Project Proposal

EECS 452 Winter 2012

Title of Project:
Digital Soloer

Team members:
Iman Aboutaleb (aaboutal), Ahmad Aldabbagh (aldabbag), Arya Bandari (siavash), Mohammed Sarraj (msarraj), Giovanni Zhang (huade).

1. Introduction and overview of project

The movie industry has come to us wanting to re-master old movies but needing the vocals and other audio separated. The music industry has come to us wanting a way to sample separate instruments and sounds without having the original tracks. Our team is going to create a system to solo tracks in audio recordings.

Remixing and re-mastering almost always require the original separated audio tracks in a recording, but many times these are unavailable. This project solves the classical and notorious source separation problem in music. The problems come from the fundamental frequency of a note being played having many harmonics that might overlap with other fundamentals and harmonics of other notes. We will use a variety of techniques to isolate the instruments to help artists in their own projects.

We will create a system by April 2012 that will separate the different instruments in an audio recording. We are still learning how automatic and generalized we can make the algorithm. For the final report, we will test the system's ability to accurately separate a source. We will show at the expo that our system works by inputting other people's music and separating the track.

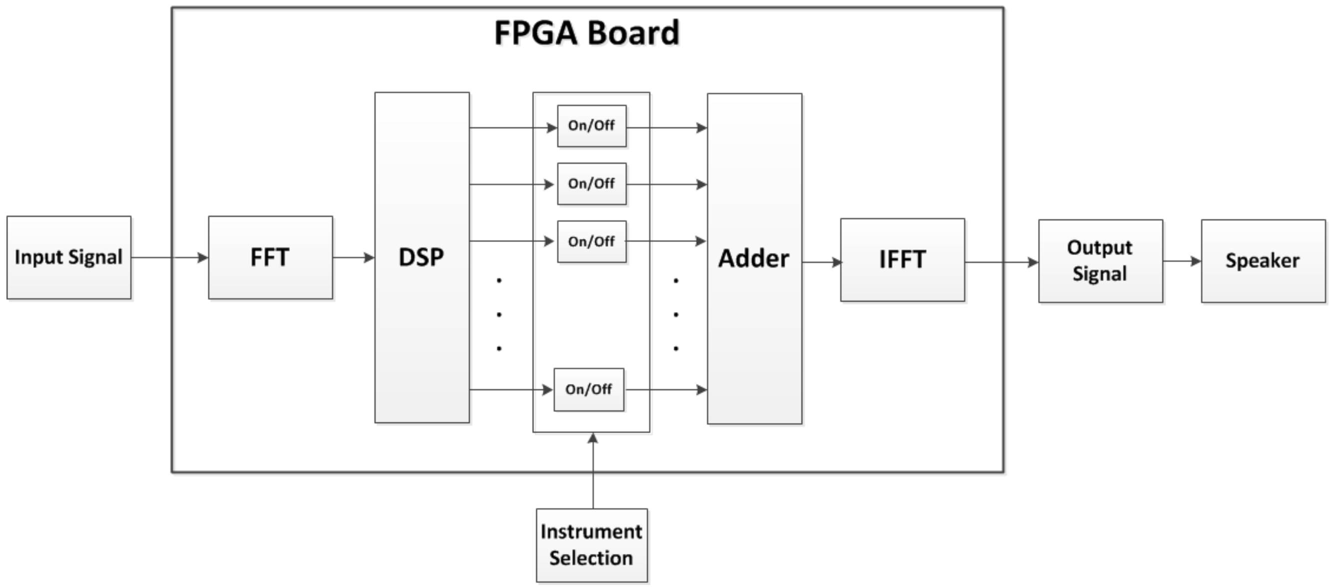
2. Description of project

1. **Goals:**
   - Create the system that can take a song and output its separate tracks.
   - As accurate and automatic as possible.

2. **System concepts:**
   Create frequency characteristics of instruments using sample database [2]
   Estimate number and types of sources (might be inputs) [1]
   Estimate tempo and note length (might be inputs)
   Pattern match frequency characteristics to STFT [1,3]

3. **Feasibility:**
    - When will the system work the best?
    - What techniques should we use?
    - How automatic can the algorithm be?

4. **System architecture:**
    High level system architecture:



The DSP block has one input and several outputs, each designated to a particular instrument. It includes all the system concepts noted above and will be done on the FPGA board.

5. **Problem prediction:**

Problem	**Threat-Source/ Vulnerability**	**Existing Controls**	**Likelihood**	**Impact/Risk Rating**	**Recommended Controls**
Audio track not decomposed properly	Lack of time and resources / Too difficult to implement	Implement method in a Matlab simulation	Medium	High	Choose a track that only has a few very distinct instrument sounds
Decomposed audio samples do not sound like the original track	Filtered out part of sample due to frequency overlap of the other instruments	Show in a simulation that the only parts filtered out were the parts that overlapped	Medium	Medium	Use a track with known instrument samples to compare to result

Decomposed audio samples have lower volume than original track	The filter decreased the magnitude of the volume when decomposing the track	Manually increase the volume of the audio output before playing the sample	Medium	Low	Increase the magnitude of the signal after the other instrument sounds are filtered out

6. **Preliminary parts list:**
   - C5515 eZDSP stick.
   - DE2-70 FPGA board.
   - Cables and wires.
   - Computer software (i.e. MATLAB).
   - Electronic Keyboard.

3. Milestones

1. **Milestone 1 (achievable by March 15):**
   Successful Matlab simulation of the audio decomposer
   To reach this milestone, we will confirm that we are able to decompose a track made by one of the group members. We will have known samples of each instrument used in the track and a final composition of the entire track. We will use these samples to compare our decompositions of the track with the original samples. Our comparison can be made by listening to the sounds of the original samples with our decomposed samples. We will also compare the plots of the frequency spectrum in Matlab.

2. **Milestone 2 (achievable by April 3):**
   Successfully working hardware system of the audio decomposer.
   To reach this milestone, we will design a circuit board that will implement our algorithm in hardware. The main components of the circuit board will include a microprocessor and and an FPGA with audio output. We will program the microprocessor to handle the decomposition techniques of our designed filter. The result of our system will be outputted to the audio jack, which will be connected to a headphone set to allow the user to hear the instrument solo sample. We will compare our decomposed sample to a known sample used to compose the original track.

3. **Challenges**
   One thing that could cause us to not achieve Milestone 1 is that if we are not able to decompose all of the instrument samples from the original track.
   One thing that could cause us to not achieve Milestone 2 is that the decomposed audio instrument samples do not sound like the original track.

4. Contributions of each member of team

(idea: 1 person on Instrument analysis, 2 people focus on Verilog/C/Matlab programming, 1 person helping in programming and Optimization, 1 person on Hardware development)

Each member of our team has the same role in the current stage of the project, which is learning the basic algorithms and concepts related to the topic addressed. The roles should change as we progress in the project.

- Iman Aboutaleb:
- Ahmad Aldabbagh: Hardware development.
- Arya Bandari: Instrument analysis and methods/design
- Mohammed Sarraj: Programming Optimization (Matlab/C/Verilog)
- Giovanni Zhang: Programming (Matlab/C/Verilog)

We meet on Sundays at 12:00 and Tuesdays at 4:00. For contact, we have a UMich directory e-mail (eecs452soloer@umich.edu) and a Google group (eecs452soloer@googlegroups.com). We also share documents using Google docs.

5. References and citations

1. *Multiple fundamental frequency estimation and polyphony inference of polyphonic music signals*; Yeh, C. and Roebel, A. and Rodet, X.; Audio, Speech, and Language Processing, IEEE Transactions on Vol. 18 num. 6 pgs. 1116-1126; 2010

2. *Purging Musical Instrument Sample Databases Using Automatic Musical Instrument Recognition Methods*;Livshin, A. and Rodet, X. Audio, Speech, and Language Processing, IEEE Transactions on Vol. 17 num. 5 pgs. 1046-1051; 2009

3. *Real-time polyphonic music transcription with non-negative matrix factorization and beta-divergence*; Dessein, Arnaud and Cont, Arshia and Lemaitre, Guillaume; Proceedings of the 11th International Society for Music Information Retrieval Conference pgs. 489-494; August, 2010