Feature Extraction of a PPG (Photoplethysmogram) Signal

Photoplethysmography (PPG) is a simple and non-invasive technique used to measure changes in blood volume in the body. It works by shining light onto the skin and detecting variations in the reflected or transmitted light, which correspond to blood flow changes. PPG signals are widely used in heart rate monitoring, oxygen saturation measurement, and cardiovascular health analysis.

Since the PPG signal contains valuable physiological information, we use **feature extraction** techniques to analyze it effectively. These techniques help us identify key characteristics such as heart rate, blood circulation patterns, and heart rate variability.

---

1. Understanding the PPG Signal

The PPG signal consists of repeating waveforms that represent heartbeats. Each cycle of the signal contains:

- **A Systolic Peak (Highest Point):** This corresponds to the maximum blood volume in the arteries when the heart contracts.
- **A Diastolic Point (Lowest Point):** This occurs when the blood volume is at its lowest before the next heartbeat.

The frequency of these peaks is directly related to heart rate, and their shape provides insights into blood circulation. However, the raw PPG signal can be noisy due to motion artifacts, ambient light, and sensor imperfections.

---

2. Preprocessing the PPG Signal (Filtering)

Before extracting features, the PPG signal must be cleaned to remove unwanted noise. This process involves:

- **Removing High-Frequency Noise:** Using a **low-pass filter** to eliminate sudden spikes and artifacts.
- **Removing Slow Variations (Baseline Drift):** Using a **high-pass filter** to focus on heartbeat-related changes.
- **Smoothing the Signal:** Applying signal processing techniques to improve peak detection.

The cleaned signal is now ready for feature extraction.

---

3. Peak Detection and RR Interval Calculation

After filtering, we detect peaks in the PPG signal. These peaks represent heartbeats, and the time difference between two consecutive peaks is called the **RR interval** (measured in seconds).

$$\text{RR Interval} = \text{Time between two successive peaks}$$

By analyzing RR intervals, we can calculate:

- **Heart Rate (HR):** The number of heartbeats per minute (BPM).

Interval (s)HR= 60 / RR Interval (s)

- **Heart Rate Variability (HRV):** The variation in RR intervals, which provides insights into stress levels and cardiovascular health.

---

4. Other Important Features in PPG Signals

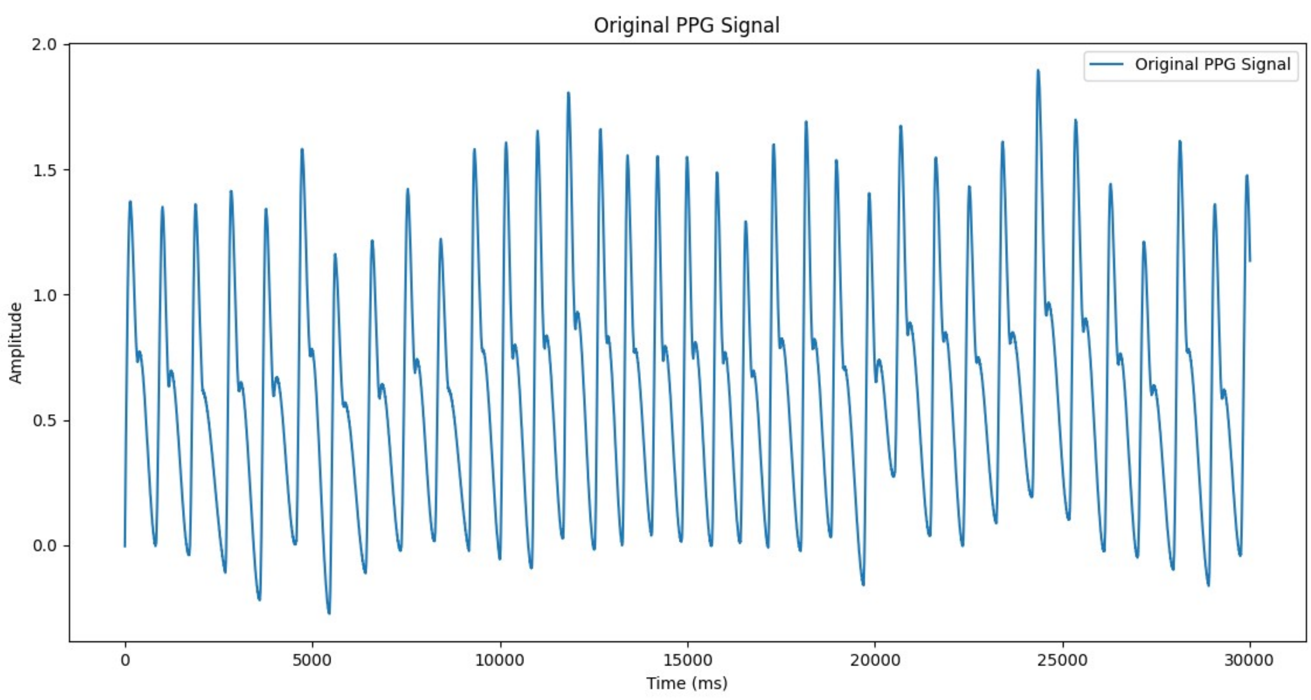
Apart from heart rate and RR intervals, we can extract additional features from the PPG signal:

- **Pulse Amplitude:** The height difference between peaks and valleys, indicating blood flow strength.
- **Pulse Width:** The time duration of each pulse, giving insights into arterial stiffness.
- **Pulse Slope:** The rate at which blood flow increases or decreases.

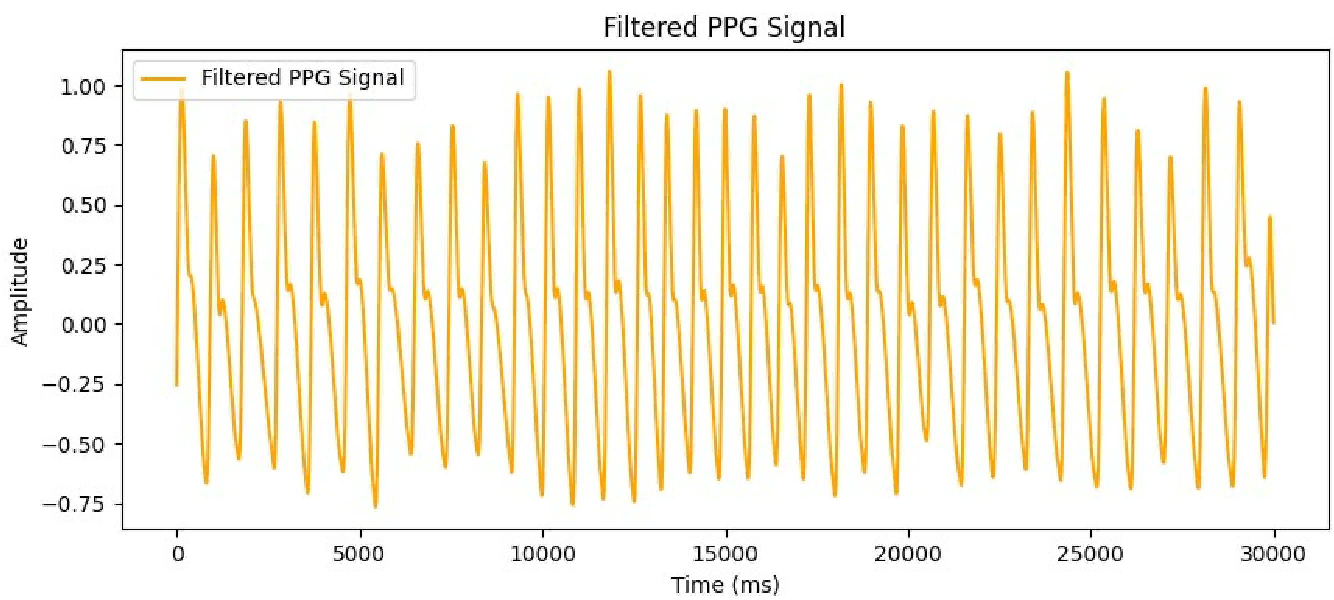
These features are useful for diagnosing heart conditions and monitoring overall health.

Source code:

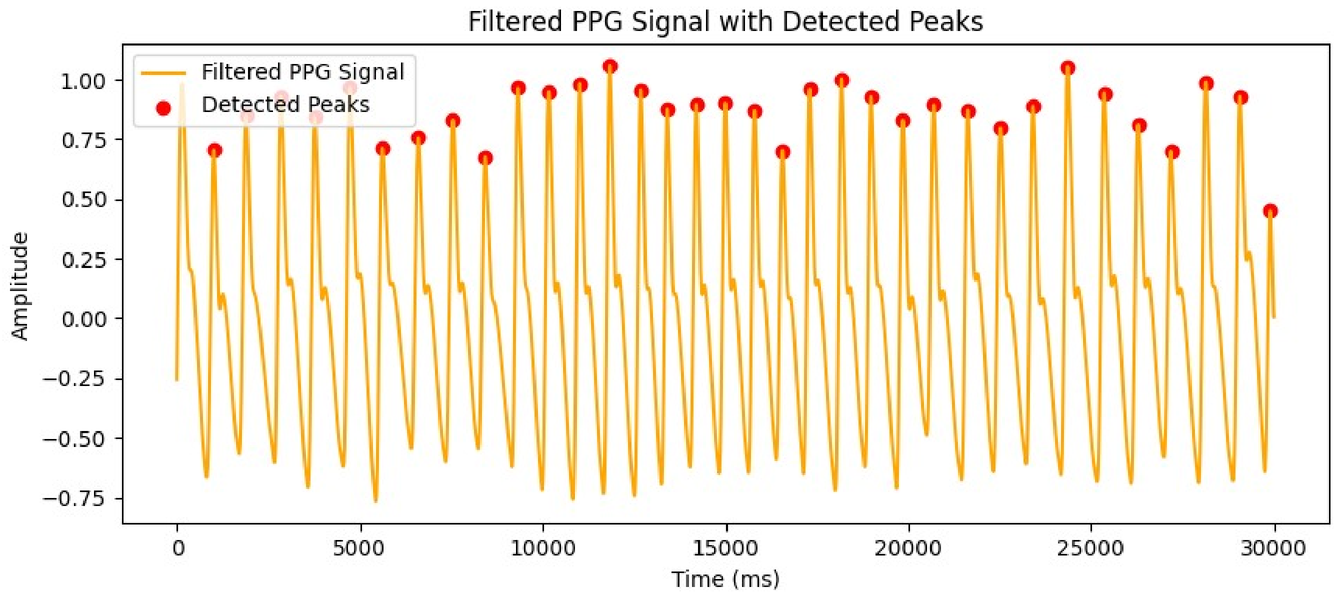
```
import neurokit2 as nk
import numpy as np
import matplotlib.pyplot as plt

# 1. Generate a synthetic PPG signal (replace this with real PPG data)
ppg_signal = nk.ppg_simulate(duration=30, sampling_rate=1000)

# Plot the original PPG signal
plt.figure(figsize=(10, 4))
plt.plot(ppg_signal, label="Original PPG Signal")
plt.title("Original PPG Signal")
plt.xlabel("Time (ms)")
plt.ylabel("Amplitude")
plt.legend()
plt.show()
```



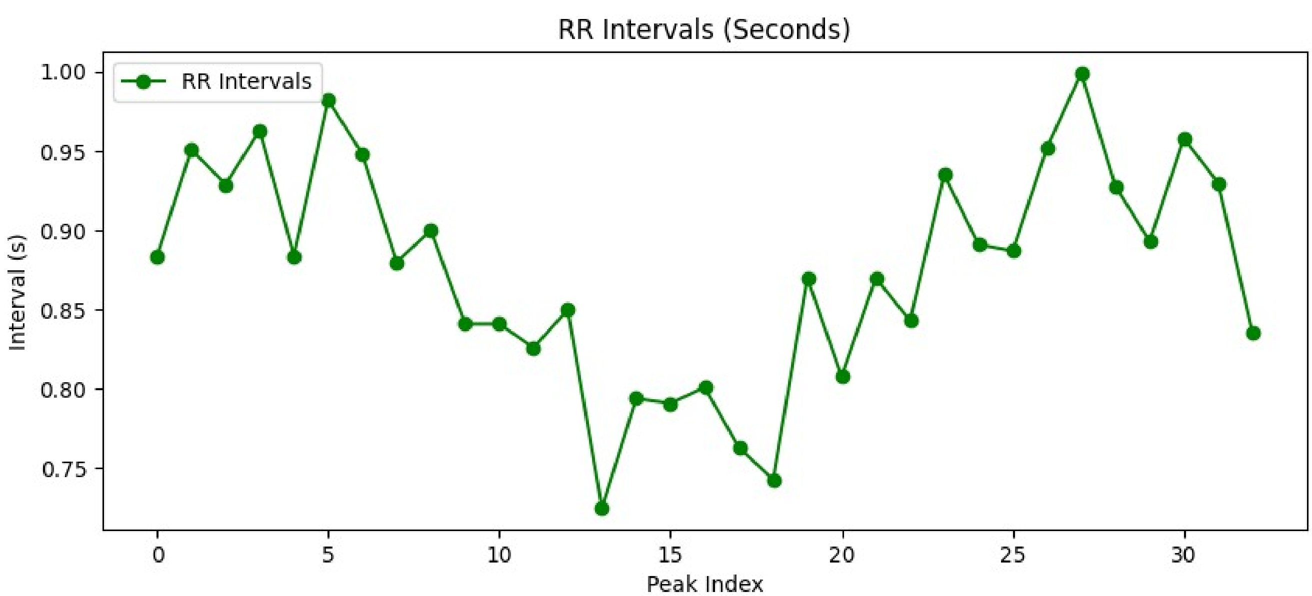
```
# 2. Filter the signal
ppg_filtered = nk.ppg_clean(ppg_signal, sampling_rate=1000)

# Plot the filtered PPG signal
plt.figure(figsize=(10, 4))
plt.plot(ppg_filtered, label="Filtered PPG Signal", color="orange")
plt.title("Filtered PPG Signal")
plt.xlabel("Time (ms)")
plt.ylabel("Amplitude")
plt.legend()
plt.show()
```



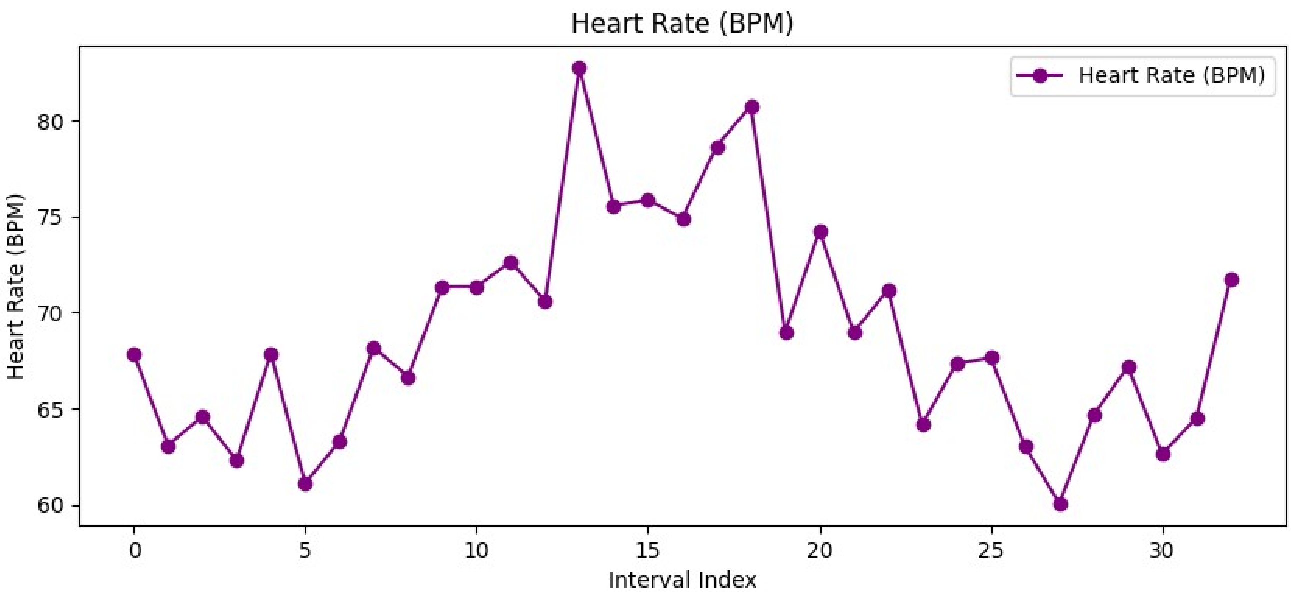

```
# 3. Detect Peaks
peaks = nk.ppg_findpeaks(ppg_filtered, sampling_rate=1000)
peak_indices = peaks["PPG_Peaks"]

# Plot the filtered signal with detected peaks
plt.figure(figsize=(10, 4))
plt.plot(ppg_filtered, label="Filtered PPG Signal", color="orange")
plt.scatter(peak_indices, ppg_filtered[peak_indices], color="red",
label="Detected Peaks")
plt.title("Filtered PPG Signal with Detected Peaks")
plt.xlabel("Time (ms)")
plt.ylabel("Amplitude")
plt.legend()
plt.show()
```




```
# 4. Calculate RR Intervals
rr_intervals = np.diff(peak_indices) / 1000 # Convert to seconds

# Plot RR intervals
plt.figure(figsize=(10, 4))
plt.plot(rr_intervals, label="RR Intervals", color="green", marker="o")
plt.title("RR Intervals (Seconds)")
plt.xlabel("Peak Index")
plt.ylabel("Interval (s)")
plt.legend()
plt.show()
```



```
# 5. Calculate Heart Rate
heart_rate = 60 / rr_intervals # BPM

# Plot Heart Rate
plt.figure(figsize=(10, 4))
plt.plot(heart_rate, label="Heart Rate (BPM)", color="purple", marker="o")
plt.title("Heart Rate (BPM)")
plt.xlabel("Interval Index")
plt.ylabel("Heart Rate (BPM)")
plt.legend()
plt.show()
```



Conclusion

Feature extraction from a PPG signal is crucial for understanding heart health. By filtering the signal, detecting peaks, and analyzing RR intervals, we can calculate heart rate and other important cardiovascular metrics. These techniques make PPG-based monitoring an effective tool for wearable devices and medical applications.