Guitar Tuner for Symbian OS

Students:
Reut Ochayon
Gilad Cohen

Supervisors:
Yevgeni Litvin
Yoav Shechtman
Outline

- Motivation
- Project goal
- Requirements
- General architecture
- HPS Pitch detection algorithm
- Live demonstration
Motivation

- Why tune a guitar?
- What existed before?
- Why do we need something better?
Motivation

- Modern cellphones are strong enough to run a tuner program

- This implementation has major advantages over the rest
  - No need to carry a special device
  - Easily updated
Project goal

- To create a "Guitar Tuner" application for symbian OS
  - Symbian is an OS designed for mobile devices, it is very commonly used today
• 8-48KHz sampling rates, we will work with 8KHz

• The desired pitch range is [C₂, ~E₅] – frequency range is [65,700] Hz
Requirements

- Minimal frequency resolution: 25 cents
  \( \approx 1 \text{ Hz} \) around the low frequencies

- RealTime processing

- Fixed point arithmetic
General architecture

Algorithm
Harmonic Signal Structure

- Fundamental frequency \( (f_0) \) and its products
- Different energies
Harmonic Product Spectrum

\[ x(n) \xrightarrow{\text{FFT}} \text{LP filter} \xrightarrow{\downarrow 2, \downarrow 3, \downarrow 4, \ldots, \downarrow M} \text{Max} \xrightarrow{f_0} \]

\[ M - \text{parameter} \]

* Patricio de la Cuadra, Aaron Master, Craig Sapp. Efficient Pitch Detection Techniques for Interactive Music. [Center for Computer Research in Music and Acoustics, Stanford University]
Harmonic Product Spectrum
Implementation Problems & Solutions

- Example: FFT algorithm makes redundant calculations
  - Solution: using FFT optimized for real signals
Implementation Problems & Solutions

- Example: low frequency resolution
- Solution: use values of higher harmonies
$f_0 = 110.8$ Hz
The program is implemented using Carbide C++ environment for Symbian.

Run on Nokia cellphone emulator, and Nokia N95 cellphone purchased for the project.
Nokia N95 specifications:
- Operating System: Symbian OS v9.2
- CPU Type: ARM 11 (clock rate: 332MHz)
- Free Executable RAM Memory: 18 MB
- Audio sampling rates: 8 – 48 KHz
- Has on-device debug capability
Conclusions

- Project goal was achieved
- All requirements were met
- The application could be used freely
- Good feedbacks
Live Demonstration
FFT algorithm: Real signal optimization

Original signal:

\[ u[n] \in \mathbb{R} \quad n \in [0, N - 1] \]

Define \( x[n] \):

\[ x[m] = u_e[m] + j * u_o[m] \]

\[ \mathbb{F}\{x[m]\} = X[k] = U_e[k] + j * U_o[k] \]

According to classic FFT:

\[ U[k] = U_e[k] + \mathbf{W}_N^{-k} U_o[k] \]