

Real Time Music Vibrato Visualization

Minhao Zhang, Xinzhao Liu
Electrical Engineering University of Rochester

Real Time Audio I/O Unit

This unit gets audio data from microphone. When the amount of data reaches the size of the block, it stores it into the shared memory. It also can play back the sound by reading the shared memory or directly transfer from input data.

Reason to choose Portaudio:

PortAudio
Portable Cross-platform Audio I/O

it's across platform, can be used in all operating systems. Also we can apply sample by sample processing, which can give us a really short delay

Parameters:
Block Size: 1024
FS: frequency:44100Hz

FFT

Data Processing Unit

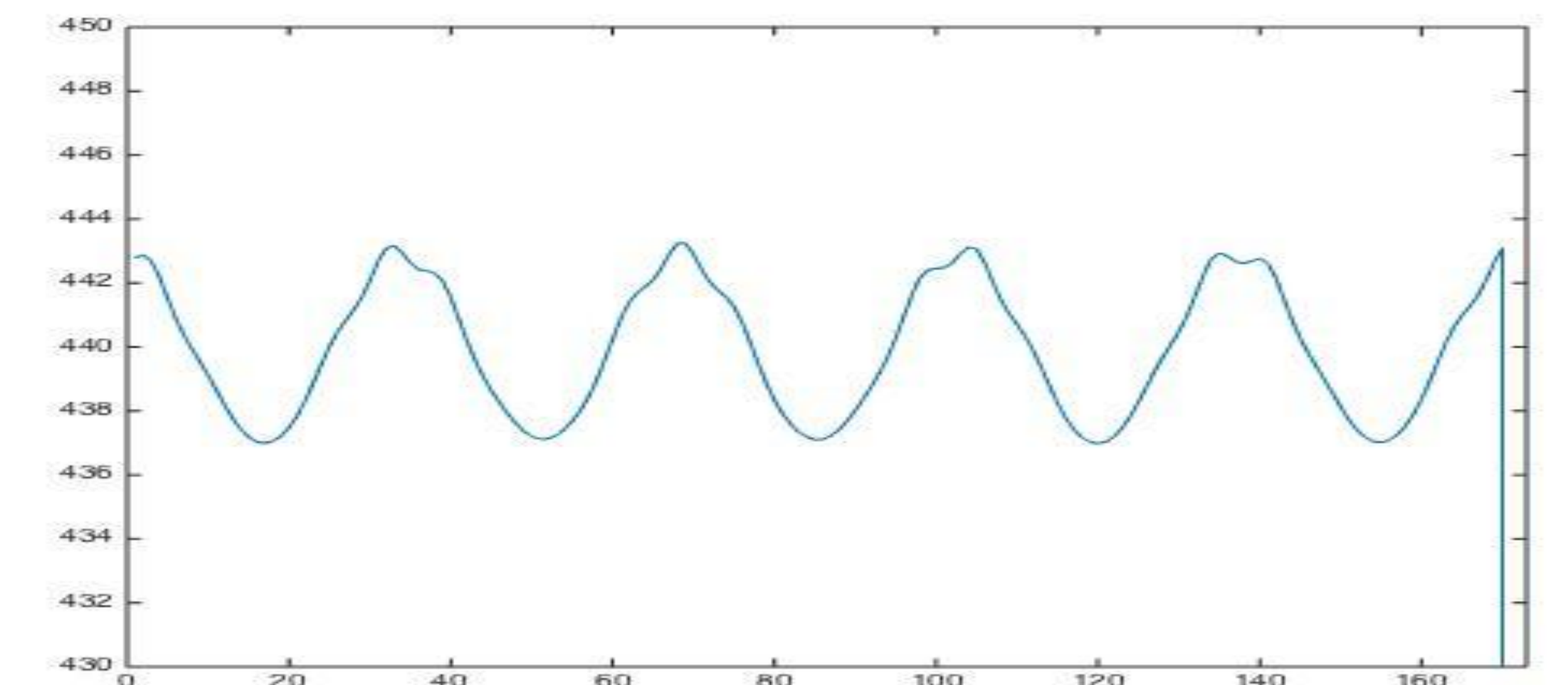
This part is the core algorithm part, which detects the vibrato in music. The algorithm main detects AM and FM in music. The main Process include several step: 1. gets one block of data from the shared memory. 2. Filtering and smoothing. 3. Apply FFT to get frequency domain information. 4. Do the time frequency analysis get AM and FM information.

Time frequency analysis to calculate the AM, FM

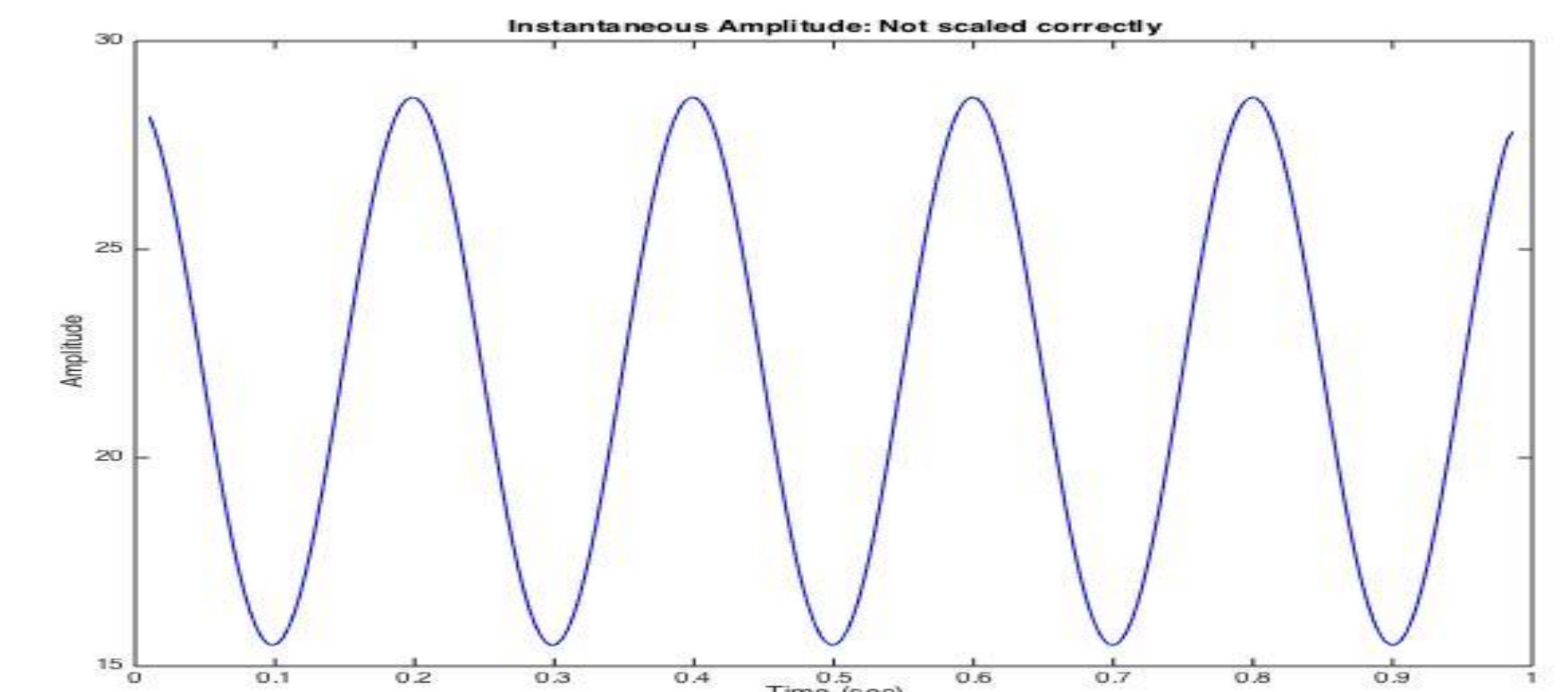
In frequency domain, for each block of data, treat it like one time frame data after STFT. Then do peak picking, store the Amplitude of the first reasonable peak as fundamental peak. Also store this frame's phase data. When next frame comes, calculate the unwrapped phase difference then divided by time to get the instantaneous frequency, which is FM we need to detect.

Some other methods:

Parabola fitting: May not be accurate enough
YIN: Very accurate but slow not a good choice for real time analysis



FM vs Time



AM vs Time

Shared Memory to store data from microphone

Data after processing mainly contain AM, FM

Audio Info data, converted to OpenGL rendering data

Future Work

1. Implement vibrato detecting in C
2. Better visualization, more like a game
3. Testing with real vibrato sound
4. robustness

Graphic Visualization Unit

Using OpenGL, The graphic rendering can be very fancy. Currently it just show us a spectrum. The spectrum can spin it's also easy to make it 3D version.

Why OpenGL:

To serve the future goal, making it more like a audio game. Plus OpenGL is cross-platform.

