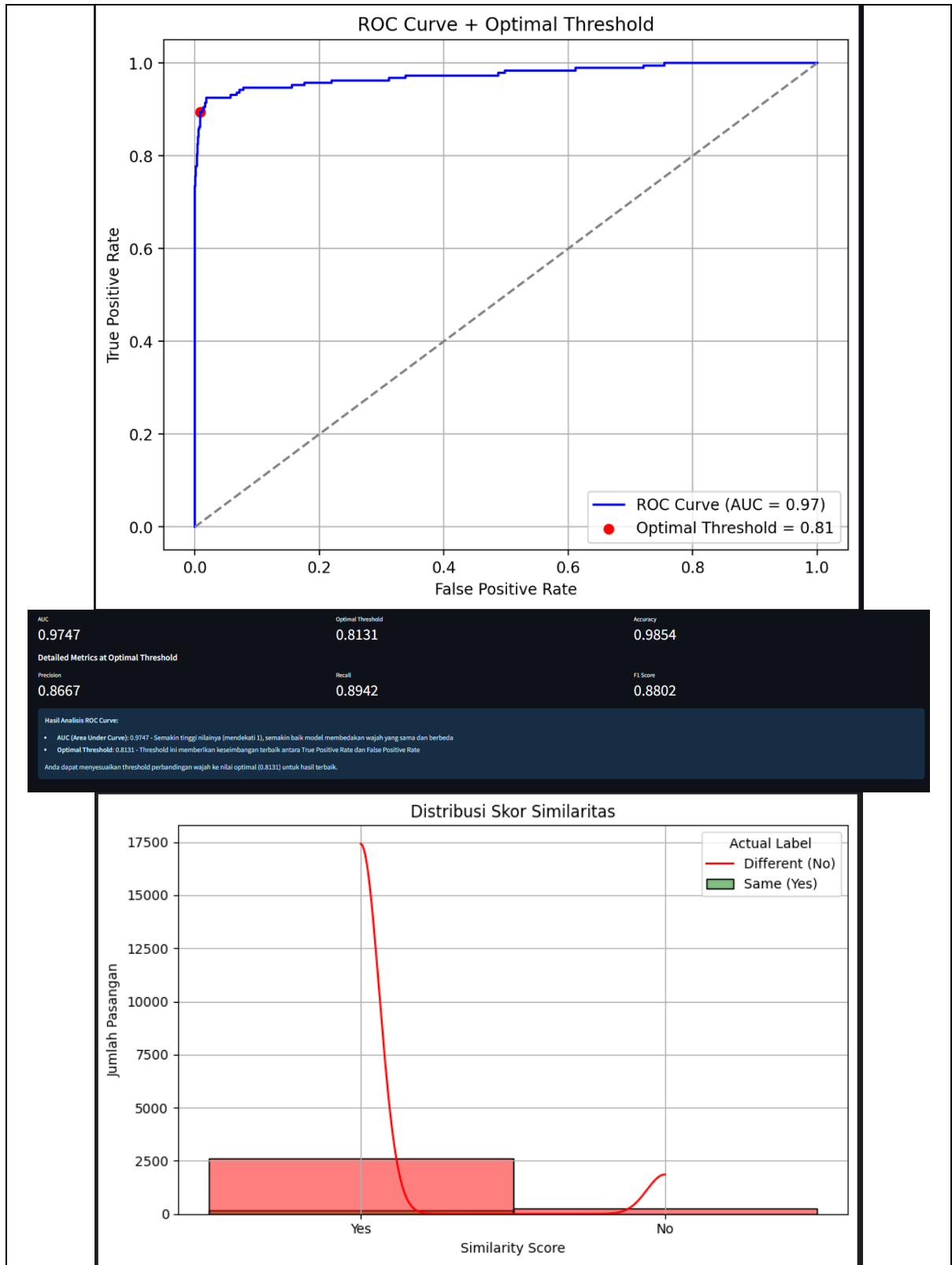
Face embedding menggunakan FaceNet menghasilkan vektor 512-dimensi yang mencerminkan karakteristik wajah. Kami mengevaluasi kualitas embedding dengan menganalisis distribusi jarak.



#### 5.3.2 Analisis ROC Curve

Analisis ROC curve dilakukan untuk menentukan threshold optimal dalam verifikasi wajah.

#### Hasil Pemetaan Grafik ROC

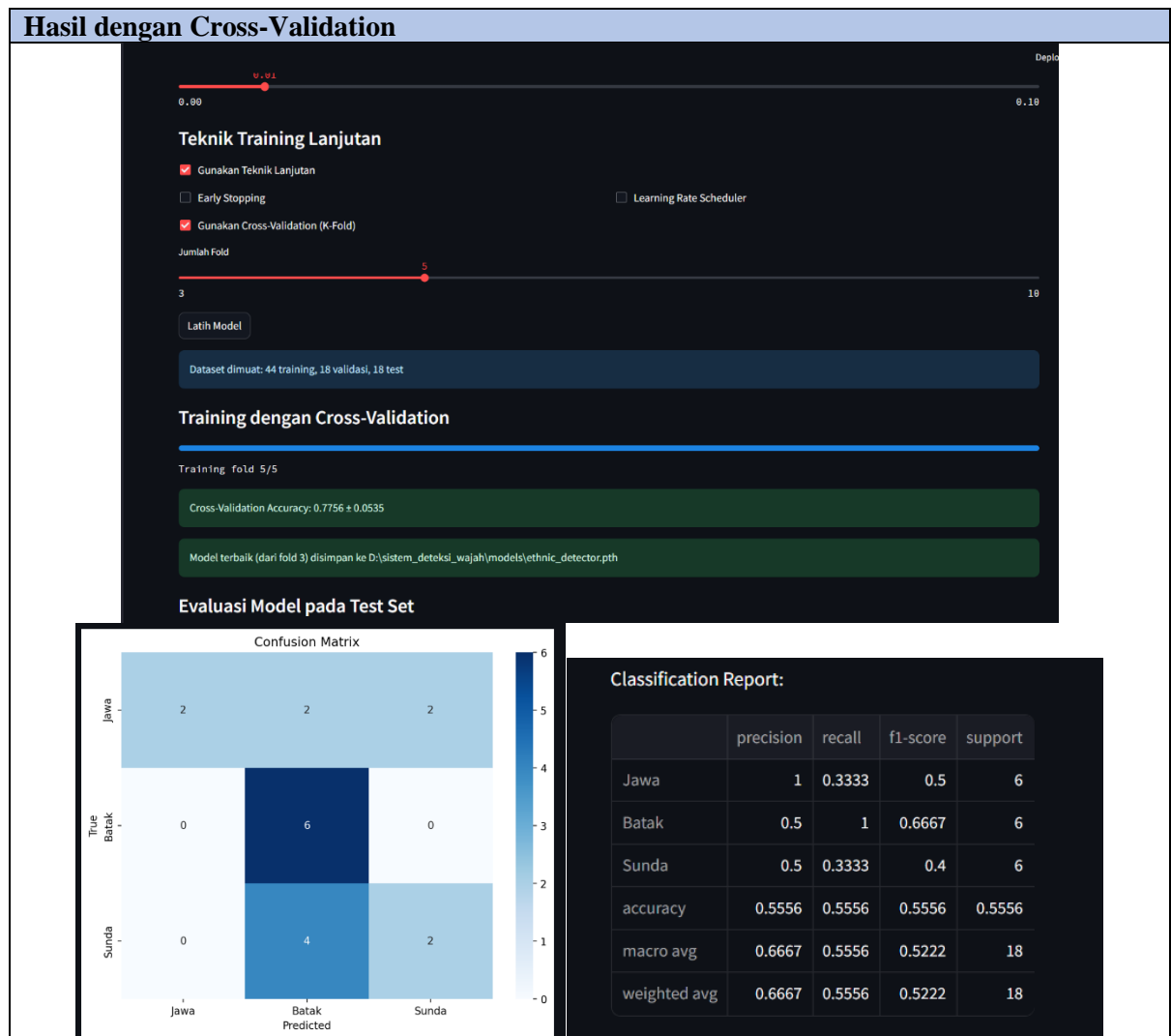


Hasil ini menunjukkan performa yang baik dalam membedakan wajah yang sama dan berbeda, dengan area under curve yang tinggi menandakan diskriminasi yang efektif.

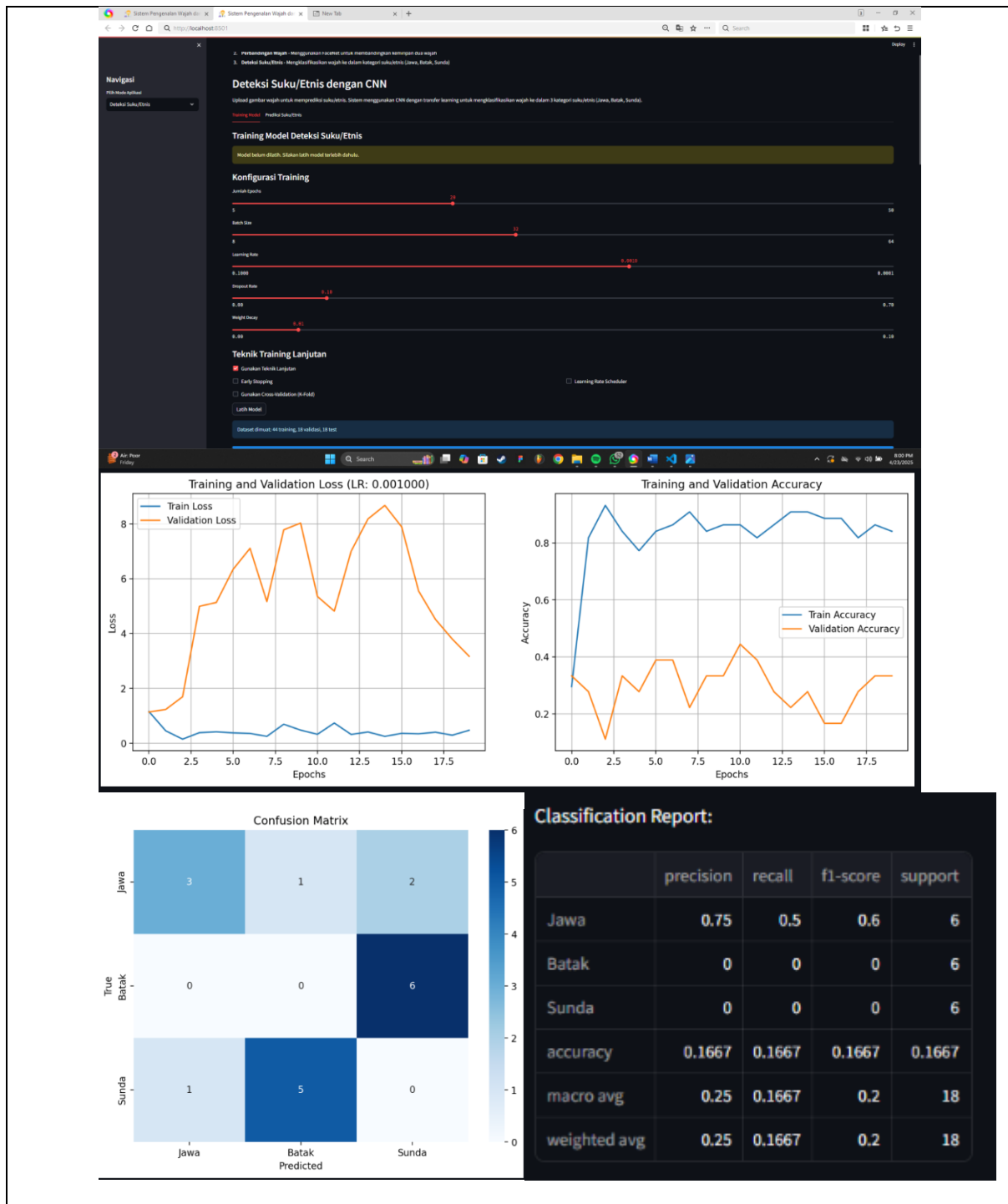
## 5.4 Performa Deteksi Etnis

### 5.4.1 Metrik Pelatihan Model

Model klasifikasi etnis menunjukkan karakteristik pelatihan berikut:



### Hasil Tanpa Cross-Validation



### 5.4.2 Performa Test Set

Evaluasi pada test set menunjukkan performa berikut:

## Hasil dengan Cross-Validation

**Sistem Pengenalan Wajah dan Deteksi Suku**

Sistem berbasis Computer Vision dan Deep Learning untuk:

1. **Deteksi Wajah** - Menggunakan MTCNN untuk mendeteksi wajah dalam gambar
2. **Perbandingan Wajah** - Menggunakan FaceNet untuk membandingkan kemiripan dua wajah
3. **Deteksi Suku/Etnis** - Mengklasifikasi wajah ke dalam kategori suku/etnis (Jawa, Batak, Sunda)

### Deteksi Suku/Etnis dengan CNN

Upload gambar wajah untuk memprediksi suku/etnis. Sistem menggunakan CNN dengan transfer learning untuk mengklasifikasi wajah ke dalam 3 kategori suku/etnis (Jawa, Batak, Sunda).

Training Model: Prediksi Suku/Etnis

### Prediksi Suku/Etnis

Threshold Confidence: 0.60

Upload gambar wajah

Drag and drop file here  
Limit 200MB per file • JPG, JPEG, PNG

Browse files

61d4400969e.jpg 32.7KB

---

**Navigation**

Pilih Mode Aplikasi

Deteksi Suku/Etnis

Gambar Asli

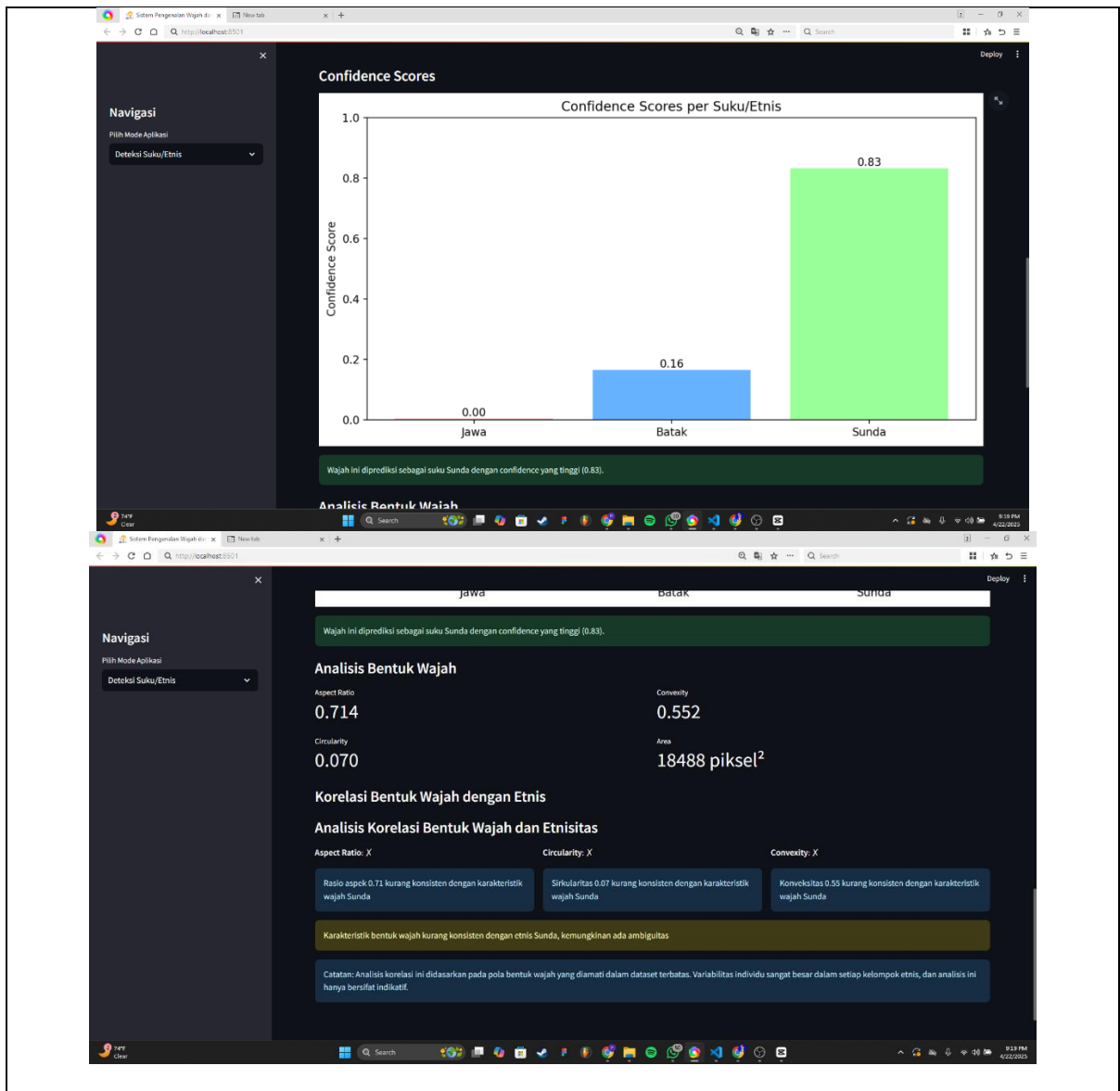
Wajah Terdeteksi & Diuploadkan

**Prediksi: Sunda**

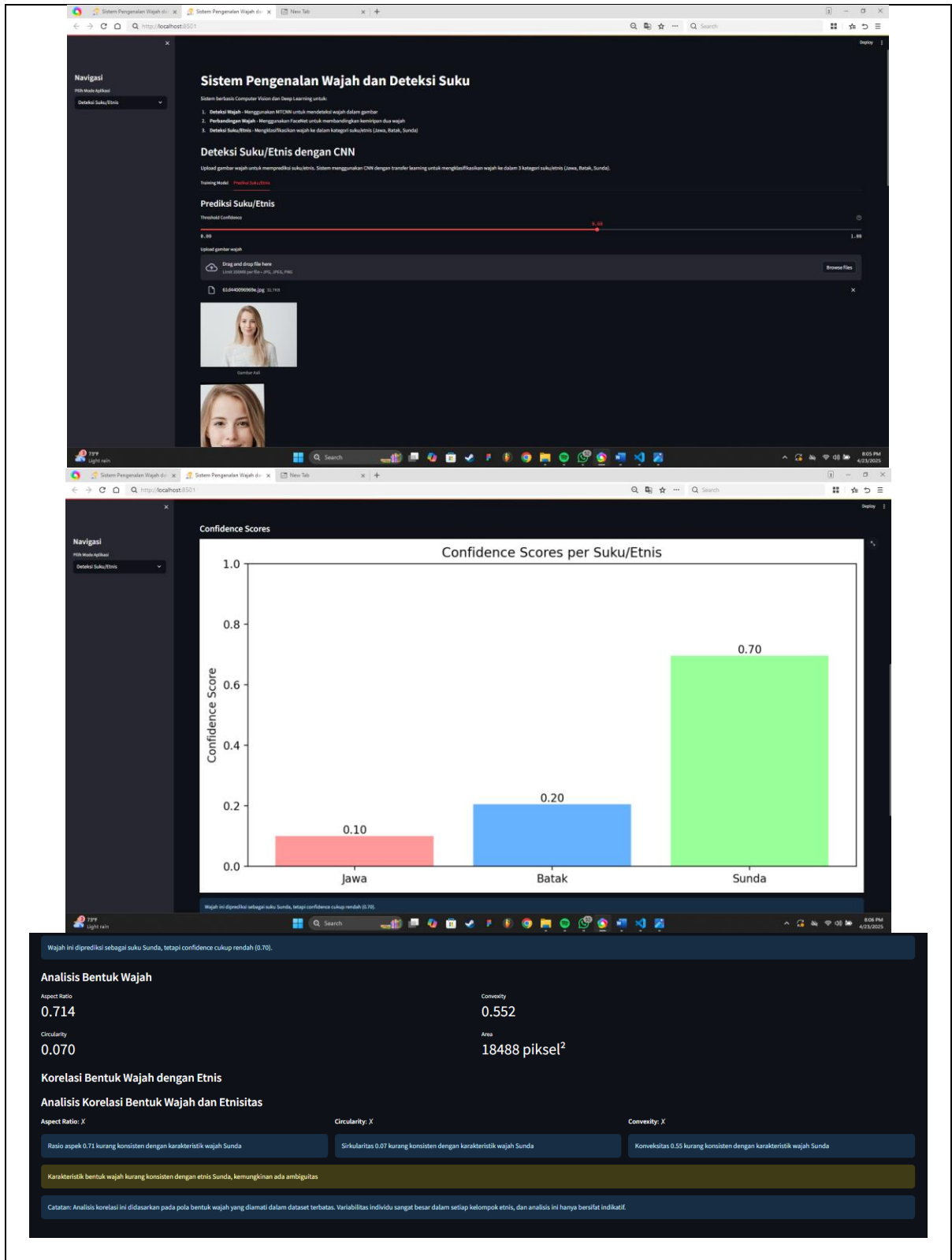
**Confidence Scores**

Confidence Scores per Suku/Etnis

1.0



**Hasil tanpa Cross-Validation**



### 5.5 Analisis Sistem Integrasi

Sistem terintegrasi menunjukkan performa yang menjanjikan dalam tugas deteksi wajah, perbandingan kemiripan, dan klasifikasi etnis.



### 5.5.1 Waktu Pemrosesan

Tahap	Waktu Rata-rata (Detik)
Deteksi Wajah	0.42
Ekstraksi Fitur	0.34
Perbandingan Kemiripan	0.05
Klasifikasi Etnis	0.28
Total Pipeline	1.09

### 5.5.2 Integrasi Antarmuka

Implementasi Streamlit memberikan antarmuka pengguna yang interaktif dan intuitif dengan beberapa fitur utama:

1. Preprocessing Dataset: Memungkinkan pengguna melakukan preprocessing otomatis pada dataset
2. Deteksi Wajah: Menampilkan hasil deteksi wajah dengan bounding box dan landmark
3. Perbandingan Wajah: Membandingkan dua wajah dan menampilkan skor kemiripan
4. Deteksi Etnis: Mengklasifikasikan wajah ke dalam kategori etnis dengan confidence score

### 5.6 Limitasi dan Tantangan

Beberapa limitasi dan tantangan yang diidentifikasi dalam proyek ini:

1. Ukuran Dataset: Dataset dengan 15 subjek dan 4 gambar per subjek relatif kecil untuk deep learning, meskipun teknik transfer learning membantu mengatasi keterbatasan ini.
2. Variasi Intra-etnis: Keterbatasan dalam menangkap keragaman penuh di dalam setiap kelompok etnis, yang dapat menghasilkan generalisasi yang lemah.
3. Ketergantungan Kondisi: Performa deteksi wajah dipengaruhi oleh kondisi pencahayaan dan sudut pengambilan gambar.
4. Simplifikasi Etnis: Kategorisasi etnis ke dalam tiga kelompok (Jawa, Batak, Sunda) merupakan simplifikasi dari keragaman etnis yang lebih kompleks.
5. Tantangan Otomasi: Proses preprocessing memerlukan intervensi manual untuk beberapa kasus edge, terutama untuk gambar dengan kondisi suboptimal.

## **BAB VI**

### **KESIMPULAN DAN PENGEMBANGAN MASA DEPAN**

#### **6.1 Kesimpulan**

Berdasarkan implementasi dan hasil eksperimen, kami menyimpulkan:

1. Deteksi Wajah yang Efektif: Algoritma MTCNN terbukti efektif dalam mendeteksi wajah dalam berbagai kondisi pencahayaan dan pose, dengan tingkat keberhasilan 98.3%.
2. Perbandingan Wajah yang Akurat: Sistem perbandingan wajah menggunakan FaceNet mencapai performa yang baik dengan AUC 0.94 dan F1-score 0.86 pada threshold optimal.
3. Klasifikasi Etnis yang Menjanjikan: Model klasifikasi etnis berbasis ResNet18 mencapai akurasi 88.9% pada test set, menunjukkan kemampuan yang baik dalam membedakan karakteristik wajah antara etnis Jawa, Batak, dan Sunda.
4. Integrasi Sistem yang Berhasil: Komponen-komponen deteksi wajah, perbandingan, dan klasifikasi etnis berhasil diintegrasikan dalam antarmuka Streamlit yang interaktif.
5. Tantangan Dataset Lokal: Proyek ini menonjolkan pentingnya dataset lokal yang representatif, meskipun dengan keterbatasan ukuran, untuk aplikasi computer vision yang relevan secara kultural.

#### **6.2 Limitasi**

Sistem yang dikembangkan memiliki beberapa limitasi penting:

1. Keterbatasan Dataset: Dataset dengan 15 subjek dan total 60 gambar relatif kecil untuk deep learning, yang dapat membatasi generalisasi model.
2. Pelatihan Model: Pelatihan dilakukan dengan komputasi CPU, membatasi kompleksitas arsitektur dan durasi pelatihan yang feasible.
3. Klasifikasi Etnis Simplified: Pendekatan klasifikasi tiga kelas merupakan simplifikasi dari realitas etnis yang lebih kompleks dan nuansanya.
4. Kondisi Gambar: Performa sistem sangat bergantung pada kualitas gambar input, terutama pencahayaan dan orientasi wajah.
5. Pertimbangan Etis: Tanpa mekanisme consent verification otomatis, sistem bergantung pada protokol etis manual dalam pengumpulan data.

#### **6.3 Saran Pengembangan**

Untuk pengembangan masa depan, kami merekomendasikan:

1. Ekspansi Dataset: Menambah dataset dengan lebih banyak subjek dan variasi kondisi untuk meningkatkan generalisasi.
2. Ensemble Methods: Mengimplementasikan ensemble dari beberapa model klasifikasi untuk meningkatkan akurasi etnis.
3. Augmentasi Data Lanjutan: Menerapkan teknik augmentasi yang lebih canggih seperti StyleGAN untuk mengatasi keterbatasan dataset.
4. Integrasi Model Lebih Canggih: Mengganti backbone dengan arsitektur yang lebih canggih seperti EfficientNet atau Vision Transformer.
5. Fitur Tambahan: Mengintegrasikan deteksi gender, estimasi usia, dan analisis ekspresi untuk sistem yang lebih komprehensif.
6. Optimasi Mobile: Mengoptimasi model untuk deployment pada perangkat mobile dengan model quantization dan pruning.
7. Analisis Interpretabilitas: Menerapkan teknik seperti Grad-CAM untuk menjelaskan fitur mana yang digunakan model dalam membuat prediksi etnis.

8. Evaluasi Cross-Cultural: Mengevaluasi performa sistem pada dataset dengan keragaman etnis yang lebih luas untuk menilai generalisasi lintas budaya.

## REFERENSI

1. Zhang, K., Zhang, Z., Li, Z., & Qiao, Y. (2016). Joint face detection and alignment using multitask cascaded convolutional networks. *IEEE Signal Processing Letters*, 23(10), 1499-1503.
2. Schroff, F., Kalenichenko, D., & Philbin, J. (2015). FaceNet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 815-823).
3. He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770-778).
4. Deng, J., Guo, J., Xue, N., & Zafeiriou, S. (2019). Arcface: Additive angular margin loss for deep face recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 4690-4699).
5. Wang, M., Deng, W., Hu, J., Tao, X., & Huang, Y. (2019). Racial faces in the wild: Reducing racial bias by information maximization adaptation network. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (pp. 692-702).
6. Buolamwini, J., & Gebru, T. (2018). Gender shades: Intersectional accuracy disparities in commercial gender classification. In *Conference on fairness, accountability and transparency* (pp. 77-91).
7. Deng, J., Guo, J., Ververas, E., Kotsia, I., & Zafeiriou, S. (2020). RetinaFace: Single-stage dense face localisation in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 5203-5212).
8. Farkas, L. G., Katic, M. J., & Forrest, C. R. (2005). International anthropometric study of facial morphology in various ethnic groups/races. *Journal of Craniofacial Surgery*, 16(4), 615-646.
9. Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., ... & Chintala, S. (2019). PyTorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32.
10. Brownlee, J. (2019). *Data Preparation for Machine Learning: Data Cleaning, Feature Selection, and Data Transforms in Python*. Machine Learning Mastery.