



Nama: **Dharma Rizki Geraldo, Intan Permata Sari, Yossi Afridho** Tugas : **Final Project**
Mata Kuliah: **Sistem/Teknologi Multimedia (IF4021)** Tanggal: **30 April 2025**

1 Deskripsi Proyek

Loud Scream adalah sebuah filter game interaktif berbasis Python yang memanfaatkan input suara mikrofon untuk mengontrol karakter dalam permainan. Pemain mengeluarkan suara keras untuk membuat karakter melompat melewati rintangan secara real-time. Game ini menggunakan OpenCV untuk rendering grafis, PyAudio untuk menangkap input suara, dan Pygame untuk musik latar, sehingga menghadirkan pengalaman bermain yang unik dan seru seperti konsep game “Scream Go Hero”.

2 Teknologi

Teknologi yang digunakan dalam Loud Scream adalah:

- **OpenCV (cv2)**
OpenCV digunakan sebagai kerangka utama untuk tampilan visual dan logika permainan 2D. OpenCV menjadi pilihan ideal karena fleksibel dalam manipulasi frame per frame dan kompatibel dengan webcam.
- **Numpy**
Digunakan sebagai alat bantu pemrosesan sinyal audio. NumPy diperlukan karena kemampuannya memproses array dengan sangat efisien dan cepat.
- **PyAudio**
PyAudio sangat penting dalam proyek ini karena kontrol utama permainan dilakukan melalui volume suara, bukan tombol fisik.
- **Pygame**
Digunakan untuk memutar musik latar belakang selama permainan berlangsung, dan memberikan nuansa audio yang mendukung atmosfer permainan. Meskipun seluruh grafik ditangani oleh OpenCV, penggunaan pygame hanya terbatas pada pengolahan suara, menjadikannya alat tambahan yang ringan namun efektif.

3 Cara Kerja

Permainan Loud Scream bekerja dengan menggabungkan pengolahan audio real-time, rendering visual, dan logika permainan berbasis platformer. Berikut adalah tahapan cara kerja permainan:

- **Inisialisasi**
Saat program dijalankan:
Kamera (melalui OpenCV) dan mikrofon (melalui PyAudio) diaktifkan. Sumber daya seperti gambar karakter dan platform dimuat. Musik latar dipersiapkan menggunakan pygame.mixer. Variabel status permainan (state) diinisialisasi.

- **Tampilan Layar Awal**
Sebelum permainan dimulai, layar akan menampilkan: Pesan "Click or Press SPACE to Start. Highscore terakhir Pengguna dapat memulai permainan dengan klik mouse atau tombol spasi.
- **Mulai Permainan**
Saat pengguna memulai: Permainan di-reset menggunakan fungsi `reset_game(...)`. Musik latar diputar. Platform awal dan posisi karakter disiapkan. Game memasuki mode loop utama.
- **Loop Utama Permainan**
Loop utama berjalan terus-menerus sampai pengguna keluar. Setiap iterasi melibatkan:
 - a. **Pengambilan Frame dan Audio**
 - Frame baru diambil dari webcam.
 - Data audio dibaca dari mikrofon.
 - Volume dihitung menggunakan NumPy untuk menentukan kekuatan lompatan.
 - b. **Logika Lompatan**
Jika volume melebihi ambang batas dan karakter tidak sedang melompat, karakter akan:
 - Melompat dengan kecepatan vertikal (`y_velocity`) yang proporsional terhadap volume suara.
 - Efek visual lompatan ditambahkan menggunakan warna dan animasi.
 - c. **Pergerakan Karakter dan Platform**
 - Karakter jatuh karena gravitasi jika tidak di atas platform.
 - Jika karakter menyentuh platform dari atas, dia akan *mendarat* dan skor bertambah.
 - Semua platform bergerak ke kiri seiring waktu untuk menciptakan ilusi pergerakan karakter ke depan.
 - Platform baru akan di-generate secara dinamis menggunakan `generate_ground_block(...)`.
 - d. **Deteksi Game Over**
 - Jika karakter jatuh melewati batas bawah layar, permainan masuk mode Game Over.
 - Musik berhenti.
 - Skor dibandingkan dengan highscore.
 - e. **Tampilan dan Efek Visual**
 - Seluruh tampilan diperbarui:
 - Gambar karakter ditampilkan dengan atau tanpa efek lompatan.
 - Platform ditampilkan dengan posisi terbaru.
 - Skor, volume, dan instruksi ditampilkan secara real-time.
- **Game Over**
 - Saat permainan berakhir:
 - Layar menampilkan "GAME OVER", skor, highscore, dan instruksi.
 - Pengguna bisa:
 - Menekan *spasi* untuk mulai ulang.
 - Menekan *ESC atau Q* untuk keluar.

4 Penjelasan Kode Program

Loud Scream memiliki 2 kode program (`main.py` dan `game_utils.py`) dan 1 jupyter notebook (`Loud-Scream.ipynb`). Berikut merupakan penjelasannya:

4.1 `main.py`

Merupakan program utama yang menangani:

- Inisialisasi kamera, mikrofon, dan sumber daya

```
cap = cv2.VideoCapture(0)
p = pyaudio.PyAudio()
pygame.mixer.init()
background_music = pygame.mixer.Sound("resources/background_music.mp3")
```

- Loop utama permainan

```
while True:
    ret, frame = cap.read()
```

- Deteksi suara dan kontrol karakter

```
data = stream.read(chunk)
audio_data = np.frombuffer(data, dtype=np.int16)
volume = np.mean(np.abs(audio_data))
```

4.2 game_utils.py

Berisikan fungsi yang membantu logika permainan dalam kode utama:

- `display_message(...)`
Menampilkan teks ke dalam frame di posisi tertentu.
- `show_game_over_screen(...)`
Menampilkan layar game over lengkap dengan skor dan leaderboard.
- `generate_ground_block(...)`
Membuat platform atau celah (gap) secara acak untuk menantang pemain:
 - Terdapat logika agar tidak terlalu banyak "gap" berurutan
 - Platform memiliki tinggi, lebar, dan jarak acak
- `reset_game(...)`
Mengatur ulang semua variabel game ke kondisi awal, termasuk:
 - Memasang platform awal
 - Menyetel ulang skor dan posisi karakter
 - Memulai permainan baru

4.3 LoudScream.ipynb

File ini berisikan penjelasan dan kegunaan setiap fungsi dari kode `main.py` dan `game_utils.py`

5 Hasil Analisis

- Fungsionalitas Program
Program dapat dijalankan sesuai dengan tujuan yang telah dirancang, yaitu sebagai permainan platformer yang dikendalikan dengan suara. Karakter dapat melompat saat pengguna berteriak atau mengeluarkan suara cukup keras, dan skor akan bertambah saat berhasil mendarat di platform.

- **Responsivitas Terhadap Input**
Deteksi suara berjalan secara real-time dan responsif. Penggunaan buffer dan jeda waktu (`jump_cooldown_ms`) mencegah terjadinya input berlebihan, sehingga kontrol lompatan terasa alami dan tidak terlalu sensitif.
- **Struktur Kode**
Kode dibagi ke dalam dua file utama: `main.py` sebagai logika utama permainan dan `game_utils.py` sebagai kumpulan fungsi pendukung. Pemisahan ini meningkatkan keterbacaan, kemudahan pemeliharaan, serta memungkinkan pengembangan lebih lanjut secara modular.
- **Penggunaan Teknologi**
Perpaduan antara OpenCV, PyAudio, NumPy, dan Pygame memberikan solusi lengkap yang mencakup visual, audio input, analisis data, dan output suara. Seluruh pustaka bekerja secara sinergis untuk menciptakan permainan yang interaktif dan imersif.
- **Tampilan Visual**
Permainan menyajikan elemen grafis seperti karakter, platform, skor, dan pesan teks dengan cukup menarik. Efek seperti bayangan, animasi skor, efek warna saat lompat, serta tampilan layar Game Over memberi nilai tambah pada aspek visual.
- **Pengelolaan Sumber Daya**
Gambar karakter dan platform serta file musik dimuat dari folder `resources/`, dan leaderboard disimpan dalam file teks. Ini menunjukkan pengelolaan resource yang sederhana namun efektif.
- **Kelebihan**
 - Inovasi kontrol dengan suara membuat permainan berbeda dari platformer pada umumnya.
 - Visualisasi cukup menarik dengan efek animasi dan warna.
 - Struktur kode bersih dan mudah dikembangkan.
 - Menggunakan pustaka open-source yang ringan dan powerful.
- **Kekurangan**
 - Ketergantungan pada perangkat keras (mikrofon dan kamera) membuatnya tidak optimal di semua perangkat.
 - Tidak ada kontrol alternatif selain suara.
 - Leaderboard belum otomatis menyimpan skor akhir.
 - Pengambilan video dan audio dilakukan dalam thread yang sama, berpotensi menghambat performa jika terjadi overload