Music Coach

Real-time Evaluation of Music Performance using Nokia N900

CS290I Fall 2009 Group Project
David Johnson
Dianna Han
Overview

- **Music Coach**: An mobile phone application that
  - Checks if the user is playing a note at the correct pitch
  - Checks if the user is playing a note at the correct timing
  - Compares the user’s musical performance to a pre-loaded score
  - Controls the tempo of the performance by shaking/rocking the phone in a rhythmic manner

- Other applications focus on Karaoke and tuning identification

- Ours aims more at serious learners who needs to evaluate their performance on real instruments
Main Technical Challenges

- Different pitch ranges for different instruments (logarithmic scale problem)
- Real-time frequency analysis of a pitch
- Extracting tempo from rhythmic phone movements (especially with slow movements)
- Fuzzy boundaries for evaluation (slight imperfections should be tolerated)
Pitch Range
System Architecture
Pitch Detection

[Diagram showing the process of pitch detection with steps: Sample window, FFT, Pitch decision, and output pitch.]

[Graph illustrating the time domain withbuffer lengths and FFT positions.]
Pitch Detection: Accuracy/Resolution

- The pitch spacing between adjacent notes is logarithmic rather than linear.
- Frequency resolution also depends on the number of samples in a sampling window.
Pitch Detection: Sample Rate

- Determined by
  - The shortest note duration expected in the performance
  - The lowest expected frequency

- In Music Coach, 20Hz is used.
  - R = sample rate (Hz)
  - N = number of samples in the time window
  - T = N/R (period of time window)
  - F = R/N (frequency resolution of spectrum analysis)
System Processing Time

- Determined by:
  - Computational overhead of the application
  - Processing capability of the device

- Computational Load
  - Mainly introduced by pitch detection (FFT thread)
  - Minimized on tempo detection

- Overall, the processing time should not be more than 3ms for a 50ms-sample of audio data
  - In this case feedback will be delivered to the user 53ms after the note started playing
Metronome

- Tempo can be either pre-set or controlled by accelerometer
- Metronome provides both visual and audio indicators
- It doesn’t interfere with the instrument recording because the metronome ticks are generated at high frequencies (6000Hz and 4500Hz)
Tempo Detection using Accelerometer Readings

- Peak Detection in Digital Signals
  - Fast
  - Robust
- Algorithms
  - Significant Changes
Future Work

- GUI Design: Get user feedback from musicians.
- Improved Audio Isolation with Bluetooth headset.
- Do note timing in the time domain.
- Make use of professional musical type setting libraries, e.g. Guido.
- Display-Free Feedback using Buzzer
- Bird song recognition
GUI Design & Demo

[Diagram showing a musical software interface with various controls and displays, including:
- METRONOME DISPLAY
- NOTE DISPLAY FOR PLAY AND RECORD
- ACTIVATE ACCELEROMETER
- FREQUENCY INDICATOR
- PITCH INDICATOR
- RHYTHM INDICATOR
- METRONOME CONTROL (Tempo: 60, options for Detect Tempo and Play Clicks)
- Other interface elements such as buttons for play, pause, and a slider for threshold level]

Xephyr on :2.0 (ctrl+shift grabs mouse and keyboard)