



Nama: **Hafiz Amrullah (119140177)**  
Mata Kuliah: **Pemrosesan Sinyal Digital (IF3024)**

Tugas Ke: **UAS**  
Tanggal: 24 Desember 2024

## 1 Deskripsi Proyek

Proyek ini bertujuan untuk menggabungkan sistem pengukuran sinyal respirasi dan remote-photoplethysmography (rPPG) menggunakan Python. Program ini memanfaatkan video real-time dari webcam untuk mengekstrak dan memvisualisasikan kedua sinyal tersebut. Teknologi ini dapat digunakan dalam aplikasi kesehatan seperti pemantauan pernapasan dan detak jantung tanpa kontak.

## 2 Deskripsi Sinyal

### 2.1 Sinyal Respirasi

Sinyal respirasi adalah sinyal biologis yang mencerminkan aktivitas pernapasan seseorang. Sinyal ini dapat diekstraksi dari perubahan volume atau intensitas pada area tubuh tertentu yang bergerak saat bernapas, seperti dada atau perut. Dalam proyek ini, sinyal respirasi diperoleh dengan menganalisis perubahan intensitas piksel dari video real-time.

### 2.2 Sinyal rPPG

Remote photoplethysmography (rPPG) adalah metode non-invasif untuk mengukur variasi volume darah berdasarkan perubahan kecil dalam intensitas cahaya yang dipantulkan dari kulit. Teknik ini menggunakan channel hijau dari video wajah untuk mengekstrak sinyal detak jantung, karena channel hijau paling sensitif terhadap perubahan volume darah.

## 3 Instalasi

### 3.1 Persyaratan Sistem

- Python 3.7 atau lebih baru
- Webcam yang berfungsi

### 3.2 Langkah Instalasi

1. Clone repository: `git clone https://github.com/hafiz-dev1/final_project_signal.git`
2. Pindah ke direktori proyek: `cd final_project_signal`
3. (Opsional) Buat virtual environment:

```
python -m venv venv
source venv/bin/activate # Untuk Linux/MacOS
venv\Scripts\activate    # Untuk Windows
```

4. Instal dependencies:

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

## 4 Cara Penggunaan

1. Jalankan program utama:

```
python src/main.py
```

2. Program akan membuka feed webcam dan memproses frame secara real-time.
3. Tekan tombol **q** pada keyboard untuk menghentikan program.

## 5 Tools yang Digunakan

### 5.1 NumPy

NumPy adalah library Python yang digunakan untuk komputasi numerik. Dalam proyek ini, NumPy digunakan untuk memproses data piksel dari frame video dan melakukan operasi matematis seperti rata-rata dan normalisasi.

### 5.2 Matplotlib

Matplotlib adalah library untuk visualisasi data. Proyek ini menggunakan Matplotlib untuk menampilkan grafik sinyal respirasi dan rPPG secara real-time.

### 5.3 OpenCV-Python

OpenCV adalah library untuk pengolahan citra dan video. Dalam proyek ini, OpenCV digunakan untuk menangkap frame dari webcam, mengonversi frame ke grayscale, dan memproses data visual lainnya.

### 5.4 SciPy

SciPy adalah library untuk komputasi ilmiah. Dalam proyek ini, SciPy digunakan untuk operasi sinyal seperti detrending dan filtering pada data rPPG dan sinyal respirasi.

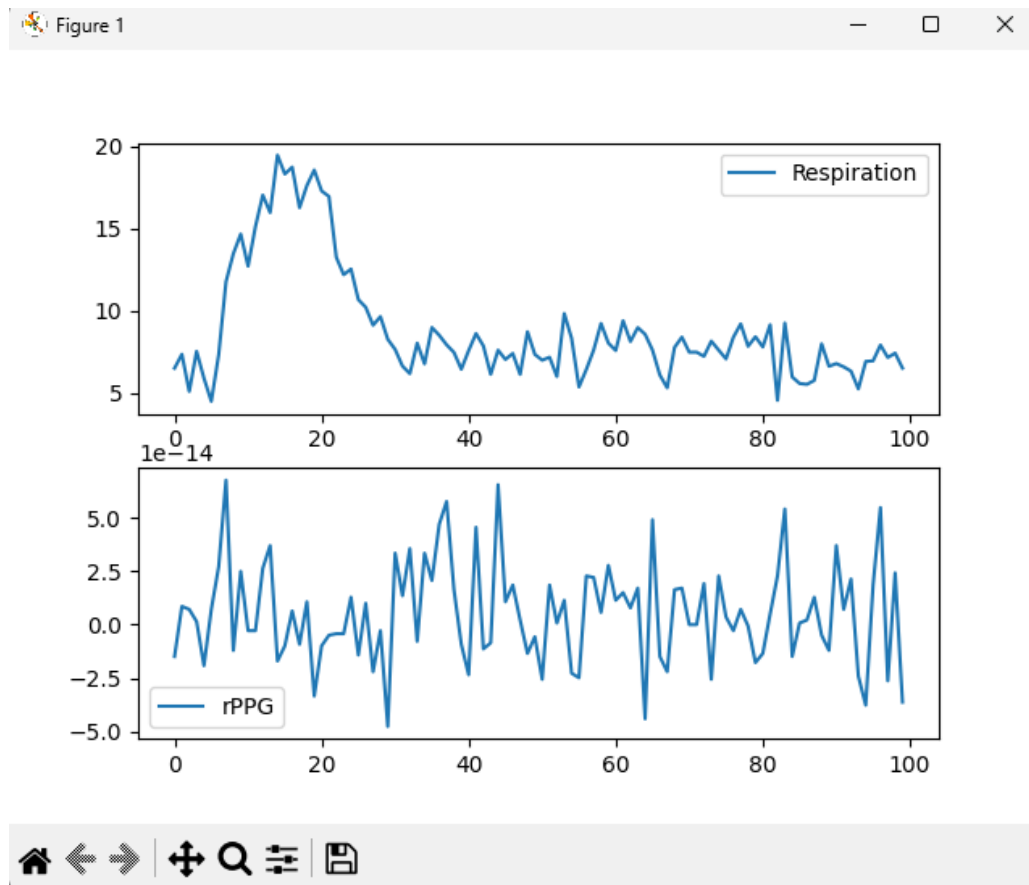
## 6 Hasil dan Visualisasi

## 7 Hasil dan Visualisasi

Visualisasi sinyal respirasi dan rPPG dapat dilihat pada Gambar 2. Gambar ini menunjukkan dua grafik yang memvisualisasikan sinyal yang diekstraksi secara real-time dari video webcam.

### 7.1 Visualisasi Sinyal Respirasi

Grafik pertama (di bagian atas) menunjukkan sinyal respirasi yang diperoleh dari perubahan intensitas piksel pada area dada atau perut selama proses pernapasan. Perubahan intensitas piksel ini mewakili gerakan pernapasan, yang terlihat dalam bentuk gelombang yang berulang dengan frekuensi yang tergantung pada kecepatan pernapasan individu. Sinyal ini dapat digunakan untuk memantau aktivitas pernapasan secara non-invasif.



Gambar 1: Visualisasi sinyal respirasi dan rPPG secara real-time

## 7.2 Visualisasi Sinyal rPPG

Grafik kedua (di bagian bawah) menunjukkan sinyal rPPG, yang diekstraksi dari channel hijau video wajah. Sinyal ini mewakili variasi volume darah yang dipompa oleh jantung dan dapat digunakan untuk mendeteksi detak jantung. Variasi kecil dalam intensitas cahaya yang dipantulkan dari kulit wajah tercatat sebagai fluktuasi pada grafik ini, yang menunjukkan ritme detak jantung. Sinyal ini berfungsi untuk mendeteksi detak jantung secara non-invasif hanya menggunakan kamera.

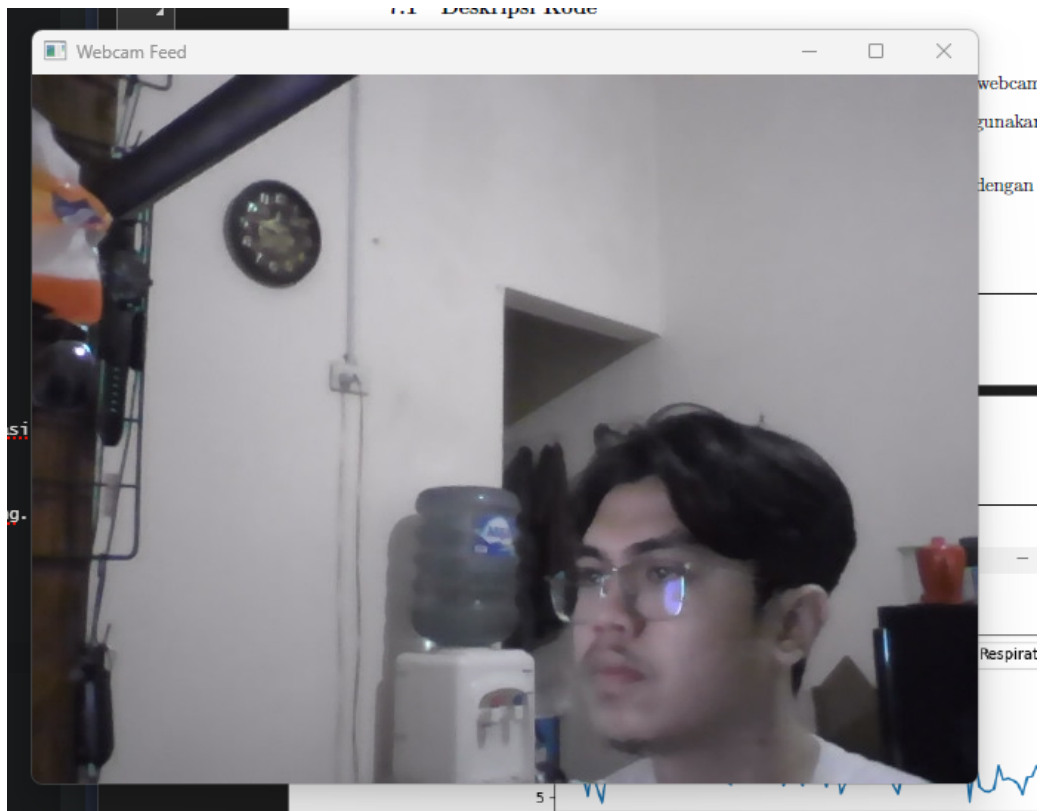
Kedua grafik ini memperlihatkan kemampuan sistem untuk mengekstraksi dan memvisualisasikan sinyal respirasi dan rPPG secara real-time, yang memiliki potensi untuk digunakan dalam aplikasi kesehatan untuk pemantauan vital signs tanpa perangkat medis invasif.

## 8 Kode Program

### 8.1 Deskripsi Kode

Kode program terdiri dari beberapa bagian utama:

- **main.py**: File utama yang mengatur alur pengolahan sinyal dari webcam hingga visualisasi.
- **respiration.py**: Modul untuk ekstraksi sinyal respirasi menggunakan filtering dan analisis intensitas piksel.
- **rppg.py**: Modul untuk ekstraksi sinyal rPPG dari channel hijau dengan metode detrending.
- **utils.py**: Fungsi utilitas tambahan seperti preprocessing frame.



Gambar 2: Objek/video yang diambil/screenshot

## 8.2 Potongan Kode Utama

Berikut adalah potongan kode utama dari `main.py`:

```
# File: main.py
import cv2
import numpy as np
from respiration import extract_respiration_signal
from rppg import extract_rppg_signal
import matplotlib.pyplot as plt

# Fungsi utama untuk membaca video dari webcam dan memproses frame secara real-time
def main():
    cap = cv2.VideoCapture(0) # Membuka akses ke webcam (kamera default perangkat).

    if not cap.isOpened(): # Memastikan webcam dapat diakses.

        print("Error: Tidak dapat mengakses webcam.")
        return

    plt.ion() # Mengaktifkan mode interaktif untuk pembaruan grafik secara real-time.
    fig, (resp_ax, rppg_ax) = plt.subplots(2, 1) # Membuat dua subplot untuk sinyal respirasi dan rPPG
    resp_ax.set_title("Sinyal Respirasi")
    rppg_ax.set_title("Sinyal rPPG")

    respiration_signal = []
    rppg_signal = []
```

```

while True: # Loop utama untuk membaca dan memproses frame webcam.
    ret, frame = cap.read()
    if not ret:
        print("Error: Tidak dapat membaca frame dari webcam.")
        break

    frame_gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY) # Mengubah frame ke grayscale.
    resp_signal = extract_respiration_signal(frame_gray)
    rppg_signal.append(extract_rppg_signal(frame))

    respiration_signal.append(resp_signal)

    if len(respiration_signal) > 100: # Jika panjang data sinyal respirasi melebihi 100 sampel
        respiration_signal.pop(0) # Hapus data terlama untuk menjaga jumlah maksimum sampel tetap

    if len(rppg_signal) > 100: # Jika panjang data sinyal rPPG melebihi 100 sampel
        rppg_signal.pop(0) # Hapus data terlama untuk menjaga jumlah maksimum sampel tetap 100

    resp_ax.clear()
    resp_ax.plot(respiration_signal, label="Respiration")
    resp_ax.legend()

    rppg_ax.clear()
    rppg_ax.plot(rppg_signal, label="rPPG")
    rppg_ax.legend()

    plt.pause(0.01) # Memberikan jeda kecil untuk memperbarui grafik matplotlib.

    cv2.imshow('Webcam Feed', frame)

    if cv2.waitKey(1) & 0xFF == ord('q'): # Tekan 'q' untuk keluar dari program.
        break

cap.release() # Melepaskan resource webcam yang digunakan.
cv2.destroyAllWindows() # Tutup semua jendela OpenCV yang terbuka.

if __name__ == "__main__":
    main()

```

## 9 Kesimpulan dan Saran

### 9.1 Kesimpulan

Proyek ini berhasil mengimplementasikan sistem pengukuran sinyal respirasi dan rPPG menggunakan Python. Sinyal respirasi dapat diekstraksi dari perubahan intensitas piksel, sementara sinyal rPPG diperoleh dari channel hijau video. Program ini menunjukkan visualisasi sinyal secara real-time yang dapat diterapkan dalam berbagai aplikasi kesehatan non-invasif.

### 9.2 Saran

- Memperluas fitur program dengan menambahkan analisis lebih lanjut, seperti deteksi anomali dalam sinyal respirasi atau rPPG.
- Mengintegrasikan algoritma machine learning untuk prediksi pola sinyal yang lebih kompleks.

- Mengoptimalkan performa program agar dapat bekerja pada perangkat dengan spesifikasi rendah tanpa mengorbankan kualitas sinyal.
- Menambahkan antarmuka pengguna grafis (GUI) untuk meningkatkan pengalaman pengguna.

## 10 Referensi dan Daftar Pustaka

[1]. [2]. [3].

- Matplotlib documentation, <https://matplotlib.org/stable/contents.html>
- OpenCV documentation, <https://docs.opencv.org/>
- SciPy documentation, <https://docs.scipy.org/doc/scipy/>

## References

- [1] J. Allen, "Photoplethysmography and its application in clinical physiological measurement," *Physiological Measurement*, vol. 28, no. 3, pp. R1–R39, 2007.
- [2] K. Humphreys and et al., "Noncontact photoplethysmography monitoring in the clinic," *BMJ Innovations*, vol. 2, no. 3, pp. 107–114, 2016.
- [3] M. Van Gastel and et al., "Real-time blood pulse extraction from facial video," *IEEE Transactions on Biomedical Engineering*, vol. 63, no. 12, pp. 2578–2587, 2016.