



Anggota: **Giovanni Lucy Faustine Sitompul (121140060)**

Tugas: **Final Project**

Anggota: **Qaisya Dwi Aryana (121140063)**

Anggota: **Marsella Yesi Natalia Sinaga (121140174)**

Mata Kuliah: **Sistem Teknologi Multimedia (IF4021)**

Tanggal: 24 Desember 2024

Ringkasan

Real Time Mood Scanner adalah aplikasi yang dirancang untuk mendeteksi suasana hati pengguna secara real-time menggunakan teknologi pembelajaran mesin. Proyek ini bertujuan untuk menyediakan alat yang dapat membantu dalam analisis emosional melalui ekspresi wajah atau input lain yang relevan. Hasil proyek menunjukkan bahwa aplikasi ini mampu mendeteksi berbagai suasana hati dengan akurasi yang memadai, meskipun terdapat beberapa tantangan dalam hal akurasi dan responsivitas yang perlu diperbaiki di masa mendatang.

1 Pendahuluan

1.1 Latar Belakang

Dalam era digital saat ini, telah menjadi kebutuhan utama dalam berbagai bidang, termasuk analisis emosi. Proyek Real-Time Mood Scanner bertujuan untuk menciptakan sistem yang mampu mendeteksi suasana hati secara langsung melalui pengolahan data visual dan algoritma pembelajaran mesin. Dengan meningkatnya kebutuhan akan solusi yang dapat memahami dan merespons emosi pengguna, proyek ini menjadi relevan untuk berbagai aplikasi, mulai dari kesehatan mental hingga pemasaran digital.

1.2 Rumusan Masalah

Bagaimana membangun sistem yang dapat mendeteksi suasana hati secara real-time dengan tingkat akurasi tinggi?

1.3 Tujuan

Mengembangkan filter yang mampu mendeteksi suasana hati pengguna secara real-time

1.4 Ruang Lingkup

Proyek ini hanya berfokus pada pengembangan filter sederhana yang:

1. Mendeteksi ekspresi wajah menggunakan kamera.
2. Mengklasifikasikan suasana hati menjadi beberapa kategori dasar seperti senang, sedih, dan terkejut.

1.5 Metodologi

1. Penggunaan MediaPipe atau pustaka serupa untuk deteksi wajah.
2. Implementasi model pembelajaran mesin sederhana untuk klasifikasi suasana hati.
3. Integrasi model ke filter real-time menggunakan Python.

2 Deskripsi Proyek

2.1 Rincian Tugas

1. Giovanni Lucy Faustine Sitompul:
 - Melatih model dan menyimpan hasilnya.
 - Mengelola file media, dependensi, dan konfigurasi lainnya.
 - File terkait : training.py, model.pkl, requirements.txt, *.mp4
2. Qaisya Dwi Aryana:
 - Membuat dan mengelola logika utama deteksi mood real-time.
 - Menangani penggantian latar belakang berdasarkan mood pengguna.
 - File terkait : main.py, README.md
3. Marsella Yesi Natalia Sinaga:
 - Membuat dataset untuk pelatihan model.
 - Menulis skrip pengumpulan data dari kamera.
 - File terkait : datagen.py, data.csv, README.md

2.2 Proses Proyek

1. Analisis kebutuhan dan desain sistem.
2. Pengumpulan dataset yang berisi gambar ekspresi wajah.
3. Pelatihan model pembelajaran mesin untuk klasifikasi ekspresi.
4. Implementasi deteksi wajah real-time dengan MediaPipe dan OpenCV.
5. Integrasi efek visual berdasarkan hasil klasifikasi

2.3 Teknologi yang digunakan

1. Pustaka Utama
 - OpenCV: Pemrosesan gambar dan video.
 - MediaPipe: Deteksi landmark wajah dan segmentasi.
 - scikit-learn: Untuk melatih model deteksi ekspresi wajah.
2. Bahasa Pemrograman: Python
3. Pipeline Model:
 - Pipeline Model: Preprocessing menggunakan StandardScaler.
 - Model menggunakan LogisticRegression dari scikit-learn.

3 Penjelasan Kode Program

3.1 Inisialisasi dan Persiapan

```
1 import cvzone
2 import pickle
3 from cvzone.FaceMeshModule import FaceMeshDetector
4 import cv2
5 import numpy as np
6 import mediapipe as mp
7
8 # Initialize webcam
9 cap = cv2.VideoCapture(0)
10 if not cap.isOpened():
11     print("Error: Unable to access the webcam.")
12     exit(1)
13
14 # Load face mesh detector
15 FMD = FaceMeshDetector()
16
17 # Load the behavior prediction model
18 try:
19     with open('model.pkl', 'rb') as f:
20         Behaviour_model = pickle.load(f)
21 except FileNotFoundError:
22     print("Error: Behavior prediction model file 'model.pkl' not found.")
23     exit(1)
24
25 # Initialize mediapipe segmentation
26 mp_selfie_segmentation = mp.solutions.selfie_segmentation
27 segment = mp_selfie_segmentation.SelfieSegmentation(model_selection=1)
28
29
```

Kode 1: Inisialisasi dan Persiapan

Penjelasan:

- Webcam digunakan untuk menangkap gambar secara real-time.
- FaceMeshDetector mendeteksi fitur wajah pengguna.
- Model prediksi perilaku memanfaatkan file model.pkl untuk mendeteksi mood berdasarkan titik wajah.
- mediapipe digunakan untuk segmentasi latar belakang.

3.2 Segmentasi Latar Belakang dan Penggantian

```
1 # Use mediapipe for segmentation
2 frame_rgb = cv2.cvtColor(frame, cv2.COLOR_BGR2RGB)
3 results = segment.process(frame_rgb)
4
5 # Create a mask for the background
6 mask = results.segmentation_mask
7 condition = mask > 0.5 # Threshold to distinguish background and foreground
8
9 # Replace the background
10 output_frame = np.where(condition[..., None], frame, bg_frame)
11
```

12

Kode 2: Segmentasi Latar Belakang dan Penggantian

Penjelasan:

- Frame diubah menjadi format RGB untuk segmentasi.
- Mask digunakan untuk membedakan area latar belakang dan objek utama.
- Latar belakang diganti dengan video tertentu sesuai mood.

3.3 Prediksi Mood

```
1  # Face Mesh Processing
2  img, faces = FMD.findFaceMesh(output_frame)
3  cvzone.putTextRect(output_frame, 'Mood', (10, 80))
4  if faces:
5      face = faces[0]
6      face_data = list(np.array(face).flatten())
7
8      try:
9          # Predict behavior
10         result = Behaviour_model.predict([face_data])
11         mood = result[0]
12
13
```

Kode 3: Prediksi Mood

Penjelasan:

- Data titik wajah diambil dan diflatkan menjadi array.
- Model prediksi digunakan untuk menentukan mood seperti "happy", "sad", atau "shock".

3.4 Penggantian Video Berdasarkan Mood

```
1  # Change background video if mood changes
2  if mood != current_mood:
3      if mood in background_video_paths:
4          print(f"Detected mood change: {current_mood} -> {mood}")
5          current_mood = mood
6          background_video.release() # Release the current video
7          background_video = initialize_video(background_video_paths[current_mood]) #
8  Reinitialize
9
```

Kode 4: Penggantian Video Berdasarkan Mood

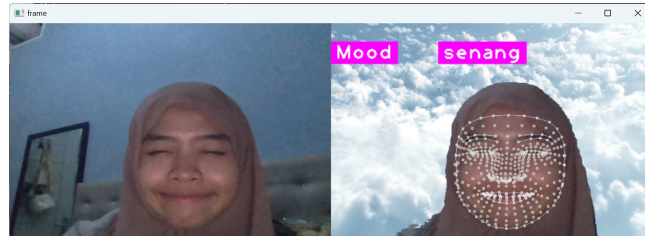
Penjelasan:

- Jika mood yang terdeteksi berbeda dari sebelumnya, video latar belakang diganti.
- Video lama dilepaskan, dan video baru diinisialisasi ulang.

4 Hasil

Proyek berhasil mengembangkan aplikasi yang mampu mendeteksi ekspresi wajah pengguna secara real-time dan menampilkan efek visual interaktif yang sesuai. Berikut adalah beberapa hasil yang dicapai:

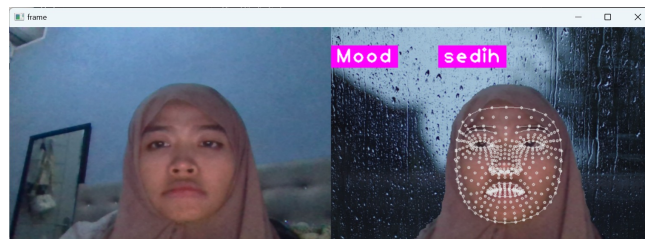
1. Deteksi ekspresi senyum menampilkan animasi awan.



Gambar 1: Gambar 3.1 Hasil Filter Senang

Gambar 3.1 diatas merupakan hasil outputan ketika sistem mendeteksi ekspresi senang, Ekspresi senang terjadi ketika ujung bibir naik dan mata menyempit

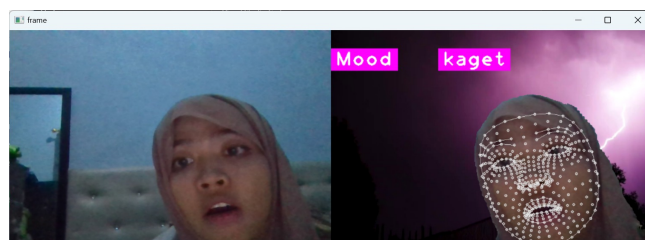
2. Deteksi ekspresi sedih menampilkan efek hujan.



Gambar 2: Gambar 3.2 Hasil Filter Sedih

Gambar 3.2 di atas merupakan hasil outputan ketika sistem mendeteksi ekspresi sedih. Ekspresi sedih terjadi ketika ujung bibir turun dan mata terlihat sayu.

3. Deteksi ekspresi kaget menampilkan efek petir.



Gambar 3: Gambar 3.3 Hasil Filter Kaget

Gambar 3.3 di atas merupakan hasil outputan ketika sistem mendeteksi ekspresi kaget. Ekspresi kaget terjadi ketika mata terbuka lebar dan alis terangkat.

5 Analisis dan Pembahasan

Beberapa tantangan yang dihadapi meliputi:

- Kesulitan mendeteksi ekspresi tertentu akibat variasi pencahayaan dan sudut kamera.

- Performa yang menurun pada perangkat keras dengan spesifikasi rendah.

Untuk mengatasi tantangan ini, optimisasi dataset dan algoritma pemrosesan dilakukan sehingga performa aplikasi meningkat.

6 Kesimpulan

Proyek Real-Time Mood Scanner berhasil mengembangkan filter interaktif yang mendeteksi ekspresi wajah pengguna dan menampilkan efek visual yang sesuai secara real-time. Meskipun demikian, terdapat ruang untuk perbaikan lebih lanjut dalam hal akurasi deteksi dan responsivitas sistem.

Proyek ini menunjukkan potensi besar dalam aplikasi analisis ekspresi wajah, baik untuk hiburan, penelitian, maupun pengembangan teknologi yang lebih canggih. Langkah selanjutnya meliputi peningkatan model pembelajaran mesin yang digunakan dan eksplorasi fitur tambahan, seperti analisis sentimen suara.

7 Cara Instalansi dan Penggunaan

7.1 Langkah Instalasi

Berikut revisi langkah instalasi dengan penambahan tautan untuk unduh data pada nomor 3:

1. Kloning Repositori

```
1 git clone https://github.com/qaisyadwi/RealTimeMoodScanner_MUL.git
```

2. (Opsional) Buat Virtual Environment

Langkah ini direkomendasikan untuk menjaga dependensi proyek tetap terisolasi.

```
1 python -m venv env
2 source env/bin/activate # Untuk Linux/macOS
3 env\Scripts\activate    # Untuk Windows
```

3. Unduh Background dan Siapkan File Video

- Unduh background dan file video dari tautan berikut: [Unduh data](#)
- Masukkan file di folder data ke folder RealTimeMoodScanner

4. Instal Dependensi Instal semua pustaka yang diperlukan:

```
1 pip install -r requirements.txt
```

5. Jalankan datagen.py untuk Persiapan Dataset

Skip datagen.py digunakan untuk mempersiapkan dataset yang akan digunakan dalam pelatihan model. Pastikan file dataset yang relevan tersedia atau diatur sesuai kebutuhan. Jalankan perintah:

```
1 python datagen.py
```

6. Jalankan training.py untuk Melatih Model

Skip training.py akan digunakan untuk melatih model berdasarkan dataset yang telah disiapkan. Jalankan perintah:

```
1 python training.py
```

7. Jalankan main.py untuk Menjalankan Aplikasi

Skip main.py berfungsi sebagai program utama untuk menjalankan aplikasi Real-Time Mood Scanner. Jalankan perintah:

```
1 python main.py
```

7.2 Cara Penggunaan

Setelah instalasi selesai, berikut cara menggunakan Real-Time Mood Scanner:

1. Buka Terminal Arahkan terminal ke direktori proyek, jika belum berada di sana:

```
1 cd RealTimeMoodScanner_MUL
```

2. Aktifkan Virtual Environment (Jika Digunakan)

```
1 source env/bin/activate # Untuk Linux/MacOS
2 env\Scripts\activate    # Untuk Windows
```

3. Jalankan Aplikasi Jalankan skrip utama untuk memulai aplikasi:

```
1 python main.py
```

4. Antarmuka Aplikasi

- Aplikasi akan menampilkan antarmuka kamera real-time.
- Pastikan kamera perangkat berfungsi.

5. Deteksi Mood

- Arahkan wajah ke kamera, dan aplikasi akan memindai ekspresi wajah.
- Sistem akan menampilkan hasil prediksi mood seperti senang, sedih, atau kaget.