



Nama Anggota dan NIM:

- Bani Adam Tampubolon (121140187)
- Zahra Areefa Ananta (121140138)

Tugas Ke: UAS

Mata Kuliah: Multimedia (IF4021)

Tanggal: 18 desember 2024

1 Deskripsi Game

Proyek ini adalah sebuah filter kuis interaktif yang menggunakan deteksi gerakan kepala untuk memilih jawaban, dikembangkan dengan Python menggunakan pustaka OpenCV dan MediaPipe. Filter ini menampilkan pertanyaan secara acak dengan dua pilihan jawaban, di mana pengguna dapat menjawab dengan menggerakkan kepala ke kanan atau ke kiri. Setelah setiap jawaban diberikan, pertanyaan secara otomatis berganti hingga 5 pertanyaan selesai dijawab, dan skor akhir akan ditampilkan. Proyek ini dirancang untuk memberikan pengalaman interaktif tanpa memerlukan perangkat keras tambahan seperti mouse atau keyboard, sehingga cocok digunakan untuk hiburan, pembelajaran, atau aplikasi berbasis teknologi lainnya.

2 Kode Program

1. Library

```
import cv2
import mediapipe as mp
import numpy as np
import time
import random
import os
```

Library OpenCV, MediaPipe, NumPy, time, random, dan os digunakan untuk mendukung pengembangan aplikasi deteksi gerakan dan pengelolaan kuis secara interaktif. OpenCV digunakan untuk menangkap video dari kamera, memproses gambar secara real-time, dan menampilkan antarmuka kuis. MediaPipe menyediakan deteksi dan pelacakan landmark wajah yang membantu menganalisis gerakan kepala. NumPy digunakan untuk manipulasi data numerik, seperti perhitungan koordinat atau sudut. time membantu dalam fungsi terkait waktu, seperti mengatur jeda atau mencatat durasi. random digunakan untuk memilih pertanyaan secara acak, sehingga kuis lebih dinamis. Terakhir, os digunakan untuk operasi sistem seperti pengelolaan file atau path jika aplikasi membutuhkan akses ke sumber daya tambahan. Kombinasi ini memungkinkan aplikasi berfungsi secara efisien dan interaktif.



Nama Anggota dan NIM:

- Bani Adam Tampubolon (121140187)
- Zahra Areefa Ananta (121140138)

Tugas Ke: UAS

Mata Kuliah: Multimedia (IF4021)

Tanggal: 18 desember 2024

2. Fungsi mendeteksi kemiringan kepala

```
# Fungsi untuk mendeteksi kemiringan kepala
def detect_head_movement_with_angle(left_eye, right_eye):
    left_eye_center = (left_eye[0], left_eye[1])
    right_eye_center = (right_eye[0], right_eye[1])

    delta_x = right_eye_center[0] - left_eye_center[0]
    delta_y = right_eye_center[1] - left_eye_center[1]

    # Hitung sudut menggunakan formula kemiringan
    angle = np.arctan2(delta_y, delta_x)
    angle = np.degrees(angle)

    # Threshold untuk mendeteksi kemiringan
    if angle > 10:
        return "Right", angle
    elif angle < -10:
        return "Left", angle
    else:
        return "Straight", angle
```

Dalam fungsi diatas, NumPy digunakan untuk melakukan perhitungan matematis yang melibatkan koordinat mata pengguna. `np.arctan2` menghitung sudut kemiringan antara dua koordinat mata, sementara `np.degrees` mengonversi hasil sudut dari radian ke derajat agar lebih mudah diinterpretasikan. Fungsi ini menganalisis posisi pusat mata kiri dan kanan yang diperoleh sebelumnya, menghitung sudut di antara keduanya, dan menentukan arah kemiringan kepala (ke kanan, ke kiri, atau lurus) berdasarkan nilai threshold tertentu. Tanpa NumPy, proses manipulasi data numerik seperti ini menjadi jauh lebih rumit dan kurang efisien.

3. Menambahkan gambar



Nama Anggota dan NIM:

- Bani Adam Tampubolon (121140187)
- Zahra Areefa Ananta (121140138)

Tugas Ke: UAS

Mata Kuliah: Multimedia (IF4021)

Tanggal: 18 desember 2024

```
def overlay_image_with_alpha(frame, img, x1, y1, x2, y2):  
    if img.shape[2] == 4: # Gambar dengan transparansi  
        alpha_channel = img[:, :, 3] # Saluran alfa  
        rgb_channels = img[:, :, :3] # Saluran RGB  
  
        for c in range(3): # Untuk setiap saluran warna  
            frame[y1:y2, x1:x2, c] = frame[y1:y2, x1:x2, c] * (1 - alpha_channel / 255.0) + \  
                rgb_channels[:, :, c] * (alpha_channel / 255.0)  
    else:  
        # Jika gambar tidak transparan, langsung tempel  
        frame[y1:y2, x1:x2] = img
```

Fungsi `overlay_image_with_alpha` digunakan untuk menambahkan gambar (*image*) ke dalam frame utama (*frame*) dengan mendukung transparansi, sehingga bagian gambar yang transparan tidak menutupi elemen di belakangnya. Jika gambar memiliki saluran alfa (RGBA), transparansi diterapkan menggunakan kombinasi warna dari frame utama dan gambar berdasarkan tingkat alfa. Perulangan pada setiap saluran warna (R, G, B) memastikan penggabungan transparan dilakukan secara akurat. Jika gambar tidak memiliki saluran alfa, fungsi langsung menempelkan gambar ke posisi yang ditentukan di dalam frame. Fungsi ini cocok untuk menciptakan elemen antarmuka interaktif, seperti menambahkan ikon atau stiker transparan ke video real-time.

4. Skor

```
def show_score_image(score):  
    # Tentukan path untuk gambar skor  
    score_images_path = "hasil"  
    score_image_file = f"{score_images_path}/skor{score}.png"  
  
    # Cek apakah gambar ada  
    if os.path.exists(score_image_file):  
        score_img = cv2.imread(score_image_file)  
        if score_img is not None:  
            cv2.imshow("Your Score", score_img)  
            cv2.waitKey(0) # Tunggu hingga pengguna menekan tombol apapun  
            cv2.destroyAllWindows()
```

Fungsi `show_score_image` digunakan untuk menampilkan gambar skor berdasarkan hasil yang dicapai pengguna setelah kuis selesai. Fungsi ini mencari file gambar skor di dalam folder



Nama Anggota dan NIM:

- Bani Adam Tampubolon (121140187)
- Zahra Areefa Ananta (121140138)

Tugas Ke: UAS

Mata Kuliah: Multimedia (IF4021)

Tanggal: 18 desember 2024

bernama "hasil" dengan format nama file seperti skor{score}.png. Jika gambar ditemukan dan berhasil dimuat menggunakan cv2.imread, gambar ditampilkan di jendela OpenCV dengan judul "Your Score". Fungsi cv2.waitKey(0) memastikan jendela tetap terbuka sampai pengguna menekan tombol, dan cv2.destroyAllWindows menutup semua jendela OpenCV setelahnya. Fungsi ini berguna untuk memberikan umpan balik visual berupa skor yang menarik dan interaktif kepada pengguna.

5. Fungsi main

```
def main():
    cap = cv2.VideoCapture(0)
    if not cap.isOpened():
        print("Kamera tidak dapat diakses.")
        return

    print("Mulai mendeteksi kemiringan kepala...")

    # Path ke folder gambar pertanyaan
    question_images_path = "quiz-question"
    question_images = [f"{question_images_path}/q{i}.png" for i in range(1, 6)]

    current_question = 0
    score = 0
    prev_state = "Straight"
    question_start_time = time.time()
    is_shuffling = False
    shuffle_start_time = None
    selected_question = None
```

Fungsi main adalah fungsi utama yang menjalankan aplikasi kuis berbasis deteksi gerakan kepala. Fungsi ini memulai video capture menggunakan kamera perangkat dengan cv2.VideoCapture(0), memeriksa aksesibilitas kamera, dan memberi notifikasi jika kamera tidak dapat digunakan. Fungsi ini juga memuat jalur file untuk gambar-gambar pertanyaan kuis yang berada dalam folder "quiz-question" dengan nama file seperti q1.png hingga q5.png. Variabel current_question melacak indeks pertanyaan saat ini, score mencatat skor pengguna, prev_state menyimpan status kemiringan kepala sebelumnya (misalnya, "Straight"), sementara question_start_time mencatat waktu mulai pertanyaan. Variabel tambahan seperti is_shuffling, shuffle_start_time, dan selected_question digunakan untuk mengelola logika pemilihan dan penampilan pertanyaan. Secara keseluruhan, fungsi ini menyiapkan elemen-elemen inti untuk menjalankan kuis secara interaktif dengan memanfaatkan kamera dan logika gerakan kepala.

6. Loop



Nama Anggota dan NIM:

- Bani Adam Tampubolon (121140187)
- Zahra Areefa Ananta (121140138)

Tugas Ke: UAS

Mata Kuliah: Multimedia (IF4021)

Tanggal: 18 desember 2024

```
while cap.isOpened():
    ret, frame = cap.read()
    if not ret:
        print("Tidak dapat membaca frame dari kamera.")
        break

    frame = cv2.flip(frame, 1)
    rgb_frame = cv2.cvtColor(frame, cv2.COLOR_BGR2RGB)
    results = face_mesh.process(rgb_frame)
```

Kode ini menjalankan loop utama untuk memproses video real-time dari kamera, membaca frame satu per satu menggunakan `cap.read()` selama kamera masih terbuka (`cap.isOpened()`). Jika frame tidak dapat dibaca, proses dihentikan dengan pesan kesalahan. Fungsi `cv2.flip(frame, 1)` membalik frame secara horizontal untuk memberikan efek mirroring, sehingga lebih intuitif bagi pengguna. Frame yang telah dimodifikasi kemudian diubah dari format BGR ke RGB menggunakan `cv2.cvtColor(frame, cv2.COLOR_BGR2RGB)` karena pustaka MediaPipe memproses data gambar dalam format RGB. Frame RGB ini selanjutnya diproses oleh model MediaPipe `face_mesh.process()` untuk mendeteksi dan melacak titik-titik pada wajah. Bagian ini berfungsi sebagai inti dari proses pengambilan data wajah untuk analisis lebih lanjut, seperti mendeteksi gerakan kepala atau ekspresi wajah.

7. Logika utama

```
if results.multi_face_landmarks:
    for face_landmarks in results.multi_face_landmarks:
        # Landmark untuk mata kiri dan mata kanan
        left_eye_landmark = face_landmarks.landmark[159]
        right_eye_landmark = face_landmarks.landmark[386]

        # Landmark untuk hidung sebagai referensi area rambut
        nose_landmark = face_landmarks.landmark[1]

        # Koordinat mata dan hidung dalam piksel
        h, w, _ = frame.shape
        left_eye = (int(left_eye_landmark.x * w), int(left_eye_landmark.y * h))
        right_eye = (int(right_eye_landmark.x * w), int(right_eye_landmark.y * h))
        nose_x, nose_y = int(nose_landmark.x * w), int((nose_landmark.y - 0.15) * h)
```



Nama Anggota dan NIM:

- Bani Adam Tampubolon (121140187)
- Zahra Areefa Ananta (121140138)

Tugas Ke: UAS

Mata Kuliah: Multimedia (IF4021)

Tanggal: 18 desember 2024

```
# Deteksi kemiringan berdasarkan mata
movement_detected, angle = detect_head_movement_with_angle(left_eye, right_eye)

if movement_detected == "Straight" and current_question < len(question_images):
    if not is_shuffling and selected_question is None:
        is_shuffling = True
        shuffle_start_time = time.time()

    if is_shuffling:
        if time.time() - shuffle_start_time < 2.0:
            shuffle_index = random.randint(0, len(question_images) - 1)
            shuffle_img = cv2.imread(question_images[shuffle_index], cv2.IMREAD_UNCHANGED)
            shuffle_img_resized = cv2.resize(shuffle_img, (300, 200))
            x1, y1 = nose_x - shuffle_img_resized.shape[1] // 2, nose_y - shuffle_img_resized.shape[0]
            x2, y2 = x1 + shuffle_img_resized.shape[1], y1 + shuffle_img_resized.shape[0]

            if x1 >= 0 and y1 >= 0 and x2 <= w and y2 <= h:
                overlay_image_with_alpha(frame, shuffle_img_resized, x1, y1, x2, y2)
        else:
            is_shuffling = False
            selected_question = question_images[current_question]

    if selected_question:
        question_img = cv2.imread(selected_question, cv2.IMREAD_UNCHANGED)
        question_img_resized = cv2.resize(question_img, (300, 200))
        x1, y1 = nose_x - question_img_resized.shape[1] // 2, nose_y - question_img_resized.shape[0]
        x2, y2 = x1 + question_img_resized.shape[1], y1 + question_img_resized.shape[0]

        if x1 >= 0 and y1 >= 0 and x2 <= w and y2 <= h:
            overlay_image_with_alpha(frame, question_img_resized, x1, y1, x2, y2)

if movement_detected != "Straight" and movement_detected != prev_state and selected_question:
    user_answer = "YES" if movement_detected == "Right" else "NO"

    correct_answers = ["NO", "NO", "YES", "YES", "NO"]
    if user_answer == correct_answers[current_question]:
        score += 1

    current_question += 1
```

Kode ini mengelola logika utama kuis berbasis deteksi gerakan kepala dengan menggunakan hasil landmark wajah dari MediaPipe. Setiap frame, titik-titik landmark untuk mata kiri, mata kanan, dan hidung pengguna diubah menjadi koordinat piksel untuk mendeteksi arah gerakan kepala menggunakan fungsi `detect_head_movement_with_angle`. Jika kepala dalam posisi "Straight" dan pertanyaan belum selesai, proses pengacakan gambar pertanyaan dimulai selama dua detik, dan gambar yang dipilih ditampilkan di dekat posisi hidung pengguna dengan bantuan `overlay_image_with_alpha`. Ketika pengguna menggerakkan kepala ke kanan atau kiri untuk menjawab (berdasarkan arah kemiringan yang terdeteksi), jawaban dicocokkan dengan daftar jawaban benar `correct_answers`, dan skor diperbarui jika sesuai. Setelah itu, kode melanjutkan ke pertanyaan berikutnya. Secara keseluruhan, bagian ini menangani pengacakan, penampilan pertanyaan, deteksi jawaban, serta pembaruan skor dan status kuis secara real-time.

8. Menangani tampilan skor akhir



Nama Anggota dan NIM:

- Bani Adam Tampubolon (121140187)
- Zahra Areefa Ananta (121140138)

Tugas Ke: UAS

Mata Kuliah: Multimedia (IF4021)

Tanggal: 18 desember 2024

```
# Menampilkan skor setelah semua pertanyaan dijawab
if current_question >= len(question_images):
    cv2.putText(frame, f"Final Score: {score}", (50, 200), cv2.FONT_HERSHEY_SIMPLEX, 1, (0, 255, 255), 2)
    if current_question == len(question_images):
        show_score_image(score) # Tampilkan gambar skor

cv2.imshow('Head Movement Quiz', frame)

if current_question >= len(question_images) or cv2.waitKey(1) & 0xFF == ord('q'):
    break

cap.release()
cv2.destroyAllWindows()

print(f"Quiz Selesai! Skor akhir Anda: {score}/{len(question_images)}")

if __name__ == "__main__":
    main()
```

Kode ini menangani tampilan skor akhir setelah semua pertanyaan dijawab dalam kuis berbasis deteksi gerakan kepala. Jika jumlah pertanyaan yang dijawab sudah mencapai total pertanyaan, teks skor akhir ditampilkan di layar menggunakan `cv2.putText`. Setelah itu, gambar skor akhir ditampilkan melalui fungsi `show_score_image`. Fungsi `cv2.imshow` digunakan untuk menampilkan frame yang telah diperbarui dengan skor pada jendela kuis. Proses ini berlanjut hingga semua pertanyaan dijawab atau pengguna menekan tombol 'q' untuk keluar. Setelah kuis selesai, kamera dilepaskan dengan `cap.release()` dan semua jendela OpenCV ditutup dengan `cv2.destroyAllWindows()`. Program ini juga mencetak skor akhir ke konsol. Secara keseluruhan, bagian ini mengelola tampilan hasil akhir kuis dan memastikan aplikasi ditutup dengan benar setelah selesai.



Nama Anggota dan NIM:

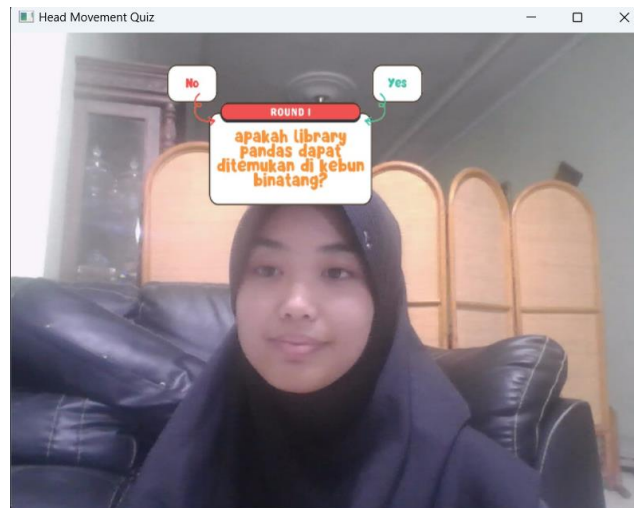
- Bani Adam Tampubolon (121140187)
- Zahra Areefa Ananta (121140138)

Mata Kuliah: Multimedia (IF4021)

Tugas Ke: UAS

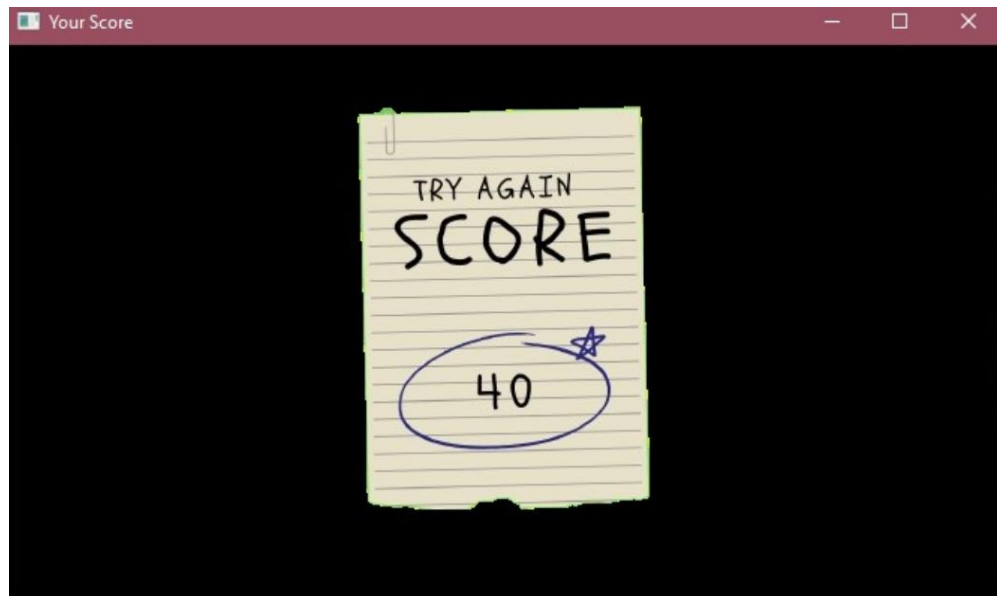
Tanggal: 18 desember 2024

9. Tampilan output



Gambar diatas adalah tampilan output ketika program dijalankan.

10. Tampilan skor



Gambar diatas adalah tampilan skor ketika quis habis.



Nama Anggota dan NIM:

- Bani Adam Tampubolon (121140187)
- Zahra Areefa Ananta (121140138)

Tugas Ke: UAS

Mata Kuliah: Multimedia (IF4021)

Tanggal: 18 desember 2024

References

[1] <https://vt.tiktok.com/ZS6YETtSJ/>