SPECTRALLY MATCHED CLICK SYNTHESIS

Matthew Wright
CREATE and MAT, U.C. Santa Barbara
Santa Barbara, CA USA
matt@create.ucsb.edu

Matt Stabile
CREATE and MAT, U.C. Santa Barbara
Santa Barbara, CA USA
mattstabile@gmail.com

ABSTRACT

We introduce Spectrally Matched Click synthesis as a novel application of FIR filter design allowing the creation of arbitrarily short duration clicks whose magnitude frequency spectra approximate those of arbitrary input sounds. We demonstrate its use on effects including incremental attack strength modification and continuous gradual "morphing" between any input sound and successively more impulsive/percussive sounds.

1. INTRODUCTION

The core idea of Spectrally Matched Click (SMC) Synthesis is simple: design FIR filters (of arbitrary order) whose magnitude frequency responses approximate the magnitude frequency of an arbitrary input sound, and then treat the resulting impulse responses as sampled sounds to be played on their own or mixed with the original sound. Equivalently, after we design our FIR filters, we use them to filter carefully placed impulses[1] rather than "real" sound. Table 1 shows how SMC synthesis is an application of FIR filter design.

Table 1: *Correspondence between spectrally matched click synthesis and FIR filter design.*

SMC Synthesis	FIR Filter Design
Input sound's spectrum	Desired nonparametric magnitude frequency response
SMC duration	FIR filter order
Outputted SMC	Outputted filter impulse response
Play the SMC as a sampled sound	Play an impulse into the filter

SMC synthesis can also be viewed as an extreme case of a source-filter model: the "source" is an ideal digital impulse and the filter approximates the magnitude frequency spectrum of the input.

Another interpretation sees SMC synthesis in terms of lossy data compression. The input sound has a certain magnitude spectrum and duration; the output sound has a shorter duration (i.e., less data) while as much as possible retaining the same spectrum or the same sounding spectrum. In the frequency domain the input spectrum has a certain level of detail and we remove data by reducing the level of spectral detail. Of course SMC synthesis is not a general-purpose perceptual audio coder; the point of compression in our case is to generate interesting new sounds.

[1] Throughout this paper, "impulse" refers to an ideal digital impulse: a single "1" sample within an infinite stream of zeros.

The main application of this family of techniques is to vary the degree of attack strength of an arbitrary input sound, which we could refer to as the *percussiveness* or *attackiness*. As early as 1979 Wessel pointed out that sharpness of attack is one of the main perceptual dimensions of musical timbre and suggested that "both the fine tuning of rhythm in music and psychoacoustic research will benefit greatly if the control software of our synthesis systems allows easy and flexible adjustment of [attack characteristics] in complex musical contexts" [8]. Almost every sound synthesis system does indeed provide easy and flexible adjustment of sharpness of attack *for synthetic sound* through features such as amplitude envelopes. Our SMC-based methods provide control of sharpness of attack for any arbitrary sampled sound, not just synthesized sound.

2. FIR FILTER DESIGN

Finite impulse response (FIR) filter design is a rich and well-established field and there are many techniques for producing FIR filters from various specifications [2,4,6,9]. Our only contribution to this field is to discover new applications for these techniques.

Most FIR filter design methods are made to produce specific "classical" filter types such as low-pass and take input parameters such as cutoff frequency, allowable pass-band ripple, etc. Such methods are not useful for SMC, where we need to specify an arbitrary sampled shape for the desired magnitude frequency response.

For SMC synthesis, the format of the filter design specification is a sampled representation of the desired magnitude frequency response, derived from the magnitude of the fast Fourier transform (FFT) of the sampled input sound. We currently zero-pad the input sound buffer so the length of the FFT input is twice the nearest power of two above the length of the input file. At this point we optionally perform critical band smoothing as described in section 2.2.

We next downsample the large array of magnitude values to a smaller set of values more suitable for calculation of the FIR filter kernel, as shown in equation (1).

$$y[i] = \frac{1}{M} \sum_{j=0}^{M-1} x[iM + j] \quad (1)$$

We choose the downsampling factor M by dividing the size of the input FFT by the desired size of the forthcoming IFFT and rounding to the next lowest integer. Each set of samples within the current step are averaged and stored in the correct position of the smaller downsampled array. Now we have a length L magni-

tude spectrum (where L is M times smaller than the input spectrum) expressing the desired magnitude spectrum of the SMC. L may still be longer than the filter order.

Before taking the inverse FFT, we must convert our magnitude spectrum into a full complex frequency spectrum by specifying the desired phase. At this point it is easiest to design a linear phase FIR filter kernel, leaving the option of later converting to minimum phase as described in section 2.1. Linear phase is achieved by setting all phase components to zero.[2] We then convert to rectangular coordinates and take the iFFT.

The resultant time-domain filter impulse response is multiplied by a Blackman window the size of the user-defined filter order. This window is centered over the middle of the symmetric impulse response. Any samples beyond the range of the filter order are zeroed out and the kernel is shifted to be linear phase.

The resultant array contains an impulse response with a spectrum approximately equal to the spectrum of the input sound. Figures 1 and 2 compare the spectrum of a piano tone with the spectra of SMCs of length 128 and 512 samples (about 2.9ms and 11.6ms) respectively. We see that the shorter click is able to follow the general spectral contour of the piano, whereas the longer click matches the piano's spectrum quite faithfully except in the extreme high frequencies.

By design, this FIR filter design procedure always returns a linear phase filter, in other words, a filter with a symmetric impulse response. In the SMC context there is no advantage to having the filter be linear-phase, but the time-domain symmetry of these sounds is audible when the duration is about 10ms or longer, manifesting qualitatively as a clear fade in and fade out around a central point.

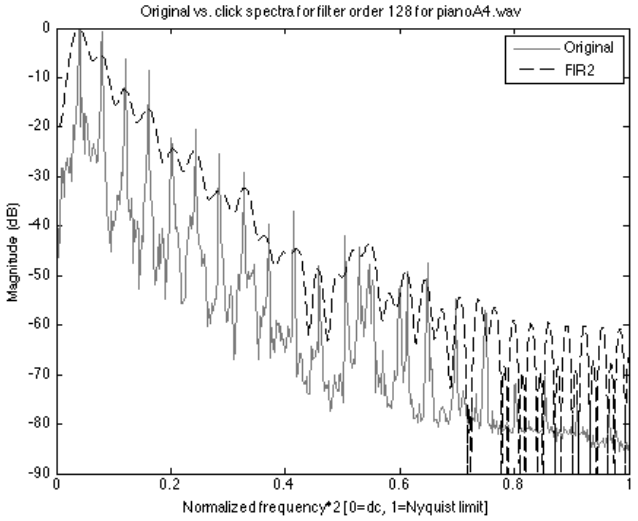

Figure 1: *Comparison of magnitude spectra of original piano tone (solid grey) and length-128 SMC (dashed black).*

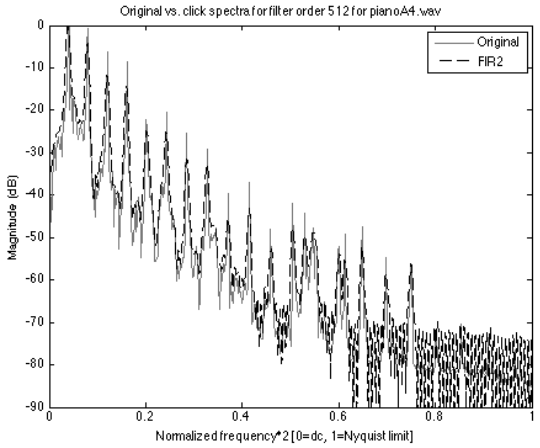

Figure 2: *Comparison of magnitude spectra of original piano tone (solid grey) and length-512 SMC (dashed black).*

2.1. Minimum Phase

We can efficiently convert any signal to its *minimum phase* version in the FFT domain by reflecting all of its poles inside the unit circle, in other words, "by computing the cepstrum and converting anti-causal exponentials to causal exponentials" [5]. The result has the same magnitude frequency spectrum as the original but with the phase spectrum altered so as to maximally concentrate the energy towards the onset of the sound [5].[3] It is important to use sufficient zero padding to produce enough frequency resolution to avoid time aliasing (but see section 3.3).

Without doing any FIR filter design, one transformation of an input sound is simply to convert it to minimum phase. The result is the theoretically most percussive sound with the identical spectral shape.[4] It also has the same duration as the original and in general does not sound like a click, but rather like a more percussive version of the input.

Converting an FIR filter's impulse response to minimum phase results in a minimum phase filter; for purposes of SMC synthesis this has the same effect of increasing the SMC's attackiness.

After converting the filter to minimum phase, there is often very little energy in the later portion of the impulse response, and so often the SMC may be further truncated by brute rectangular windowing with no audible consequences.

2.2. Critical Band Smoothing

A good method for shortening the eventual impulse response (no matter what filter design strategy is used) is to apply *critical band smoothing* to this spectrum as a pre-processing step before filter design (Smith 1982, 1983). In general any form of smoothing that removes fine detail from the desired magnitude frequency response of the filter will tend to reduce the order of the

[2] This actually designs a zero phase impulse response symmetric around a sample index of zero. To avoid the use of negative indices and ensure causality, the impulse response is made linear phase by time-shifting the kernel to use only positive numbered indices [7].

[3] See http://ccrma.stanford.edu/~jos/filters/Minimum_Phase_Means_Fastest.html

[4] To be precise, here "most percussive" means the maximum concentration of energy near the beginning, i.e., the fastest decay.

filter and hence the duration of the click. Critical-band smoothing is a specific instance in which the spectrum is smoothed with a moving average filter whose width is in perceptual units of *Equal Rectangular Bandwidths* [1].

In other words, instead of the moving average always encompassing a fixed bandwith in Hertz, the width of the moving average adapts in a nonlinear way matched to human auditory perception, so that a wider range of frequencies are averaged together in the high frequencies where the perceptual effect of smoothing by a fixed linear frequency bandwidth is less audible.

From the inputted discrete spectrum $X(\mathrm{i})$ we can compute a smoothed power spectrum $S(\omega)$ for any given bandwidth β ERBs as follows:

$$S(\omega,\beta)=\frac{1}{nbins}\sum_{i=bin(f(b(\omega)-\beta/2))}^{bin(f(b(\omega)+\beta/2))}\left|X(i)\right|^2 \qquad (2)$$

where bin(f) gives the FFT bin number of the bin that spans the frequency f Hertz and *nbins* is the number of terms in the summation. The frequency f Hertz corresponds to critical band b ERBs as follows:

$$b(f)=21.4\log_{10}(4.37f+1.0) \qquad (3)$$
$$f(b)=[10^{b/21.4}-1.0]/4.37 \qquad (4)$$

2.3. Summary of Data Reduction Steps

We start with the spectrum of the zero-padded input sound, whose size in bins will be more than twice the duration in samples of the input. The final resulting SMC will have a user-specifiable duration, which will be the order of the FIR filter. Along the way we perform the following data reduction steps:

1. Selecting the input data in the time domain: To generate an SMC to fuse with the attack of a 10-second piano tone, we may get better results by matching the spectrum of the attack portion of the tone than by using the entire 10 seconds.
2. Critical band smoothing removes detail from the spectrum and can also reduce the amount of data if we choose to sample $S(\omega)$ with lower frequency resolution.
3. Downsampling the magnitude frequency spectrum.
4. Time-domain Blackman windowing the IFFT result
5. Time-domain rectangular windowing the filter impulse response after conversion to minimum phase.

3. APPLICATIONS

SMC synthesis was discovered in the course of trying to find the best reference sounds for measuring the perceptual attack time (PAT) of arbitrary input sounds [10]; it provided a continuous tradeoff between matching the spectral envelope of the input sound (so as to be perceived in the same auditory stream with it) and having a distinct impulsive attack (so as to minimize the uncertainty in the PAT of the reference sound).

The ability of SMC synthesis to produce arbitrarily short and impulsive sounds matching any input sound has applications beyond the production of reference sounds for PAT measurement experiments, as described in the following subsections. Several sound examples are available online.[5]

3.1. Giving a Sound a Stronger Attack

First of all, simply converting any signal to minimum phase makes it maximally percussive while retaining the exact magnitude frequency spectrum, so this can be used to make "drum-like" sampled sounds from arbitrary input material. When the input is a single rhythmic event the minimum phase version generally sounds similar but more percussive. When the input is a phrase the minimum phase version tends to concentrate all the transients together at the beginning and produce a strange sort of temporal overlap of the perceptually distinct sequential events of the input.

The result of mixing an SMC (whether linear or minimum phase) back in with an original sound, aligned so that the PATs are equal, often fuses into a single perceptual event that sounds just like the original but with a stronger attack. In general, mixing in an unrelated sound with a sharp attack will give the perceived result of two distinct sounds playing together, but because of the spectral matching, an SMC will be much more likely to fuse with the original sound.

By controlling the relative volume of the SMC and the original sound, and by varying the duration of the SMC, it's possible to increase the "attackiness" of any single-rhythmic-event sound. Subtle effects can also be achieved by moving the SMC earlier or later with respect to the PAT of the original sound: though all alignments within a certain range may sound synchronous [10], moving the SMC a little bit earlier often results in a perceptually sharper attack in the mixed result. Of course the extreme case of this mixing is to play just the SMC with none of the original sound.

3.2. Morphing a Sound to a Click

Creating a series of SMCs with different durations from a single input sound results in a sort of "morph" (timbral and temporal interpolation) between the original sound all the way to an impulse, with each sound successively becoming shorter, more percussive, and more broad in frequency.

3.3. Time-aliased Minimum Phase as Compositional Effect

The use of frequency aliasing for compositional purposes is generally not useful because the frequencies of the resulting aliased components depend on the (in principle arbitrary) sampling rate of the signal and usually have no perceivable or usefully controllable relationship to the pitch of the un-aliased components. However, we can cause time aliasing in a controllable way when we convert signals to minimum phase: when computing the spectrum, instead of using sufficient zero-padding as described above, make the resulting FFT size just slightly bigger than the duration of the input sound. The useful parameter is the total FFT size (the input duration plus the number of trailing zeros).

As a compositional effect, time-aliased minimum-phase signals produce a strange form of periodicity, with an impulsive burst of energy at the beginning of the signal, then a weaker, less distinct, and generally higher-frequency second attack at exactly the midpoint of the resulting signal. This generates a form of quasi-periodicty that could be used to advantage in the synthesis of rhythmic material, and where the period is always a submultiple of the total FFT size parameter, usually one half.

[5] http://create.ucsb.edu/~matt/smc

3.4. Adding Rhythmic Markers to Music

Finally, spectrally matched clicks can be useful for the common task of adding multiple copies of a new sound on top of an existing music recording to hear the output of an algorithm such as an onset detector or pulse tracker. In this case one generally wants a percussive sound that will clearly mark the instants output by the algorithm. Such a sound should be enough like the existing recording that it will be easy to hear the relative timing of the algorithm's output against the music recording (according to the same auditory streaming concerns that motivated SMCs in the PAT measurement context), but distinct enough that it will be clearly perceived as something added to the original recording. SMC's arbitrary tradeoff of spectral similarity for percussiveness makes it ideal for creating a click sound that blends with or stands out from the original recording in the desired amount.

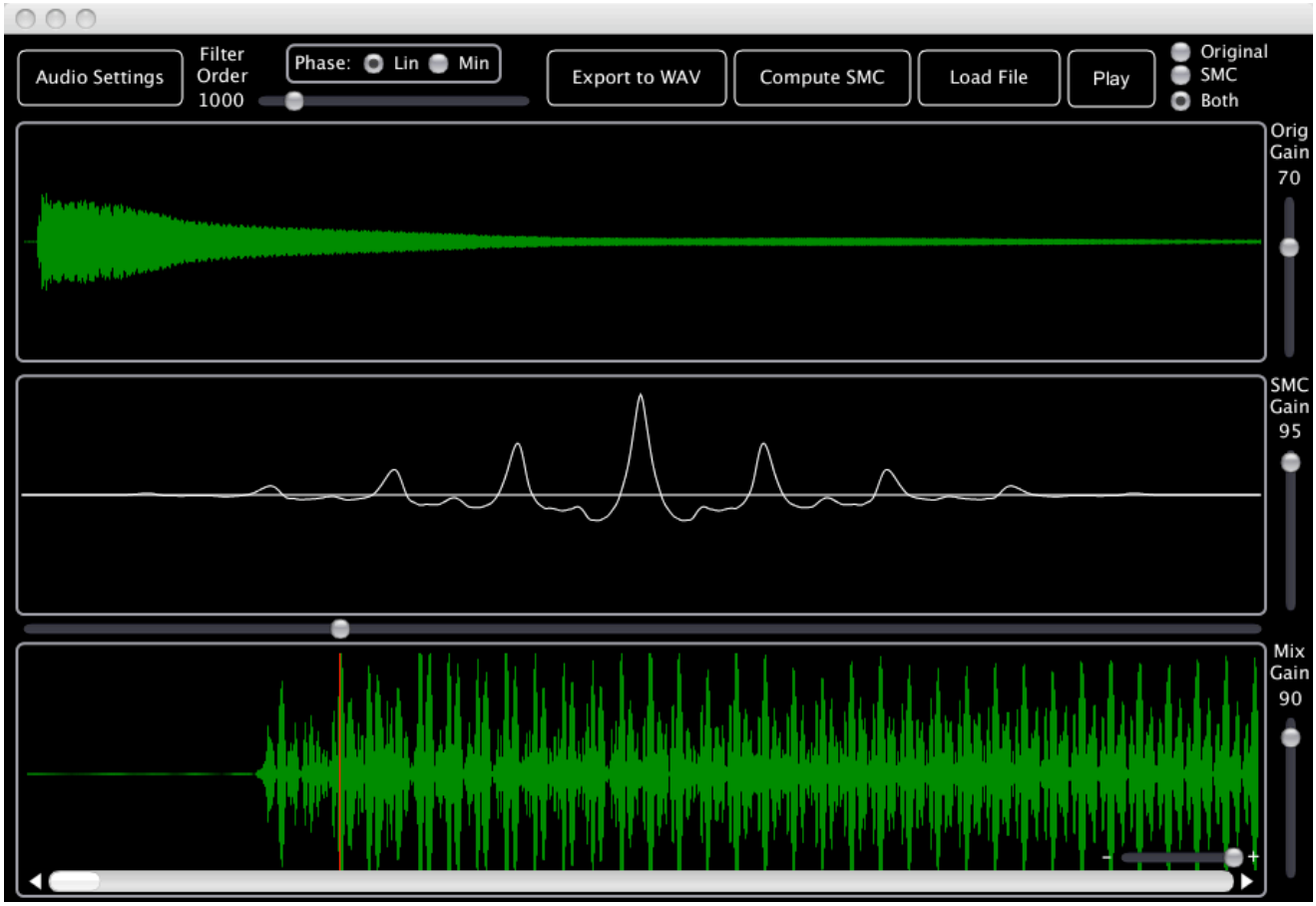

Figure 3: *Screenshot of our program.*

4. SOFTWARE IMPLEMENTATION

An SMC synthesis program has been written using the open-source JUCE C++ class library[6] and the FFTW C library[7] for FFT routines. The program, shown in Figure 3, provides a graphical user interface that allows the user to easily compute an SMC, align it with the audio source file, and export the result as a WAV file. Typical usage of the program starts with loading an arbitrary sound file, choosing the desired FIR filter order, specifying linear or minimum phase and then clicking the "Compute SMC" button. The original waveform of the audio file is displayed in the top pane and the SMC impulse response is displayed in the middle pane. The bottom pane is used for aligning the SMC with the original waveform. The waveform may be zoomed in down to one sample per pixel and a slider above the pane controls the placement of a vertical red line indicating the sample index where the SMC will be mixed with the original. The red line corresponds to the starting sample for a minimum phase SMC and the largest magnitude sample for a linear phase SMC. The gain of the SMC and original waveform may then be separately adjusted and the resultant waveform can be played back for review. Once the desired placement and mix of the SMC and original audio file is obtained, the audio can be exported as a 24 bit, 44100Hz, mono WAV file. The isolated SMC may also be previewed and exported.

Some additional features to be added to the SMC synthesis program are advanced file export options and individual control over FFT sizes. The file export options will include the capability to export a series of SMCs with control over each SMC's filter order and the spacing between individual SMCs. This additional functionality will allow for the creation of sequences embodying

[6] http://www.rawmaterialsoftware.com/juce
[7] http://www.fftw.org

the sound to click morph discussed in Section 3.2. Advanced control over the various FFT sizes will aid in further SMC research and also allow for the creation of time-aliased minimum-phase signals.

5. CONCLUSIONS AND FUTURE WORK

We have presented a family of techniques that take in any arbitrary sound segment and produce arbitrarily shorter sounds with stronger attacks than the original while retaining a similar magnitude frequency spectrum (and hence timbre) to the original. The resulting sounds can be used on their own, e.g., as part of a "morph" from one sound to a short click; they may also be added to the original sound to produce a result with an arbitrarily strong attack, in which case they have the advantage of tending to fuse perceptually with the original and not just sound like a second sound mixed in with the first.

Naturally the sonic character of the result depends critically on the exact input sound and the duration. Every minimum phase SMC we have heard so far is distinctly percussive in character, with a sharp attack and quasi-exponential decay. In general SMCs in the 50 ms range have an excellent match to the input, including pitch for all but the lowest sounds. SMCs from high pitched input sounds (e.g., an A5 piano tone) retain the pitch of the original even down to about 7ms. Below about 7 ms duration all SMCs have substantially the same sonic character (a "click") and differ primarily in brightness.

We would like to combine SMC synthesis with onset detection to create a low-latency real-time implementation that automatically adds an adjustable amount of percussiveness to a sequence of sound events. For this to be causal, the limiting factor on latency is the duration of the input sound, starting from each onset, whose spectrum is used to create each SMC.

Another musical application would be to compute an SMC matching an entire musical phrase, compute an onset detection function on the same phrase, and then convolve the detection function with the SMC. In this case the SMC would indeed be used as a traditional filter. The result should maintain the general spectrum of the input (because of the SMC) and also the rhythmic character (because of the detection function), but homogenize the timbre in a strange and potentially interesting way.

Currently we isolate the sound to be matched by slicing it out in the time domain. We have made SMCs based on source sounds taken from full music recordings because in each case there has been a segment of time containing only the desired sound event, with no other sounds playing. However, in general it would be desirable to be able to make SMCs matched to sound events taken out of arbitrary polyphonic mixes. The extremely difficult problem of extracting an individual sound event from a polyphonic mix is much easier when the end result is an SMC, because all that is required is the approximate magnitude frequency spectrum, which may survive intact even through artifacts and other imperfections of the polyphonic source separation process.

It would also be possible to compute the magnitude frequency spectrum S of the entire mix during the time span of the desired event, and then not even bother trying to recreate the desired event on its own, but instead partition S's energy arbitrarily into the desired spectrum of the SMC and a residual spectrum representing all of the other sounds and noise being removed. For example, suppose we want an SMC matching a snare drum from a recording, but that snare drum always plays in unison with a hi-hat cymbal. If we can find a time segment where the hi-hat cymbal plays a note alone, we can use it to approximate the spectrum of that instrument. Then we can find another segment where the snare drum and hi-hat play a note together, take the spectrum, subtract the spectrum of the hi-hat, and use the remainder to create an SMC of the snare drum.

As mentioned, there are very many algorithms for FIR filter design, some of which will probably perform better than the method described above. In this case we can define "better" as "producing a (perceptually) closer approximation to the desired magnitude frequency response for a given filter order." It may be the case that different techniques may be optimal depending on the duration of the SMC. It is also likely that an algorithm specifically designed to produce minimum-phase filters will perform better than the combination of designing a linear-phase filter and then converting the impulse response to minimum phase in a second step. Our downsampling step can be viewed as a moving average filter followed by decimation, which will produce aliasing and sidelobes in the frequency domain; more sophisticated frequency domain data reduction steps may produce better results. Finally, it might be advantageous to use a filter design method in which the minimized error between the desired magnitude frequency spectrum and the filter's magnitude frequency spectrum is weighted perceptually by frequency region.

6. ACKNOWLEDGMENTS

Jonathan Abel, Jerry Gibson, Julius Smith.

7. REFERENCES

[1] B. C. J. Moore and B. R. Glasberg, "A revision of Zwicker's loudness model," *Acta Acustica* 82, pp. 335-345, 1996.

[2] L. R. Rabiner and B. Gold, *Theory and Application of Digital Signal Processing*, Prentice-Hall, Englewood Cliffs, NJ, USA, 1975.

[3] J. O. Smith III, "Synthesis of bowed strings," in *Proc. Intl. Computer Music Conf*., Venice, Italy, pp. 308-340, 1982.

[4] J. O. Smith III, "Techniques for Digital Filter Design and System Identification with Application to the Violin," Ph.D. thesis, Stanford University, 1983.

[5] J. O. Smith III, *Introduction to Digital Filters with Audio Applications*, <http://ccrma.stanford.edu/~jos/filters> (online book), 2007.

[6] J. O. Smith III, *Spectral Audio Signal Processing, March 2007 Draft*, <http://ccrma.stanford.edu/~jos/sasp> (online book), 2007.

[7] S. W. Smith, The Scientist and Engineer's Guide to Digital Signal Processing, California Technical Publishing, 1997.

[8] D. L. Wessel, "Timbre space as a musical control structure," *Computer Music Journal* vol. 3, no. 2, pp. 45-52, 1979.

[9] A. B. Williams and F. J. Taylor, *Electronic Filter Design Handbook*, McGraw-Hill, New York, 2006.

[10] M. Wright, "The Shape of an Instant: Measuring and Modelling Perceptual Attack Time with Probability Density Functions," Ph.D. Diss, Stanford University, 2008.