



Nama: **Justin Halim** (120140185)

Judul Tugas: **Respiration-rPPG**

Mata Kuliah: **Pengolahan Sinyal Digital (IF3024)**

Tanggal: **24 Desember 2024**

Bab 1 Pendahuluan

1.1 Latar Belakang

Pada era digital ini, teknologi pengukuran fisiologi manusia semakin berkembang, salah satunya dengan menggunakan metode remote-photoplethysmography (rPPG) dan analisis gerakan tubuh untuk mengukur sinyal pernapasan. Kedua sinyal ini penting dalam pemantauan kesehatan, seperti dalam deteksi gangguan pernapasan dan analisis detak jantung.

Metode rPPG memungkinkan pemantauan detak jantung tanpa menggunakan sensor fisik, cukup dengan merekam video dari wajah atau kulit manusia. Sementara itu, sinyal respirasi, yang berkaitan dengan pergerakan dada atau perut, dapat diukur melalui deteksi pergerakan tubuh yang terekam oleh kamera. Gabungan kedua sinyal ini dapat memberikan gambaran yang lebih lengkap tentang status kesehatan seseorang.

Penerapan teknologi ini dapat digunakan dalam berbagai bidang seperti kesehatan jarak jauh, telemedicine, dan penelitian kesehatan manusia. Oleh karena itu, penting untuk mengembangkan metode yang dapat secara efektif menggabungkan kedua sistem pengukuran ini dalam satu aplikasi real-time berbasis webcam.

1.2 Rumusan Masalah

Rumusan masalah dari proyek ini adalah:

1. Bagaimana cara mengukur sinyal respirasi dan sinyal rPPG dari video webcam secara real-time?
2. Bagaimana menggabungkan dua sistem pengukuran sinyal tersebut dalam satu program yang efisien dan akurat?
3. Apa saja tantangan dalam pengukuran sinyal respirasi dan rPPG dengan menggunakan webcam?

1.3 Tujuan Penelitian

Tujuan dari penelitian Respiration-rPPG ini adalah sebagai berikut.

1. Mengembangkan program yang dapat mengukur dan menampilkan sinyal respirasi dan rPPG secara real-time menggunakan webcam.
2. Menggunakan teknik pemrosesan citra dan sinyal untuk memperoleh dua sinyal fisiologi tersebut dengan akurat.
3. Mengintegrasikan pengukuran sinyal respirasi dan rPPG dalam satu sistem yang dapat ditampilkan secara visual melalui grafik.

Bab 2 - Dasar Teori

2.1 Remote-Photoplethysmography (rPPG)

rPPG adalah teknik yang digunakan untuk mengukur detak jantung dengan menganalisis perubahan warna pada kulit yang disebabkan oleh fluktuasi volume darah yang terpompa ke kapiler. Perubahan ini biasanya terlihat pada wajah atau bagian tubuh yang terekspos cahaya. Dengan memanfaatkan kamera dan pemrosesan citra, sinyal PPG dapat direkonstruksi tanpa menggunakan sensor fisik yang ditempelkan pada tubuh[1].

2.2 Sinyal Respirasi

Sinyal respirasi adalah sinyal yang menggambarkan pola pernapasan tubuh manusia. Pola pernapasan dapat terdeteksi dari pergerakan dada atau perut yang dapat diamati melalui pemrosesan video. Dalam penelitian ini, gerakan tubuh manusia yang terekam dalam video akan digunakan untuk mendeteksi pola pernapasan[2].

2.3 Filtering Sinyal

Sinyal yang diukur seringkali mengandung noise. Oleh karena itu, dalam penelitian ini digunakan filter low-pass untuk menghaluskan sinyal rPPG, dengan tujuan menghilangkan noise frekuensi tinggi yang tidak relevan dengan sinyal fisiologi yang diinginkan.

Bab 3 - Metode Penelitian

3.1 Pengumpulan Data

Pengumpulan data dilakukan dengan menggunakan webcam untuk merekam video wajah atau tubuh seseorang. Video ini akan digunakan untuk memperoleh sinyal respirasi dan rPPG secara real-time.

3.2 Pemrosesan Sinyal

1. Sinyal rPPG diukur dengan menganalisis perubahan kecerahan warna kulit, yang terkait dengan detak jantung. Perubahan ini dihitung berdasarkan rata-rata saluran "V" dalam ruang warna HSV.
2. Sinyal respirasi dihitung dengan mendeteksi pergerakan wajah menggunakan deteksi wajah berbasis Haar Cascade. Perubahan warna atau gerakan pada wajah ini akan digunakan sebagai indikator respirasi.

3.3 Pengolahan dan Visualisasi

Setelah sinyal dihitung, sinyal rPPG akan difilter menggunakan filter low-pass untuk menghaluskan fluktuasi yang tidak diinginkan. Kedua sinyal ini kemudian divisualisasikan dalam grafik secara real-time menggunakan pustaka matplotlib.

3.4 Pengujian dan Evaluasi

Proses pengujian dilakukan dengan merekam video berbagai subjek yang melakukan pernapasan normal. Program akan mengevaluasi akurasi pengukuran sinyal respirasi dan rPPG berdasarkan data yang diperoleh dan membandingkannya dengan standar pengukuran fisiologi.

Bab 4 - Penjelasan dan Hasil

Pada bab ini, akan dijelaskan langkah-langkah yang dilakukan dalam pengembangan program yang menggabungkan sistem pengukuran sinyal respirasi dan rPPG secara real-time menggunakan webcam. Penjelasan akan mencakup instalasi library yang diperlukan, penjelasan tentang kode yang digunakan dalam pengembangan program, serta hasil/output yang diperoleh dari menjalankan program.

4.1 Instalasi Library

Untuk menjalankan program ini, beberapa library atau pustaka Python diperlukan. Berikut adalah langkah-langkah instalasi untuk pustaka yang digunakan.

1. OpenCV - Digunakan untuk memproses video secara real-time dan mendeteksi wajah pada frame video.

```
1 pip install opencv-python
2
```

Kode 1: Instalasi OpenCV

2. NumPy - Digunakan untuk manipulasi data array, termasuk operasi pada sinyal, seperti perhitungan rata-rata dan penerapan filter.

```
1 pip install numpy
2
```

Kode 2: Instalasi NumPy

3. Matplotlib - Digunakan untuk visualisasi sinyal respirasi dan rPPG dalam bentuk grafik secara real-time.

```
1 pip install matplotlib
2
```

Kode 3: Instalasi matplotlib

4.2 Penjelasan Kode

Kode yang disediakan menggunakan teknik pemrosesan citra dan sinyal untuk mengukur dua jenis sinyal fisiologi: sinyal rPPG (remote Photoplethysmography) dan sinyal respirasi. Berikut adalah penjelasan rinci tentang bagian-bagian utama kode.

1. Import Library

```
1
2 import cv2
3 import numpy as np
4 import matplotlib.pyplot as plt
5
6
```

Kode 4: Import Library

- (a) cv2: Digunakan untuk menangkap video dari webcam dan melakukan deteksi wajah menggunakan Haar Cascade.
- (b) NumPy: Digunakan untuk operasi array dan filter sinyal.
- (c) matplotlib: Digunakan untuk membuat grafik visualisasi sinyal secara real-time.

2. Fungsi moving_average_filter

```
1
2 def moving_average_filter(data, window_size=5):
3     weights = np.ones(window_size) / window_size
4     return np.convolve(data, weights, mode='valid')
5
6
```

Kode 5: moving_average_filter

Fungsi ini digunakan untuk membuat filter moving average untuk menyaring sinyal rPPG. Filter ini menghaluskan sinyal dengan mengambil rata-rata dari nilai-nilai dalam jendela (window) yang ditentukan.

3. Inisialisasi Webcam

```
1
2 cap = cv2.VideoCapture(0)
3 fps = cap.get(cv2.CAP_PROP_FPS)
4
5
```

Kode 6: Webcam Initialization

cv2.VideoCapture(0) membuka webcam untuk menangkap video. fps mengambil frame per detik dari webcam untuk digunakan dalam pengolahan video.

4. Fungsi calculate_rPPG

```
1
2 def calculate_rPPG(frame):
3     hsv = cv2.cvtColor(frame, cv2.COLOR_BGR2HSV)
4     v_channel = hsv[:, :, 2]
5     avg_brightness = np.mean(v_channel)
6     return avg_brightness
7
8
```

Kode 7: Fungsi calculate_rPPG

Fungsi ini menghitung sinyal rPPG dengan mengubah gambar menjadi ruang warna HSV dan menghitung rata-rata kecerahan dari saluran "V" (Value) yang menggambarkan intensitas cahaya. Perubahan kecerahan ini mencerminkan fluktuasi volume darah yang terpompa, yang berhubungan dengan detak jantung.

5. Fungsi calculate_respiration_signal

```
1
2 def calculate_respiration_signal(frame):
3     gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
4     face_cascade = cv2.CascadeClassifier(cv2.data.haarcascades + '
5     haarcascade_frontalface_default.xml')
6     faces = face_cascade.detectMultiScale(gray, scaleFactor=1.1, minNeighbors=5, minSize=(30,
7     30))
8
9     if len(faces) > 0:
10         (x, y, w, h) = faces[0]
11         roi = frame[y:y+h, x:x+w]
12         avg_color = np.mean(roi, axis=(0, 1))
13         return avg_color[1]
14     return 0
```

Kode 8: Fungsi calculate_respiration_signal

Fungsi ini mendeteksi wajah dalam frame menggunakan Haar Cascade. Setelah mendeteksi wajah, ia menghitung nilai rata-rata dari area wajah tersebut untuk menghasilkan sinyal respirasi. Perubahan warna pada wajah dapat mencerminkan pergerakan dada atau perut yang terjadi saat bernapas.

6. Proses Pengolahan dan Visualisasi Real-Time

```

1
2     while True:
3         ret, frame = cap.read()
4         if not ret:
5             break
6
7         # Menghitung sinyal rPPG dan respirasi
8         rPPG_val = calculate_rPPG(frame)
9         respiration_val = calculate_respiration_signal(frame)
10
11        # Menambahkan nilai sinyal ke dalam daftar
12        rPPG_signal.append(rPPG_val)
13        respiration_signal.append(respiration_val)
14
15        # Filter sinyal rPPG
16        if len(rPPG_signal) > window_size:
17            rPPG_signal_filtered = moving_average_filter(np.array(rPPG_signal), window_size)
18        else:
19            rPPG_signal_filtered = rPPG_signal
20
21        # Visualisasi
22        ax1.clear()
23        ax1.plot(x_axis[:len(rPPG_signal_filtered)], rPPG_signal_filtered, label="rPPG Signal")
24        ax1.set_title("Sinyal rPPG")
25        ax1.set_xlabel("Waktu (detik)")
26        ax1.set_ylabel("Amplitudo")
27        ax1.legend()
28
29        ax2.clear()
30        ax2.plot(x_axis[:len(respiration_signal)], respiration_signal, label="Sinyal Respirasi")
31        ax2.set_title("Sinyal Respirasi")
32        ax2.set_xlabel("Waktu (detik)")
33        ax2.set_ylabel("Amplitudo")
34        ax2.legend()
35
36        plt.pause(0.001)
37
38        # Menampilkan frame video
39        cv2.imshow("Webcam", frame)
40
41        if cv2.waitKey(1) & 0xFF == ord('q'):
42            break
43
44

```

Kode 9: Proses Pengolahan dan Visualisasi Real-Time

- Mengambil frame video: Setiap frame video diambil secara real-time dan kemudian diproses untuk menghitung sinyal rPPG dan respirasi.
- Visualisasi dengan matplotlib: Dua grafik akan diperbarui secara real-time untuk menampilkan sinyal rPPG dan respirasi.
- Menampilkan video: Frame video yang sedang diproses juga ditampilkan di jendela terpisah.

7. Membersihkan Resource

```
1
2 cap.release()
3 cv2.destroyAllWindows()
4 plt.ioff()
5 plt.show()
6
```

Kode 10: Membersihkan Resource

Program akan melepaskan perangkat webcam dan menutup semua jendela OpenCV serta menghentikan mode interaktif matplotlib setelah proses selesai.

4.3 Hasil atau Output

Setelah menjalankan program, berikut adalah beberapa hasil yang dapat diperoleh:

1. Visualisasi Sinyal rPPG: Grafik yang menampilkan fluktuasi sinyal yang terkait dengan detak jantung. Grafik ini akan terus diperbarui secara real-time dengan nilai-nilai baru dari sinyal rPPG yang dihitung setiap frame.
2. Visualisasi Sinyal Respirasi: Grafik yang menunjukkan perubahan amplitudo yang berhubungan dengan pergerakan tubuh saat bernapas. Ini diukur berdasarkan pergerakan wajah atau tubuh yang terdeteksi dalam video.
3. Penyaringan Sinyal: Sinyal rPPG yang ditampilkan pada grafik sudah melalui filter moving average untuk mengurangi noise, memberikan representasi yang lebih halus dari sinyal detak jantung.
4. Webcam Display: Video yang ditangkap oleh webcam akan ditampilkan dalam jendela terpisah, dan pengguna dapat menekan tombol q untuk keluar dari program.

Hasilnya menunjukkan bahwa program dapat mengukur dan memvisualisasikan kedua sinyal fisiologi ini secara real-time.

Bab 5 - Kesimpulan

Kesimpulan

Program ini berhasil mengukur sinyal respirasi dan rPPG secara real-time menggunakan video webcam dan pemrosesan citra. Gabungan kedua sinyal ini memberikan wawasan yang lebih baik tentang status fisiologis seseorang, seperti detak jantung dan pola pernapasan. Meskipun akurasi dapat ditingkatkan lebih lanjut, program ini menunjukkan potensi besar dalam penggunaan teknologi jarak jauh untuk pemantauan kesehatan.

References

- [1] d. , “Opencv: Meanshift and camshift,” docs.opencv.org. [Online]. Available: https://docs.opencv.org/4.x/d7/d00/tutorial_meanshift.html
- [2] N. Philbert, “Ta: Sistem deteksi detak jantung melalui wajah secara computer vision menggunakan metode remote photoplethysmograph - repositori universitas dinamika,” *Dinamika.ac.id*, 2023. [Online]. Available: <https://repository.dinamika.ac.id/id/eprint/7255>