DSP Course Project

Reproduction and Enhancement of the Pan-Tompkins QRS Detection Algorithm

Objectives

1. Reproduce the classic Pan-Tompkins QRS detection algorithm using real ECG data.
2. Analyze each signal processing stage using DSP tools (frequency response, pole-zero plots, group delay).
3. Implement a modern adaptive thresholding strategy using LMS filtering to improve QRS detection robustness.
4. Evaluate and compare both original and enhanced approaches on a real ECG dataset (MIT-BIH).

Reference Paper

Pan, J., & Tompkins, W. J. (1985). A Real-Time QRS Detection Algorithm.
IEEE Transactions on Biomedical Engineering, Vol. BME-32, No. 3, pp. 230–236.
DOI: 10.1109/TBME.1985.325532

Tools

- Language: Python or MATLAB
- Libraries: NumPy, SciPy, Matplotlib, WFDB (for PhysioNet), or MATLAB's Signal Processing Toolbox
- Dataset: MIT-BIH Arrhythmia Database from PhysioNet

Project Tasks

1. Literature Review & Signal Flow
   - Read and summarize Pan-Tompkins stages: Bandpass filter, Derivative, Squaring, Moving window integration, Thresholding logic
   - Diagram the full signal flow

2. Reproduce the Algorithm
   - Implement each stage as described in the paper
   - Use real ECG signal samples (e.g., MIT-BIH record 100)
   - Detect QRS complexes and annotate them

3. DSP Analysis of Filters
   - For each stage involving filtering (bandpass, derivative, integrator), compute and plot:
     - Magnitude and phase response.
     - Pole-zero plot.

- Group delay.

4. Adaptive Thresholding Using LMS
   - Implement an LMS-based adaptive threshold that learns a dynamic decision. boundary for QRS detection
   - Compare LMS thresholding with the original static or dual thresholds.
5. Evaluation
   - Compare detection performance:
     - Sensitivity, Positive Predictive Value, F1 Score.
   - Evaluate both clean and noisy ECG segments.
   - Compare Pan-Tompkins with LMS-enhanced version.
6. Reporting
   - Document code, graphs, and findings in a paper like format.
   - Discuss strengths, limitations, and possible improvements.

Expected Learning Outcomes

- Deep understanding of QRS detection and ECG signal processing
- Hands-on experience with DSP techniques and ECG datasets
- Practical use of LMS adaptive filtering in biomedical applications
- Ability to analyze and improve real-time signal processing systems

Evaluation Rubric

Component	Points
Literature Review + Signal Flow	10
Correct Implementation	20
Filter Analysis (DSP tools)	20
LMS Adaptive Thresholding	20
Comparative Evaluation	10
Code Structure & Documentation	10
Final Paper & Visualization	10

Optional Extension for interested Students

- Attempt a publishable short paper or poster (e.g., IEEE student branch, conference, or local journal)
- Explore other adaptive methods: RLS, Kalman filter, or ML-based thresholding