The Manipulation of Digital Signal Sound Files with MATLAB – a MATLAB Synthesizer

Kireet Kokala, Matthew Briethaupt, David Anderson. DSP CSc 6270, 5-4-2009
Dr. Michael C. Weeks, Georgia State University

ABSTRACT

In early 2009, the creation and manipulation of digital sound files is ubiquitous. Musical recordings frequently incorporate synthesizers into their instrumentation. The home theater systems for sale in local electronics stores boast of the most recent technologies which employ digital signal processing to enhance the sound tracks of DVD movie recordings. As students of digital signal processing, we proposed to use MATLAB to investigate how some of these manipulations can be accomplished. This paper covers our projects details of recording music from instruments and comparing them with Synthesized digital files. The manipulation of the files via pitch-shifting in MATLAB and addition manipulation and representation via a MATLAB-based equalizer are also controlled. The project's successes and shortcomings are presented in the discussion with appropriate references to literature.

1. INTRODUCTION

David, an amateur violinist, is intrigued with the qualities of a pitch that gives a violin its unique sound. Why does a note played on the violin sound clearly different from the same note played on a trumpet, or piano? If these qualities can be identified, and more importantly, quantified, then we should be able to emulate the tonal qualities of a violin using MATLAB by manipulating a digitally produced frequency, for instance, A 440. In the creation of our synthesizer, David will focus his efforts on digitally reproducing the natural sound of the violin. As part of this effort, he will record notes played on the actual instrument and apply filters to the digital recording using MATLAB in order to isolate and analyze the various components of the sound. Using this data and with information gathered by researching literature on violin acoustics, he will lead the group's attempt to have the synthesizer emulate the sound of the instrument. We expect this will involve manipulating a digitally-created fundamental frequency by adding in various harmonic frequencies. It may possibly involve using other signal manipulations as well (to be researched), to more precisely generate a wave pattern which will match the violin's acoustics. To measure our success, we will use MATLAB to quantify mathematically the correlation between the original recorded tone and the synthesized tone. Of course, the true test will be whether the synthesized tone truly sounds like the real instrument.

Matthew, also a musician, is interested in applying various digital effects to music. Using MATLAB, the group will attempt to add echoes, reverse echo effects, and arpeggiation (multiple pitch shifted delays) to recordings, then analyze the results. Since the synthesizer will generate notes of different target frequencies, it makes sense in the context of this project for a pitch shifter to simply instruct the synthesizer to generate the additional frequencies to harmonize with a melody. On the other hand, a harmonizer that works on a raw digitized audio signal must perform a much more computationally intensive task, which can also result in some undesirable artifacts, such as additional frequencies, distortions or noise. Matthew intends to compare the quality of synthesizer generated harmonies against those generated by pitch-shifting a raw signal. The generation of harmonies from raw signals can be done by using MATLAB, as well as a physical effect unit owned by Matthew, which can perform fixed interval pitch shifting as well as Intelligent Pitch Shifting (IPS). IPS generates harmonies that are always musically correct in a given musical key. The principle behind IPS is that in order to stay within a musical key, the amount of the pitch shift has to vary according to the original note. For example, in the key of C major, the 3rd note is E, which is 2 whole tones above C. If the original note is D, the harmonized note would be an F, which is only 1½ tones higher than D. If we used a constant interval of two whole tones, D would be harmonized with an F# which is not in the key of C major.

In this project, musically correct harmonies can be generated easily because the melody and key will be known in advance, so we can once again compare the synthesized result against using an Intelligent Pitch Shifter on the raw output.

Kireet's efforts will focus on building an equalizer in MATLAB to balance the inputs from various instruments in a recording. The equalizer will receive various inputs; namely, frequencies, which will be targeted to improve or distort the sound. Up to 6 bands (such as treble and bass) will be available in order to increase/decrease volume, noise, or intensity of the input frequencies. The equalizer will feature graphic sliders for the bands and will indicate appropriate decibel ranges. Time permitting, custom presets such as "Rock", "Dance", and "Metal" may be introduced.

The group proposes to work together to research and implement in MATLAB techniques to accomplish these manipulations.

2. MAJOR SECTIONS

Part I

Part II

Part III

3. RESULTS

4. DISCUSSION

Discussion section

5. REFERENCES

[1] John Strawn, *Digital Audio Signal Processing*, A-R Editions, Inc., 1985

[2] Benesty and Huang, *Adaptive Signal Processing*, Springer, 2003.

[3] John Ackenhusen, *Real-time Signal Processing*, Prentice Hall, 1999.

[4] C. H. Chen, *Digital Waveform Processing and Recognition*, CRC Press, 1983.

[5] Thomas G. Hagan, Robert Treiber, "Hybrid analog/digital techniques for signal processing applications," *ACM Proceedings of the Spring joint computer conference April 26-28*, 1966, pages 379-388.

[6] Rahul Singh, Zhao Li, Pilho Kim, Derik Pack, Ramesh Jain, "Event-based modeling and processing of digital media," *ACM International Conference Proceeding Series,* vol. 66 2004, pages:

19 - 26.

[7] Stefan Müller, Simon Schubiger-Banz, Matthias Specht, "A real-time multimedia composition layer," ACM, 2006, Proceedings of the 1st ACM workshop on Audio and music computing multimedia pages: 97 - 106.

[8] Udo Zölzer, Xavier Amatriain, Daniel Arfib, *DAFX: Digital Audio Effects*, John Wiley and Sons, 2002.

[9] Perry R. Cook, *Real Sound Synthesis for Interactive Applications*, A K Peters, Ltd., 2002.

[10] Emphasizes physical modeling of sound and focuses on real-world interactive sound effects.

[11] Philippe Guillaume, *Music and Acoustics: From Instrument to Computer*, ISTE Ltd, 2006.

[12] James W. Beauchamp, *Analysis, Synthesis, and Perception of Music Sounds: The Sound of Music*, Springer Science+Business Media, LLC, 2007.

[13] John M. Chowning, John M Grey, James A Moorer, and Loren Rush, *Computer Simulation of Music Instrument Tones in Reverberant Environments,* Department of Music, Stanford University, 1974.

[14] Xavier Serra (2003). Spectral-Modeling-Synthesis-Past-and-Present.pdf "*Spectral Modeling Synthesis: Past and Present"* (PDF). 20. Retrieved on April 19, 2008.

[15] Purwins, H., Herrera, P., Grachten, M., Hazan, A., Marxer, R., Serra, X., *Computational Models of Music Perception and Cognition I: The Perceptual and Cognitive Processing Chain*. Physics of Life Reviews, 2008.

[16] McIntyre, M.E., and Woodhouse, J. "The acoustics of stringed musical instruments", *Interdisciplinary Science Reviews,* Vol.3, No.2, 1978, pages 157-173.

[17] McIntyre, M.E., Schumacher, R.T., and Woodhouse, J., "On the oscillations of musical instruments," *Journal of the Acoustical Society of America,* Vol. 74, No. 5, November 1983, pages438-451.