PPG Signal Analysis for Heart Rate and Abnormality Detection

Abstract

This report details the analysis of Photoplethysmogram (PPG) signals for heart rate monitoring and abnormality detection. It includes preprocessing, peak detection, heart rate calculation, and identification of abnormalities such as bradycardia, tachycardia, and irregular heartbeats. The report includes placeholders for visualizations.

Scope

1. Data Acquisition: Load PPG signal data from a CSV file.
2. Preprocessing: Apply bandpass filtering to remove noise.
3. Peak Detection: Identify R-peaks using the find_peaks function.
4. Heart Rate Calculation: Compute RR intervals and heart rate.
5. Abnormality Detection: Detect bradycardia, tachycardia, and irregular heartbeats.
6. Visualization: Plot raw and filtered signals, R-peaks, heart rate trends, and abnormalities.
7. Report Generation: Provide heart rate statistics and detected abnormalities.

Code Implementation

Import Required Libraries

```
import numpy as np from scipy.signal import butter,
filtfilt, find_peaks import matplotlib.pyplot as plt
import pandas as pd
```

Define PPGAnalyzer Class

```
class PPGAnalyzer:     def __init__(self,
sampling_rate=100):
        self.fs = sampling_rate
self.ppg_data = None
self.filtered_data = None
self.r_peaks = None
self.rr_intervals = None
self.heart_rates = None
```

Load Data

```
def load_data(self, file_path):
try:
        self.ppg_data = pd.read_csv(file_path).iloc[:,
0].values         return True     except Exception as e:
        print(f'Error loading data: {e}')
return False
```

Bandpass Filter

```
def bandpass_filter(self, lowcut=0.7, highcut=7.5):
    nyquist = 0.5 * self.fs
low = lowcut / nyquist
```

```
    high = highcut / nyquist     b, a = butter(2,
[low, high], btype='band')     self.filtered_data =
filtfilt(b, a, self.ppg_data)
```

Detect R-Peaks

```
def detect_r_peaks(self, height=None, distance=None):
if height is None:
        height = 0.5 * np.max(self.filtered_data)
if distance is None:
        distance = int(0.4 * self.fs)     self.r_peaks, _ =
find_peaks(self.filtered_data, height=height, distance=distance)
```

Analyze Heart Rate

```
def analyze_heart_rate(self):
    self.rr_intervals = np.diff(self.r_peaks) / self.fs
self.heart_rates = 60 / self.rr_intervals
```

Detect Abnormalities

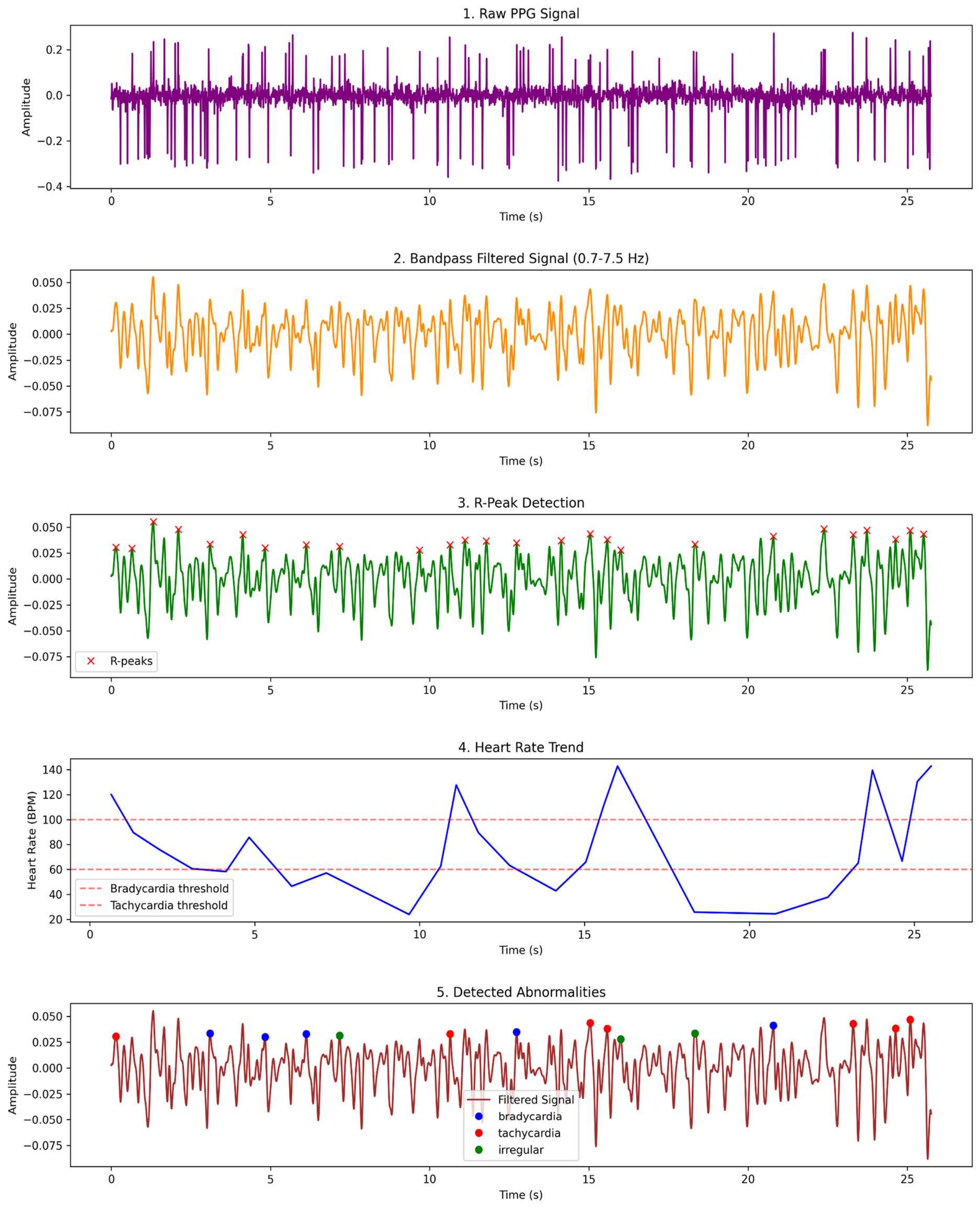
```
def detect_abnormalities(self):
    abnormalities = {'bradycardia': [], 'tachycardia': [],
'irregular': []}     bradycardia_idx = np.where(self.heart_rates <
60)[0]     abnormalities['bradycardia'] =
self.r_peaks[bradycardia_idx]     tachycardia_idx =
np.where(self.heart_rates > 100)[0]
abnormalities['tachycardia'] = self.r_peaks[tachycardia_idx]
```

Main Function

```
def analyze_ppg_file(file_path, sampling_rate=100):
    analyzer = PPGAnalyzer(sampling_rate)     if
analyzer.load_data(file_path):
analyzer.bandpass_filter()
analyzer.detect_r_peaks()
analyzer.analyze_heart_rate()         abnormalities =
analyzer.detect_abnormalities()         print('Analysis
Complete', abnormalities)     return abnormalities
```

PPG Signal Analysis for Heart Rate and Abnormality Detection

Results



PPG Signal Analysis for Heart Rate and Abnormality Detection

Conclusion

The PPG signal analysis system successfully extracts heart rate, detects abnormalities, and provides insights into heart health. The placeholders provided in this document should be replaced with actual plots for a complete analysis. Future improvements could include real-time processing and enhanced filtering techniques.