Team members : Ping He, Shuangling Zhao, Tong Li

GitHub Link: https://github.com/PingHe87/AudioLabSwift

If you made the FFT Magnitude Buffer a larger array, would your program still work properly? If yes, why? If not, what would you need to change?

When increasing the FFT buffer size, we believe some modifications are necessary to ensure the program runs smoothly.

For example, we need to make sure the FFT implementation supports the larger buffer size. We also need to update all parts related to BUFFER_SIZE, such as adjusting the size of timeData and fftData to match the new BUFFER_SIZE. Additionally, the number of points displayed on the graph should be adjusted according to the new buffer size.

Is pausing the audioManager object better than deallocating it when the view has disappeared (explain your reasoning)?

This depends on the app's use case. If the app needs to be accessed multiple times in a short period, we believe pausing the audioManager is usually better than releasing memory. Pausing keeps the playback data in memory, avoiding the overhead of reallocating and reinitializing resources when the view reappears, especially for large buffers like FFT or audio data. If you release the memory, you'll need to reinitialize and allocate everything again, which requires a lot of computation. Pausing also allows for quicker recovery when the view returns.

However, if the app is using a lot of memory for audio resources and it’s unlikely the app will be accessed again soon, releasing memory might be the better option.