



Anggota 1: **Gabriel Fico Darius (121140069)**

Tugas Besar

Anggota 2: **Muhammad Atha Akbar (121140123)**

Tanggal: **11-22-2024**

Anggota 3: **Muhamad Rizzky Nopriansyah (121140132)**

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

Filter Gaya Foto Random

1 Pendahuluan

Human Interface Design (HID) adalah disiplin yang fokus pada perancangan antarmuka pengguna (user interface) yang intuitif, efisien, dan menyenangkan [1]. Tujuan utama dari HID adalah untuk menciptakan pengalaman pengguna yang optimal dengan memadukan kebutuhan pengguna, fungsionalitas sistem, dan prinsip desain yang estetis [2]. Dalam dunia yang semakin terhubung dan digital ini, desain antarmuka yang baik menjadi sangat penting karena dapat memengaruhi kenyamanan, produktivitas, dan kepuasan pengguna saat berinteraksi dengan perangkat atau aplikasi [3].

2 Alat dan Cara Kerja

Dengan bantuan teknologi pose detection dari **Mediapipe**, aplikasi ini dapat melacak pergerakan tubuh pengguna dan membandingkannya dengan pose target untuk memastikan kesesuaian. Jika gaya yang dilakukan pengguna sesuai dengan pose target, aplikasi akan memberikan umpan balik berupa notifikasi "Match!" di layar.

3 Penjelasan Kode Program

3.1 Import Library

Library yang digunakan dalam program ini adalah OpenCV, MediaPipe, Pickle, OS, Random, Time, dan Feature_Extractor.

```
1 import cv2
2 import mediapipe as mp
3 import pickle
4 import os
5 import random
6 import time
7 import feature_extractor
8
```

Listing 3.1: Library yang digunakan

Penjelasan:

- OpenCV: untuk pemrosesan gambar dan video.

- MediaPipe : library Google yang digunakan untuk pengolahan multimodal (seperti pengolahan citra dan video).
- Pickle: library Python yang digunakan untuk serialisasi dan deserialisasi objek.
- OS: library Python yang menyediakan berbagai fungsi untuk berinteraksi dengan sistem file dan operasi sistem.
- Random: library untuk menghasilkan angka acak.
- Time: untuk mengakses informasi waktu, seperti mengukur durasi eksekusi kode, menunda proses, atau mendapatkan waktu saat ini.
- Feature_Extractor: library eksternal atau file Python yang kemungkinan berisi kode untuk mengekstraksi fitur dari gambar atau video.

3.2 Inisialisasi MediaPipe

```
1 mp_face_mesh = mp.solutions.face_mesh
2 face_mesh = mp_face_mesh.FaceMesh(min_detection_confidence=0.5, min_tracking_confidence=0.5)
3
```

Listing 3.2: Inisialisasi MediaPipe

Penjelasan:

- Menggunakan MediaPipe untuk mendeteksi landmark wajah.
- FaceMesh akan mendeteksi hingga 468 titik wajah pada gambar/video dengan tingkat kepercayaan minimal untuk deteksi dan pelacakan yang ditetapkan pada 0.5.

3.3 Fungsi untuk Menghitung Jarak Euclidean

```
1 def euclidean_distance(p1, p2):
2     return ((p2[0] - p1[0]) ** 2 + (p2[1] - p1[1]) ** 2) ** 0.5
3
```

Listing 3.3: Fungsi untuk Menghitung Jarak Euclidean

Penjelasan:

- Fungsi ini menghitung jarak Euclidean antara dua titik (p1, p2) dalam ruang dua dimensi (koordinat x, y). Jarak ini digunakan untuk menghitung ukuran fitur wajah seperti jarak antara mata atau mulut.

3.4 Fungsi untuk Membandingkan Fitur

```
1 def compare_features(user_features, target_features):
2     threshold = 0.03
3     for user_feature in user_features:
4         for key in user_feature:
5             if abs(user_feature[key] - target_features[key]) > threshold:
6                 return False
7     return True
8
```

Listing 3.4: Fungsi untuk Membandingkan Fitur

Penjelasan:

- Fungsi ini membandingkan fitur wajah pengguna dengan fitur target yang telah disimpan dalam file pickle.
- Fitur dihitung berdasarkan jarak Euclidean antara beberapa titik wajah, seperti mata dan mulut.
- Jika perbedaan antara fitur pengguna dan target lebih besar dari 0.03, maka hasilnya dianggap tidak cocok (match).

3.5 Fungsi untuk Merandom Gambar

```
1 def randomize_image(folder_path):
2     image_files = [f for f in os.listdir(folder_path) if f.endswith(('jpg', 'jpeg', 'png'))
3 ]
4     if not image_files:
5         print("Folder tidak mengandung file gambar.")
6         return None
7     selected_image = random.choice(image_files)
8     return os.path.join(folder_path, selected_image)
```

Listing 3.5: Fungsi untuk Merandom Gambar

Penjelasan:

- Fungsi ini memilih gambar secara acak dari folder yang diberikan (folder_path) dengan ekstensi .jpg, .jpeg, atau .png.
- Jika tidak ada gambar dalam folder, akan menampilkan pesan kesalahan.

3.6 Membaca File Fitur Target

```
1 if not os.path.exists("features.pkl"):
2     print("File features.pkl tidak ditemukan. Jalankan pickle.py terlebih dahulu untuk
3     menyimpan fitur target.")
4     return
5
6 with open("features.pkl", "rb") as f:
7     target_features = pickle.load(f)
```

Listing 3.6: Membaca File Fitur Target

Penjelasan:

- File features.pkl memuat fitur wajah target yang sebelumnya telah disimpan menggunakan pickle.
- Jika file ini tidak ditemukan, pengguna diminta untuk menjalankan proses penyimpanan fitur terlebih dahulu.

3.7 Inisialisasi Kamera dan Folder Gambar

```
1 target_folder = os.path.join(os.getcwd(), 'TUBES-MULTIMEDIA-IF4021', 'poses')
2 cap = cv2.VideoCapture(0)
3 if not cap.isOpened():
4     print("Gagal membuka kamera!")
5     return
6
```

Listing 3.7: Inisialisasi Kamera dan Folder Gambar

Penjelasan:

- Menginisialisasi folder yang berisi gambar pose (poses) dan membuka kamera untuk mendapatkan video stream.
- Jika kamera gagal dibuka, program akan berhenti.

3.8 Animasi Acak Gambar selama 5 Detik Pertama

```
1 animation_start_time = time.time()
2 animation_duration = 5 # Durasi animasi (detik)
3
```

Listing 3.8: Animasi Acak Gambar selama 5 Detik Pertama

Penjelasan:

- `random_image_path` akan dipilih secara acak setiap frame untuk animasi gambar yang ditampilkan selama 5 detik pertama setelah kamera dibuka.
- Setelah 5 detik, gambar terakhir yang dipilih akan dipertahankan.

3.9 Proses Frame Kamera

```
1 ret, frame = cap.read()
2 if not ret:
3     break
4 frame = cv2.flip(frame, 1)
5 frame_rgb = cv2.cvtColor(frame, cv2.COLOR_BGR2RGB)
6
```

Listing 3.9: Proses Frame Kamera

Penjelasan:

- Frame video dibaca dari kamera dan dibalik horizontal menggunakan `cv2.flip` untuk menciptakan efek mirror (menyerupai cermin).
- Frame kemudian diubah dari format BGR ke RGB (karena MediaPipe memerlukan format RGB).

3.10 Proses Face Mesh pada Pengguna

```
1 face_results = face_mesh.process(frame_rgb)
2
```

Listing 3.10: Proses Face Mesh pada Pengguna

Penjelasan:

- Menggunakan FaceMesh dari MediaPipe untuk mendeteksi landmark wajah pada frame yang diterima.
- Setiap wajah yang terdeteksi akan diproses dan fitur seperti jarak antara mata dan mulut dihitung menggunakan fungsi `euclidean_distance`.

3.11 Membandingkan Fitur Pengguna dengan Target

```
1     if user_features and selected_image_path:
2         match = compare_features(user_features, target_features[0])
3         if match:
4             cv2.putText(frame, "Pose Matched!", (50, 100), cv2.FONT_HERSHEY_SIMPLEX, 1, (0,
255, 0), 2)
5         else:
6             cv2.putText(frame, "Pose Not Matched!", (50, 100), cv2.FONT_HERSHEY_SIMPLEX, 1, (0,
0, 255), 2)
7
```

Listing 3.11: Membandingkan Fitur Pengguna dengan Target

Penjelasan:

- Fitur wajah pengguna dibandingkan dengan fitur target yang telah dimuat sebelumnya.
- Jika fitur cocok, pesan "Pose Matched!" ditampilkan, jika tidak cocok, pesan "Pose Not Matched!" akan ditampilkan.

3.12 Menampilkan Gambar Target pada Frame

```
1     if target_image is not None:
2         h, w, _ = frame.shape
3         frame[0:200, w-200:w] = target_image_resized
4
```

Listing 3.12: Menampilkan Gambar Target pada Frame

Penjelasan:

- Gambar target yang dipilih sebelumnya akan ditempelkan di pojok kanan atas dari frame kamera.

3.13 Menampilkan Frame Kamera

```
1     cv2.imshow("Pose Matching", frame)
2
```

Listing 3.13: Menampilkan Frame Kamera

Penjelasan:

- Menampilkan hasil akhir dari frame kamera beserta gambar target yang telah ditempelkan.

3.14 Menutup Program

```
1     if cv2.waitKey(1) & 0xFF == ord('q'):
2         break
3     cap.release()
4     cv2.destroyAllWindows()
5
```

Listing 3.14: Menutup Program

Penjelasan:

- Program akan terus berjalan hingga pengguna menekan tombol 'q' untuk keluar.
- Setelah itu, kamera dan semua jendela OpenCV akan ditutup.

3.15 Eksekusi Program

```
1  if __name__ == "__main__":  
2      main()  
3
```

Listing 3.15: Eksekusi Program

4 Hasil dan Pembahasan

4.1 Hasil

Program menggunakan deteksi wajah dan fitur wajah untuk membandingkan pose pengguna dengan pose yang sudah ditentukan sebelumnya. Gambar acak ditampilkan pada awalnya selama 5 detik, dan setelah itu, sistem akan mengevaluasi kecocokan pose pengguna dengan gambar target berdasarkan perbandingan fitur wajah yang telah diekstraksi.



Figure 4.1: Final Matched



Figure 4.2: Final Not Matched

Program dapat mendeteksi pose yang ditampilkan dari kamera realtime sudah sesuai dengan pose yang ditentukan atau tidak.

4.2 Pembahasan

Program yang telah Anda berikan adalah aplikasi berbasis OpenCV dan Mediapipe yang digunakan untuk mencocokkan pose wajah pengguna dengan gambar acak yang ditampilkan selama 5 detik. Program ini mengandalkan analisis fitur wajah, seperti jarak antara titik-titik tertentu pada wajah (misalnya mata dan mulut), untuk membandingkan pose pengguna dengan gambar yang sudah ada di dalam folder tertentu.

5 Cara Instalasi dan Penggunaan

5.1 Cara Instalasi

Clone Repository ini

```
1 git clone https://github.com/Mrnopri/TUBES-MULTIMEDIA-IF4021.git
2
```

Listing 5.1: Clone Repository

Install Requirement

```
1 pip install -r requirements.txt
2
```

Listing 5.2: Install Requirement

5.2 Cara Penggunaan

Jalankan ekstraksi fitur

```
1 python feature_extractor.py
2
```

Listing 5.3: Jalankan Ektrai Fitur

Jalankan program

```
1 python main.py
2
```

Listing 5.4: Jalankan Program

- Arahkan muka ke kamera dan samakan pose yang sudah disediakan dengan muka anda.
- Hasil kesesuaian akan ditampilkan di layar secara realtime.

References

- [1] Harmayani and P. Simamora, “Interaksi manusia dan komputer dalam ilmu filsafat,” *Jurnal Deli Sains Informatika*, vol. 2, no. 2, pp. 1–5, 2023.
- [2] L. H. Annisa and Y. H. C. Pratama, “Implementasi paradigma interaksi manusia komputer pada di era society 5.0: Systematic literature review,” *Technology and Informatics Insight Journal*, vol. 1, no. 2, pp. 8–98, 2022.
- [3] N. Kamilia and I. P. Wardhani, “Penerapan sistem interaksi manusia dan komputer pada website di stmik jakarta stik,” *Jurnal Ilmiah SIKOMTEK*, vol. 14, no. 1, pp. 27–36, 2024.