Project

This project combines a theoretical component with a hands-on application of signal processing techniques. The first part will deepen your understanding of Fourier-based representations and sampling, while the second invites you to design and analyze a signal processing application of your choice using these tools.

You may work individually or in pairs. Submit a detailed report, a recorded presentation, and your well-documented MATLAB code on Moodle. Equal contribution from team members should be demonstrated. The final grade will reflect the correctness, clarity, and creativity of your work.

1 Fourier-Based Signal Representation

This component focuses on developing a practical understanding of how signals can be represented using Fourier techniques.

1.1 Finite Fourier Series Approximation

Create a function to approximate a time-limited signal using a truncated Fourier series:

$$\hat{x}(t) = \sum_{k=-n}^{n} c_k e^{j2\pi kt/T}, \quad c_k = \frac{1}{T}\int_{-T/2}^{T/2} x(t)e^{-j2\pi kt/T}\, dt.$$

Function signature:

```
function [xhat, ck] = ffs(xt, t, n, T)
```

Test your function on a Gaussian pulse, $x(t) = e^{-t^2}$, and a sine wave defined over a limited interval, $x(t) = \sin(2\pi t)$ for $|t| < 1$, zero elsewhere.

Explore the effects of changing the number of harmonics n and the period T on the approximation accuracy. Visualize the square error:

$$E(n, T) = \int_{-\infty}^{\infty} |x(t) - \hat{x}(t)|^2 dt.$$

1.2 Fourier Transform and Inverse

Implement numerical routines to compute the continuous-time Fourier transform (CTFT) and its inverse:

```
function [f, xf, W] = ftr(t, xt, T)
function [t, xt_rec, T] = iftr(f, xf, W)
```

Verify your implementation using known time and frequency duals.

2 Sampling, Reconstruction, and Aliasing

Now shift to a discrete-time context, analyzing how signals behave under sampling.

2.1 Sampling

Implement a function that samples a continuous signal at a given rate:

```
function [t_sample, x_sample] = sample(t, xt, fs)
```

Apply it to:

$$x(t) = \cos(2\pi f_1 t) + 0.5\cos(2\pi f_2 t), \quad f_1 = 3\text{Hz}, \ f_2 = 7\text{Hz}$$

Try sampling rates $f_s = 0.5f_{\max}, \ f_{\max}, \ 2f_{\max}$ and analyze the result.

2.2 Reconstruction

Implement finite sinc interpolation to reconstruct a signal from its samples:

```
function xrcon = reconstruct(t, x_sample, fs)
```

Study how the reconstruction quality is affected by the sampling rate, truncation of the sinc kernel, and the original signal's bandwidth.

2.3 Aliasing and Ambiguity

Demonstrate aliasing by sampling $x(t) = \cos(2\pi \cdot 5t)$ at $f_s = 8$ Hz. Show that different cosine signals can result in the same sampled values, and discuss how prior knowledge (e.g., bandlimiting) can resolve this ambiguity.

2.4 Noise Robustness

Introduce white Gaussian noise to your sampled signal and examine the impact on reconstruction. Vary the sampling rate and filter parameters to explore robustness and tradeoffs.

2.5 Connecting Fourier Series and Transforms

Show numerically that Fourier series coefficients can be obtained by sampling the Fourier transform over one period. Verify this on one of your example signals using the Fourier and sampling function you developed in this project.

3 Application and Analysis

Apply your developed tools to a real-world signal of your choice, such as recorded speech, an ECG signal, music, temperature data, seismic data, or any other signal. First, extract a meaningful time-domain segment from the signal and visualize it. Next, analyze the frequency content of the signal using the Fourier transform. Then, sample the signal and reconstruct it using your code. Assess any aliasing effects and reconstruction errors. Finally, reflect on how these phenomena influence system design decisions, such as the selection of sampling rate and bandwidth.

3.1 Filtering and System Design

Design a low-pass FIR filter (e.g., via windowed sinc function) to remove high-frequency noise or irrelevant components from the signal. Implement the filter using convolution and analyze its impulse and frequency responses. Justify the choice of cutoff frequency based on your spectral analysis. After filtering, compare the input and output signals in both time and frequency domains.

Use the Z-transform to derive the system function $H(z)$ and comment on the stability and causality of the filter.

3.2 Sampling and Reconstruction with Filtered Signal

Sample both the original and filtered signals at various rates, and then reconstruct them using sinc interpolation. Plot and compare the results, quantifying errors and observing the effects of aliasing and reconstruction fidelity.

3.3 Error and Robustness Analysis

Add white Gaussian noise to both the filtered and sampled signals and attempt to recover the original signal post-reconstruction. Discuss the influence of noise on signal fidelity.

3.4 Design Insights and Discussion

Summarize your findings, including insights into signal characteristics, filtering effectiveness, sampling constraints, and system design. Discuss broader implications in fields like audio, biomedical, or communications systems.

Submission Guidelines

Submit a compressed archive that includes a PDF report, your MATLAB code with comments and modular structure, one test case for each function, a README file explaining how to run the code, and a brief contribution statement from each member. Additionally, a recorded presentation of 10–15 minutes, including a demo, is required and should be uploaded to Moodle along with the other materials in the .zip folder.

Presentation Guidelines

Your grade will depend not only on the correctness of your implementation but also on the clarity of your presentation.

Report

Your report should be well-organized and readable. Ensure the following:

1. **Content:** Support all claims and ensure your narrative flows logically from the problem to your proposed design.
2. **Organization:** Structure the report with consistent headings and logical sections.
3. **Style:** Use concise, precise, and professional language, maintaining a human tone—avoid sounding overly mechanical.
4. **Figures and Tables:** Include visuals that support your arguments, ensuring all are clearly labeled and referenced.
5. **Formatting:** Maintain consistent formatting throughout, and cite all references in a standard citation style.

Recorded Presentation

You are required to upload a short recorded presentation explaining your work. To improve quality, consider the following:

1. **Slide Titles:** Use slide titles that clearly communicate key ideas or claims.
2. **Visual Evidence:** Include graphs, diagrams, and equations to support your claims.
3. **Slide Text:** Limit text and minimize bullet points. Aim for no more than one slide per minute.
4. **Images and Layout:** Use high-quality, consistent visuals with clear labels. Choose high-contrast color combinations and avoid cluttered slides.
5. **Speech Delivery:** Speak naturally with a conversational tone, varied pacing, and clear articulation. Avoid reading directly from the slides.