



Nama: **Maleakhi Pratama Tobing (121140225)**
Mata Kuliah: **Digital Signal Processing (IF3024)**

Tugas Ke: **4**
Tanggal: **24 Desember 2024**

1 Analisis Sinyal Respirasi dan rPPG dari Video Webcam

Proyek ini bertujuan untuk memproses sinyal fisiologis, yaitu sinyal respirasi dan photoplethysmography berbasis video (remote photoplethysmography atau rPPG), dari rekaman video menggunakan webcam. Data sinyal diekstraksi langsung dari frame video dengan teknik pemrosesan sederhana, seperti analisis intensitas piksel. Proses ini membantu dalam pengukuran kondisi fisiologis seseorang secara non-invasif.

1.1 Latar belakang

Dalam dunia medis dan kesehatan, pengukuran sinyal fisiologis seperti laju respirasi dan detak jantung adalah aspek penting untuk memantau kondisi kesehatan seseorang. Metode konvensional untuk mengukur sinyal tersebut sering kali memerlukan alat yang bersifat invasif atau membutuhkan kontak fisik langsung dengan pasien, seperti sensor oksimeter atau alat pengukur respirasi. Meskipun metode ini memiliki akurasi tinggi, penggunaannya dapat menyebabkan ketidaknyamanan, terutama bagi pasien dengan kondisi tertentu atau selama pemantauan jangka panjang.

Seiring dengan kemajuan teknologi, metode non-invasif berbasis video telah muncul sebagai solusi yang lebih nyaman untuk memantau sinyal fisiologis. Remote Photoplethysmography (rPPG) adalah salah satu teknik non-invasif yang memungkinkan pengambilan sinyal fisiologis hanya dengan menggunakan kamera biasa, seperti webcam. Dengan mendeteksi perubahan intensitas cahaya yang dipantulkan oleh kulit, rPPG dapat mengukur variasi sirkulasi darah, yang terkait erat dengan detak jantung. Selain itu, analisis pola intensitas visual dapat digunakan untuk mengestimasi laju respirasi melalui pergerakan tubuh yang halus.

Proyek ini bertujuan untuk mengembangkan kerangka dasar pemrosesan sinyal fisiologis berbasis video. Dengan memanfaatkan webcam yang tersedia secara luas, data sinyal respirasi dan rPPG diekstraksi dari frame video. Hasil dari penelitian ini dapat digunakan sebagai landasan untuk aplikasi yang lebih kompleks dalam pemantauan kesehatan, seperti deteksi stres, analisis kondisi tidur, atau pemantauan kesehatan jarak jauh.

Metode ini memberikan keuntungan berupa:

1. Kemudahan Akses: Menggunakan perangkat yang sederhana seperti webcam.
2. Non-Invasif: Mengurangi ketidaknyamanan pada pengguna dibandingkan metode tradisional.
3. Efisiensi Biaya: Tidak memerlukan perangkat khusus yang mahal.

1.2 Teknologi yang dipakai

1. Perangkat keras (Hardware)

- Webcam digunakan untuk menangkap video secara real-time sebagai input utama. Webcam berfungsi sebagai alat pengambil data visual yang nantinya diolah untuk mengekstraksi sinyal fisiologis seperti sinyal respirasi dan rPPG.

2. Perangkat Lunak (Software)

- Python Programming Language: Bahasa pemrograman utama yang digunakan dalam pengembangan proyek ini karena fleksibilitasnya serta dukungan pustaka yang luas untuk pengolahan citra dan sinyal.

3. Library Python

- OpenCV (cv2): Library untuk pengolahan gambar dan video. Dalam proyek ini, OpenCV digunakan untuk membaca data video dari webcam, mengonversi frame menjadi skala abu-abu, dan menampilkan video secara real-time.
- NumPy (numpy): Library untuk operasi numerik dan manipulasi array. Digunakan untuk menghitung rata-rata dan standar deviasi intensitas piksel sebagai sinyal fisiologis.
- Matplotlib (matplotlib.pyplot): Library untuk visualisasi data. Digunakan untuk menampilkan sinyal respirasi dan rPPG dalam bentuk grafik.
- SciPy (scipy.signal): Library untuk analisis sinyal. Fungsi butter dan filtfilt dari SciPy digunakan untuk membuat filter bandpass Butterworth, yang memungkinkan penghapusan noise dari sinyal.

1.3 Cara Kerja Sistem

Sistem yang dikembangkan dalam proyek ini bertujuan untuk menangkap, memproses, dan menganalisis sinyal fisiologis berupa sinyal respirasi dan rPPG dari data video yang diambil menggunakan webcam. Berikut adalah tahapan kerja sistem secara rinci:

1. Inisialisasi dan Pengambilan Data Video:

- Sistem memulai dengan inisialisasi webcam sebagai alat untuk menangkap data video secara real-time.
- Webcam akan mengambil data berupa frame video dengan kecepatan tertentu (30 frame per detik).
- Frame video yang berhasil diambil akan digunakan sebagai input untuk langkah pemrosesan lebih lanjut

2. Pemrosesan frame:

- Setiap frame video yang diterima diubah menjadi format skala abu-abu (grayscale) menggunakan library OpenCV.
- Proses konversi ini dilakukan untuk menyederhanakan analisis, karena skala abu-abu hanya memiliki satu kanal intensitas dibandingkan dengan gambar berwarna (RGB).
- Dari frame dalam skala abu-abu, dua jenis sinyal fisiologis diekstraksi: - Sinyal Respirasi: Dihitung sebagai rata-rata intensitas piksel pada seluruh frame. - Sinyal rPPG: Dihitung sebagai standar deviasi intensitas piksel pada seluruh frame.

3. Penyimpanan dan Pencatatan Waktu:

- Nilai sinyal yang diekstraksi dari setiap frame disimpan dalam daftar (list) untuk dianalisis lebih lanjut.

- Waktu pengambilan sinyal juga dicatat berdasarkan jumlah frame yang telah diolah, dengan asumsi frekuensi pengambilan data adalah 30 frame per detik.

4. Visualisasi Video:

- Frame yang sedang diproses ditampilkan kembali ke layar untuk memberikan gambaran langsung tentang video yang sedang diambil.
- Pengguna dapat menghentikan proses dengan menekan tombol q.

5. Visualisasi Sinyal:

- Setelah proses pengambilan data selesai, sistem menampilkan grafik sinyal respirasi dan rPPG berdasarkan waktu.
- Grafik ini memberikan representasi intensitas sinyal terhadap waktu, yang dapat digunakan untuk analisis lebih lanjut.

1.4 Penjelasan Code

Kode ini adalah implementasi program Python yang memanfaatkan webcam untuk menangkap video real-time, memproses frame video, dan menampilkan sinyal fisiologis berupa sinyal respirasi dan rPPG (Remote Photoplethysmography). Berikut adalah deskripsi fungsi dan komponen utama dalam kode:

(a) Library yang digunakan

```
1 import cv2
2 import numpy as np
3 import matplotlib.pyplot as plt
4 from scipy.signal import butter, filtfilt
```

Kode 1: Library yang digunakan

- cv2 (OpenCV): Untuk membaca video dari webcam, memproses frame, dan menampilkan video.
- numpy: Untuk operasi numerik, seperti menghitung rata-rata dan standar deviasi intensitas piksel.
- matplotlib.pyplot: Untuk membuat grafik real-time dari sinyal respirasi dan rPPG.
- scipy.signal: Untuk membuat filter bandpass Butterworth.

(b) Fungsi butter bandpass filter Fungsi ini membuat dan menerapkan filter bandpass Butterworth.

```
1 def butter_bandpass_filter(data, lowcut, highcut, fs, order=5):
2     nyquist = 0.5 * fs
3     low = lowcut / nyquist
4     high = highcut / nyquist
5     b, a = butter(order, [low, high], btype='band')
6     return filtfilt(b, a, data)
```

Kode 2: Fungsi butter bandpass filter

input:

- data: Data sinyal yang ingin difilter.
- lowcut, highcut: Frekuensi cutoff bawah dan atas.
- fs: Frekuensi sampling (contoh: 30 FPS).
- order: Orde filter.

output : Sinyal yang telah difilter untuk mengurangi noise atau frekuensi tidak diinginkan.

- (c) Fungsi process frame Fungsi ini memproses frame video untuk mengekstraksi sinyal respirasi dan rPPG.

```
1 def process_frame(frame):
2     gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
3     signal_resp = np.mean(gray)
4     signal_rppg = np.std(gray)
5     return signal_resp, signal_rppg
```

Kode 3: Fungsi process frame

Langkah Pemrosesan:

- i. Frame dikonversi menjadi skala abu-abu menggunakan cv2.cvtColor.
- ii. Sinyal Respirasi: Rata-rata intensitas piksel pada frame.
- iii. Sinyal rPPG: Standar deviasi intensitas piksel.

Output:

- signal resp: Sinyal respirasi.
- signal rppg: Sinyal rPPG.

- (d) Fungsi release resources Fungsi ini memastikan semua sumber daya (kamera dan jendela) dirilis dengan aman saat program selesai.

```
1 def release_resources(cap):
2     if cap.isOpened():
3         cap.release()
4     cv2.destroyAllWindows()
5     print("Sumber daya kamera dan jendela sudah dirilis.")
```

Kode 4: Fungsi release resources

- (e) Fungsi Utama main Fungsi utama yang mengatur alur kerja program.

```
1 def main():
2     cap = cv2.VideoCapture(0) # Inisialisasi webcam
3     if not cap.isOpened():
4         print("Error: Tidak dapat membuka webcam.")
5         return
6
7     print("Tekan 'q' untuk keluar.")
8
9     respiration_signals = []
10    rppg_signals = []
11    timestamps = []
12    fs = 30.0 # Frekuensi sampling (30 FPS)
13
14    # Grafik Real-Time
15    plt.ion()
16    fig, (ax1, ax2) = plt.subplots(2, 1, figsize=(10, 6))
17    line1, = ax1.plot([], [], label='Sinyal Respirasi', color='b')
18    line2, = ax2.plot([], [], label='Sinyal rPPG', color='g')
```

Kode 5: Fungsi Utama main

- Inisialisasi webcam untuk menangkap video. Jika gagal, program berhenti.
- Grafik real-time dibuat untuk sinyal respirasi dan rPPG.
- loop utama

```

1  try:
2      while True:
3          ret, frame = cap.read()
4          if not ret:
5              print("Error: Tidak dapat membaca frame.")
6              break
7
8          # Ekstraksi sinyal
9          resp_signal, rppg_signal = process_frame(frame)
10         respiration_signals.append(resp_signal)
11         rppg_signals.append(rppg_signal)
12         timestamps.append(len(timestamps) / fs)
13
14         # Batasi panjang data ke 10 detik terakhir
15         if len(timestamps) > fs * 10:
16             respiration_signals = respiration_signals[-int(fs * 10):]
17             rppg_signals = rppg_signals[-int(fs * 10):]
18             timestamps = timestamps[-int(fs * 10):]
19
20         # Perbarui grafik
21         line1.set_data(timestamps, respiration_signals)
22         line2.set_data(timestamps, rppg_signals)
23         ax1.set_xlim(timestamps[0], timestamps[-1])
24         ax2.set_xlim(timestamps[0], timestamps[-1])
25         fig.canvas.draw()
26         fig.canvas.flush_events()
27
28         cv2.imshow("Webcam", frame)
29
30         if cv2.waitKey(1) & 0xFF == ord('q'):
31             print("Menutup program atas permintaan pengguna...")
32             break

```

Kode 6: Fungsi Utama main

Ekstraksi Sinyal: Setiap frame diproses untuk menghasilkan sinyal respirasi dan rPPG.

Grafik Real-Time: Data diperbarui setiap frame untuk ditampilkan dalam grafik.

(f) Penanganan Error dan Akhir Program:

```

1  except Exception as e:
2      print(f"Terjadi error: {e}")
3  finally:
4      release_resources(cap)
5      plt.ioff()
6      plt.show()
7      print("Kamera dirilis dan jendela ditutup.")

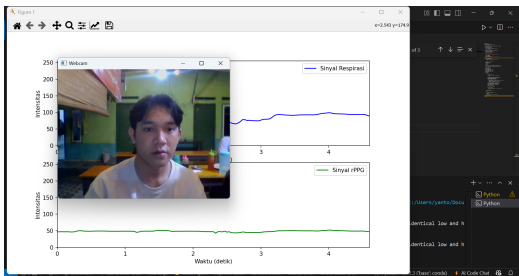
```

Kode 7: Fungsi Utama main

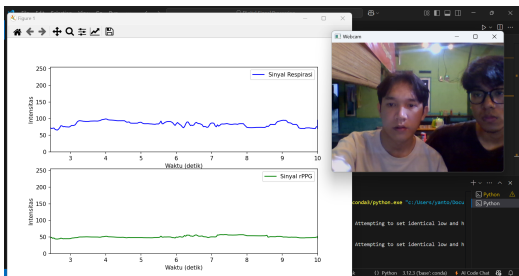
- Penanganan Error: Jika terjadi kesalahan, program tetap merilis sumber daya.
- Akhir Program: Grafik ditampilkan sebagai hasil akhir.

2 Dokumentasi Digital Signal Processing

Berikut ini adalah dokumentasi dari proyek Drawing with Hands:



(a) Sinyal Respirasi



(b) Sinyal rPPG

Gambar 1: Dokumentasi Digital Signal Processing