

Feature Extraction and Peak Detection of Cosine Signal

Code :

```
import numpy as np
import matplotlib.pyplot as plt
from scipy.signal import find_peaks

# Generate a cosine wave to simulate a PPG signal
fs = 100
t = np.linspace(0, 10, fs * 10)
ppg_signal = np.cos(2 * np.pi * 1.2 * t)

# Add some noise to make it more realistic
noise = 0.1 * np.random.randn(len(t))
ppg_signal_noisy = ppg_signal + noise

# Detect peaks
peaks, _ = find_peaks(ppg_signal_noisy, height=0)

# Feature extraction (example: peak-to-peak interval)
peak_intervals = np.diff(t[peaks])
average_interval = np.mean(peak_intervals)
heart_rate = 60 / average_interval

# Plot the signal and detected peaks
plt.figure(figsize=(10, 5))
plt.plot(t, ppg_signal_noisy, label='Noisy PPG Signal')
plt.plot(t[peaks], ppg_signal_noisy[peaks], 'rx', label='Detected Peaks')
```

```
plt.title('PPG Signal with Peak Detection')
plt.xlabel('Time (s)')
plt.ylabel('Amplitude')
plt.legend()
plt.grid(True)
plt.show()
```

```
# Display extracted features
average_interval, heart_rate
```

Output :

(average peak-to-peak interval : 0.0660262911918541,
heart rate : 908.7289156626506)

