MULTIPLE F0 ESTIMATION IN THE TRANSFORM DOMAIN

Christopher A. Santoro[+*] **Corey I. Cheng**[*#]

[+]LSB Audio
Tampa, FL 33610
chris@lsbaudio.com

[*]University of Miami
Music Engineering Technology
Frost School of Music
Coral Gables, FL 33124

[#]University of Miami
Department of Electrical and
Computer Engineering
coreyc@miami.edu

ABSTRACT

A novel algorithm is proposed to estimate the fundamental frequencies present in polyphonic acoustic mixtures expressed in a transform domain. As an example, the algorithm operates on Modified Discrete Cosine Transform (MDCT) coefficients in order to demonstrate the utility of the method in commercially available perceptual audio codecs which use the MDCT. An auditory model is developed along with several optimizations that deal with the constraints of processing in the transform-domain, including an interpolation method, a transform-domain half-wave rectification model, tonal component estimation, and sparse convolution. Test results are separated by instrument and analyzed in detail. The proposed algorithm is shown to perform comparably to state of the art time-domain methods.

1. INTRODUCTION

Perceptually coded formats such as mp3 and AAC have become the dominant storage and distribution format for commercial digital music. These formats are popular because they greatly reduce bandwidth and memory requirements related to transmission and storage. As a result of the successes of these formats, portable media players are becoming increasingly important platforms for the analysis and synthesis of digital media. These devices have limited processing power and battery life, and therefore require analysis and synthesis algorithms with minimal computational complexity where possible.

One emerging family of algorithms that is finding increased applicability in music information processing is multiple fundamental (F0) estimation. Loosely speaking, the purpose of multiple F0 estimation is to estimate the perceived pitches of multiple harmonic series, such as those created by the human voice or various musical instruments, when sounding concurrently. Multiple F0 estimation finds widespread use as a front-end to various pitch tracking and source separation algorithms.

State of the art analysis algorithms are typically designed to begin their operations on uncompressed PCM audio signals in the time-domain. Because music files on portable devices are stored in a perceptually coded format, they must first be decoded before the algorithms can begin their analysis. For example, many perceptual audio codecs spend considerable resources in using the well-known Inverse Modified Discrete Cosine Transform (IMDCT) to synthesize a time-domain signal during the decoding process. Therefore, it would be especially advantageous to avoid this expensive decoding process where possible, and operate directly on the native MDCT representation used in a perceptually coded file.

This work adapts a state of the art multiple F0 estimation algorithm to operate directly on a transform-domain representation used in modern audio codecs as a starting point for this research. Very few authors have researched transform-domain processing. Previous works include beat detection [1], music/speech classification [2], and sinusoidal analysis [3]. A primary reason for the limited body of work related to F0 estimation algorithms is the limited frequency resolution used in transform-based audio coders. When working in the transform domain, we are stuck with whatever frame size the coder uses. A secondary difficulty with transform domain processing is that some time-domain processes do not easily lend themselves to operation in the transform-domain. In this work several novel modifications to the auditory model are proposed that successfully mitigate both of these limitations. A third difficulty with transform domain processing is that processing in some transform domains, such as the MDCT domain, can be problematic because of some of the aliasing properties of the transforms [10].

We propose an algorithm for multiple F0 estimation in the transform domain that adapts the work of Klapuri [4] to function in the MDCT domain. Like [4], the proposed algorithm uses a model of the human auditory system along with iterative estimation and cancellation to estimate component F0s, as the auditory model described by Klapuri consists of an auditory filterbank followed by half-wave rectification and low pass filtering. However, in this work, each of these processes is adapted to operate in the transform-domain. We also incorporate modifications to the iterative estimation portion of the algorithm by Paz [6] in order to improve performance.

2. OVERVIEW OF REFERENCE ALGORITHM

The design of the proposed algorithm is based on the work of Klapuri [4]. The basic framework of Klapuri's

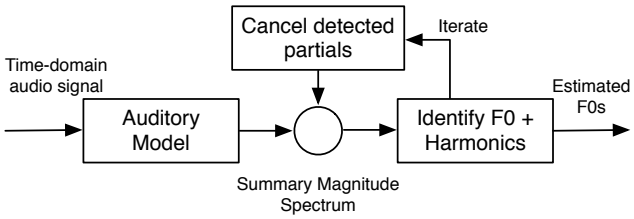

Figure 1. Overview of reference multiple F0 estimation method

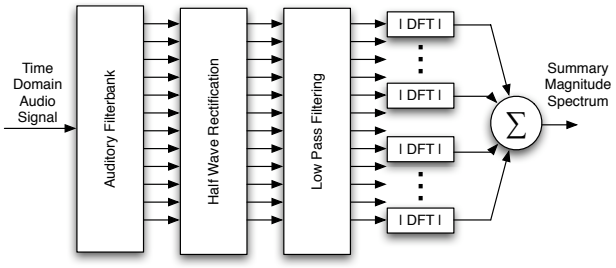

Figure 2. Detail of auditory model used in reference method

method is shown in Figure 1. A key feature of the algorithm is the auditory model, shown in Figure 2. The auditory model used by Klapuri consists of a 70 channel gammatone filterbank design, using the computationally efficient implementation by Slaney [11], followed by half-wave rectification and low-pass filtering. Finally, a Discrete Fourier Transform (DFT) is taken in each channel, and the spectra in each channel are summed to create a summary magnitude spectrum.

To identify F0s, the *salience* of each candidate F0 is calculated using (1), where U_{SMS} is the summary magnitude spectrum, $K_{\tau,m}$ is a region where each partial is expected to be based on the period (τ) of the F0 and the harmonic number (m), and $\omega(\tau,m)$ is an exponentially decreasing weighting function dependent on F0 and harmonic number.

$$s(\tau) = \sum_{m=1}^{M} \omega(\tau,m) \max_{k \in K_{\tau,m}} U_{sms}(k) \qquad (1)$$

The salience can be interpreted as a measure of the perceptual strength of each candidate F0. On each iteration, the F0 with the highest salience is chosen. Its partials are then identified and partially subtracted from the mixture according to an exponentially decreasing weighting scheme. This process is repeated until a known number of F0s have been estimated. In Klapuri's work the polyphony considered to be known *a priori* in most cases.

3. PROPOSED ALGORITHM

Our proposed algorithm comprises three main stages: interpolation, a low complexity transform-domain auditory model plus iterative estimation and subtraction. The algorithm makes changes mainly to the auditory model and adds transform domain interpolation to the front end of the salience calculation in an attempt to deal with the limited frequency resolution of the transform-domain representation used in audio codecs.

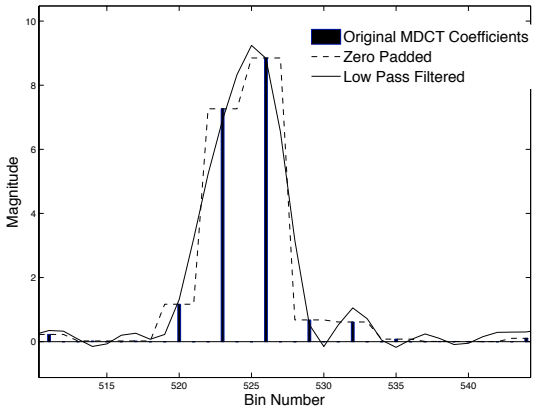

Figure 3. Example of interpolation process. In this figure the peak has been shifted to the left after interpolation based on the distribution of energy around the old peak.

3.1. Interpolation of MDCT coefficients

As stated previously, one of the fundamental limitations of transform-domain processing is that we are stuck with whatever frame size the codec uses. Codecs like AAC use large frame sizes of 2048 samples for tonal content, and smaller frame sizes of 256 samples for transients [9]. A frame size of 2048 samples at 44.1 kHz yields a frequency resolution of roughly 21.5 Hz. Since F0s are more closely spaced at lower frequencies in contemporary western scales, this means that a peak in one MDCT bin can span as many as six notes. In fact, F0s are not spaced more than 21 Hz apart until the 4th octave (E4 or 330 Hz). If nothing is done to address this, this means that any peak corresponding to an F0 below 330 Hz could be one of many F0s. This assumes an equal tempered scale. The details of other tuning systems will be different.

To solve this problem, we use a simple interpolation method in the transform-domain. While this method does not increase the real frequency resolution of the transform-domain representation, it does have the effect of shifting peaks to a more accurate location corresponding to the true F0, making estimates of frequency when peak picking more accurate. The implementation of this method consists of zero padding/upsampling, then performing a zero order hold and low pass filtering. An upsampling factor of 3 was used here, but any odd factor may be used.

The low pass filtering was implemented as a simple FIR filter, which takes the form of a convolution with a sinc function. We found a filter length of 24 samples to be adequate for an upsampling factor of 3. An example of the result of the interpolation process is shown in Figure 3.

What is left as a result of the interpolation process can no longer be called a realistic MDCT spectrum, but this is of no consequence to the proposed algorithm. What we do have at this point is a reasonable estimation

of the magnitude spectrum of our signal, and a better estimation of true peak locations due to the interpolation process. It is important to remark that this is a computationally expensive process, which could probably be replaced by a less intensive interpolation method. However, this method was chosen here for its simplicity and good results.

3.2. Transform-Domain Auditory Model

After interpolation, the spectrum is passed onto a transform-domain auditory model, where we implement a modified version of the Unitary model of human pitch perception, proposed by Meddis [7]. In this section, we describe our modifications and improvements to the model's four following steps:

1. The stimulus is passed through a filterbank of band-pass filters, which simulate the action of the basilar membrane.
2. Each sub-band signal is compressed, half wave rectified, and low-pass filtered to obtain the time domain amplitude envelope.
3. Periodicity estimation is carried out on each sub-band.
4. Periodicity information from each sub-band is combined across channels.

Step 1 of the unitary model is said to mimic the frequency selectivity of the inner ear. Typically a gammatone filterbank implementation by Slaney is used, although the number of channels necessary to achieve good results is debated. Depending on the application, previous works using auditory models use as few as 2 channels [8] and as many as 70 [4]. For this reason, we implemented filterbanks with 8, 16, 32, 64, and 70 channels to explore the effect on performance of the algorithm. If the number of channels can be reduced, then the computational complexity of the algorithm can be reduced significantly.

Step 2 of the unitary model processes information contained in the time domain amplitude envelope of the stimulus signal. Many musicians know the information we are looking for here as *beating*. Beating occurs when two sinusoids that have slightly different frequencies cancel and/or reinforce each other periodically. The fundamental period of the beating corresponds to the difference in frequency between the two sinusoids. Thus, this process (half wave rectification and low pass filtering) can be considered as a way of analyzing the intervals between harmonics, which corresponds to the F0 of a harmonic sound. This type of information is called *spectral interval information*.

Steps 3 and 4 are merely ways to extract the periodicity of the time-domain amplitude envelope, which is reinforced in step 2. Typically an autocorrelation function (time-domain) or a Discrete Fourier Transform (frequency-domain) is used in each channel. Given these processes, it is easy to see why we do not want a large number of channels if it is not necessary.

Some of these steps lend themselves easily to a transform domain implementation, while others prove more difficult. For example, step 1 of the auditory model is trivial in the transform domain. The auditory filterbank can be implemented easily by a matrix multiply if we store the magnitude response of each channel in an $N \times nC$ matrix, where N is the number of coefficients in our upsampled MDCT spectrum and nC is the number of channels in our filterbank. The magnitude response of each channel of the auditory filterbank can be obtained easily by first obtaining a standard time-domain design, and processing each channel with an impulse. Taking the magnitude DFT of the result yields the magnitude response of each channel. Since the filterbank is static, it can be calculated once, and the magnitude response can simply be stored in memory.

Step 2 of the auditory model is the most difficult to adapt for the transform domain. It is not obvious how half-wave rectification can be translated into the transform domain. However, Klapuri [5] points out that half wave rectification can be modeled as a convolution operation. The mathematical details of that argument are beyond the scope of this paper, but the interested reader is encouraged to check the source for an in depth analysis. Instead, we simply note that since the goal of this step of the model is to reinforce spectral interval information, that a convolution operation is an intuitive method for extracting that information.

There are two difficulties with the convolution of spectra to extract spectral interval information. One is that spectra have a DC offset due to the fact that all magnitude coefficients are positive. This causes a triangular shaped buildup around DC that obscures peaks indicating prominent spectral intervals. The other is that the process is prohibitively expensive computationally, especially since this is a process that must be performed on each channel individually. A standard way to attack the first problem is to subtract the mean from the signal. Not only does this not work in this case because the DC offset is caused by a small region of disproportionately large peaks, but it also does not address the second problem. We propose here a method for solving both of these problems based on tonal component estimation.

Since we are looking for intervals between peaks, we begin the process by finding the locations of the peaks in the subband spectrum. This is called tonal component estimation (TCE). First, the derivative of each subband is taken. Next, the derivative is used to analyze the slope of each peak. Using a sliding window of 15 coefficients, local maxima are found by identifying locations in which the derivative transitions from a positive value to a negative value, and the difference between the two is greater than some threshold. The mean of each subband signal was found to be a good threshold, though this value may be changed to adjust sensitivity. Once we have identified the locations of tonal components, we replace each peak

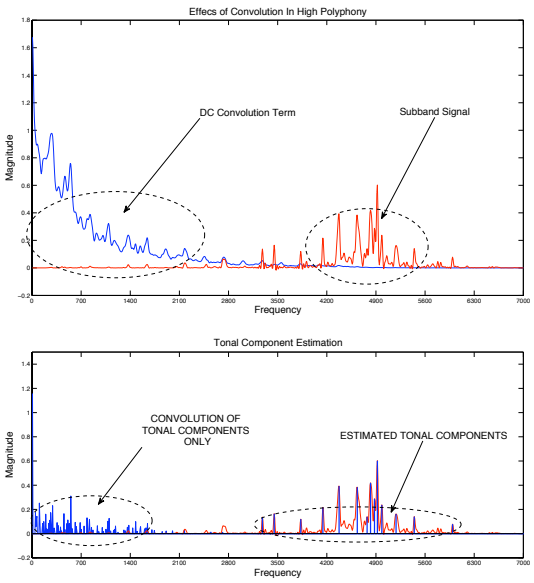

Figure 4. Comparison of TCE and sparse convolution to traditional convolution of MDCT spectra.

with a single spike that has the same amplitude as the peak.

Using this process, each subband signal is reduced to only the most pertinent information (i.e., the locations and magnitudes of harmonics). This transforms the signal into a sparse vector, since most of the elements in each subband signal are now zeros. Next, a sparse convolution may be used to extract spectral interval information. This process is shown in Figure 4 and compared to a standard convolution. Not only does it solve the problem of DC buildup, but it also reduces the computational complexity drastically.

The result of this process is a vector of extracted spectral interval information, V_c. By combining this with the original subband spectrum, we can obtain both spectral interval and spectral location information. Therefore, we calculate a weighted combination of the original spectrum (X_c), using (2), where α is a simple parameter which can be used to adjust the importance of spectral interval information. Y_c is the resulting signal in each channel after the half wave rectification process.

$$Y_c = (1-\alpha)\, X_c + \alpha\, V_c \qquad (2)$$

Step 3 of the auditory model is performed by a DFT in the reference method. Here, we are already in the frequency domain, so this step can be skipped, yielding a large computational savings. Step 4 is also trivial and is computed by summing across channels to create a summary magnitude spectrum, U_{SMS}. This is shown in (3). In this step, channels that have peaks in the same location in their V_c components (meaning their spectral interval information is in agreement) reinforce each other to accentuate (or in some cases, reproduce) the peaks corresponding to the F0s in the mixture.

$$U_{sms}(k) = \sum_{c=1}^{nC} Y_c(k) \quad for\ all\ k \qquad (3)$$

3.3. Iterative Estimation and Subtraction

Once the summary magnitude spectrum has been calculated, the algorithm performs an iterative estimation and subtraction process largely similar to that in the reference algorithm. On each iteration, the salience is calculated for all fundamental candidate periods τ as described in (1). The candidate period with the maximum salience is chosen to be an F0. Next, we attempt to identify the peaks that contributed to the salience of the currently estimated F0. An adaptive scheme is used to capture peaks as well as their side lobes, which was developed by Paz [6] and was found to improve performance significantly. First local maxima are identified within each region defined by $K_{\tau m}$. Next, the boundaries are expanded until they lie at adjacent local minima. Once this is completed, a detected spectrum U_D is formed consisting of the partials of the estimated fundamental. These partials are then weighted by the same weighting function that was used to calculate the salience. This allows us to remove some of the energy in each partial, but not all of it. This is critical for cases in which multiple sounds have partials that overlap.

Finally, a residual spectrum U_R is formed by subtracting U_D from U_{SMS}. The process of calculating the salience and estimating the partials of F0s is repeated on U_R a number of times that is equal to the polyphony, which is known *a priori*. The estimated F0s are then quantized to the nearest frequency value corresponding to a valid note on the equal tempered scale, with A4 corresponding to 440Hz.

4. RESULTS

The proposed algorithm was tested in a similar manner to the reference algorithm. Polyphonic mixtures of 2, 4, and 6 notes were created from four different types of instruments: Saxophone, Flute, Violin, and Cello. Sample recordings of individual notes were used from the University of Iowa[1] database. For each polyphony and instrument, 100 mixtures were created by lining up the onsets of notes and mixing them at equal RMS levels. Each individual file was then encoded using the LAME mp3 encoder[2] at 128 kbps. The results presented in all tests for the reference algorithm are based on a careful implementation based on the information given in the papers published by the author.

4.1. Decoder Model

To modify an actual decoder to return just MDCT coefficients (prior to taking the IMDCT and performing overlap and add) would have taken considerable time and effort. Instead, we constructed a simplified decoder

[1] http://theremin.music.uiowa.edu/MIS.html
[2] http://lame.sourceforge.net/

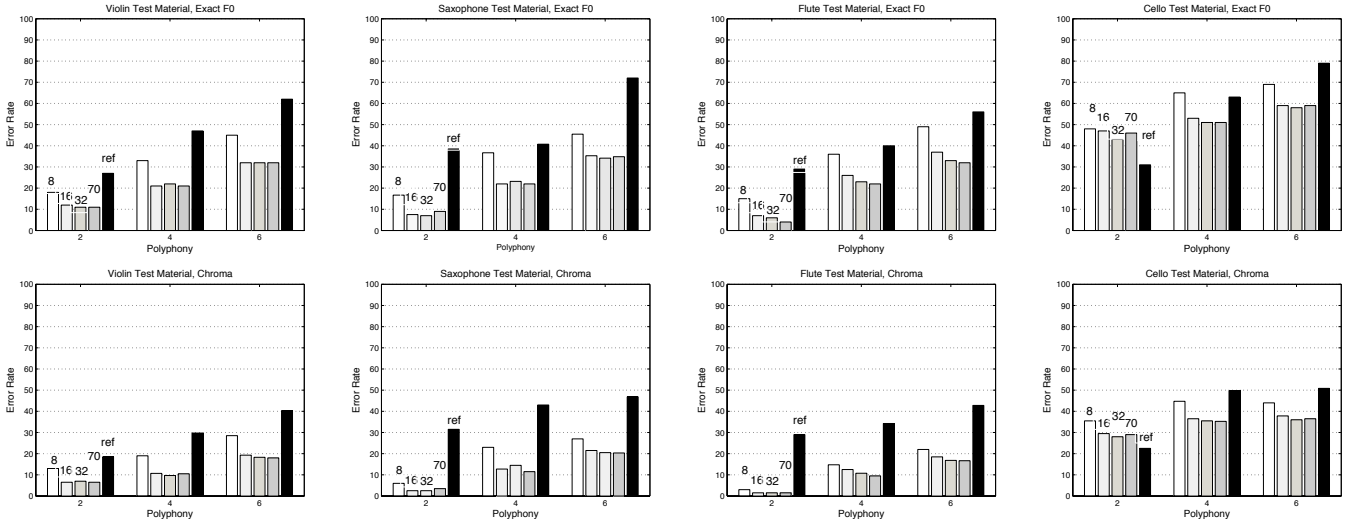

Figure 5. Multiple F0 estimation error rates for several musical instruments and several polyphonies. The reference method is in black. The proposed algorithm is tested for a Unitary model of hearing having 8, 16, 32, and 70 frequency bands.

model that fully decodes the mp3 file and then reverses the last two steps by windowing and then taking the MDCT.

While a real partial decoder would be best, we consider this decoder model to be a sufficient first attempt at multiple F0 estimation in the transform-domain. To implement the MDCT, we used a "fast MDCT" which utilizes an FFT with two rotations to perform an MDCT. In order to have a consistent basis for comparison with the reference algorithm, we used a frame size of 46ms, which corresponds to 2048 time-domain samples. This is also the largest frame size used in AAC [9].

4.2. F0 Estimation Results

The results of the F0 estimation tests for each instrument and filterbank design are shown in the top of Figure 5.

Error rate is calculated in the same manner as in the reference. The most important result of the F0 estimation results is that performance is roughly equal for filterbank designs with as few as 16 channels. The algorithm performed significantly worse when using less than 16 channels. Interestingly, a 16 channel filterbank design of the range of 60 Hz to 2.2 kHz roughly corresponds to a 1/3 octave filterbank design (which would have 19 channels in this case). This is a common psychoacoustically motivated design for equalizers in stereo systems. This result is important, as it demonstrates that we can drastically reduce the complexity of our filterbank while paying a minimal performance penalty.

Additionally, the results show that the algorithm performs well, outperforming the reference algorithm in most cases. The error rates published here are slightly higher than previously published for the reference using a 46ms window. This could be due to implementation inaccuracies or a discrepancy in test material.

4.3. Chroma Estimation Results

In some applications, the exact octave that a note is from may not be as important as the chroma of the note. That is, in a mixture that contains the notes C3, E4, and G3, an estimation of C, E, and G may be sufficient. To investigate the proposed algorithm's perfomance in chroma estimation, the F0 estimation tests were re-run, but this time octave errors were not counted as errors. The results in the bottom half of Figure 5 show that the error rates dropped drastically for all instruments except for Cello. Error rates for each filterbank design with more than 8 channels were less than 5% for low polyphonies. This shows that a majority of the errors from the F0 estimation tests were octave errors.

4.4. Discussion

While the proposed algorithm's performance was impressive on each task, the question still remains as to why the Cello performed so poorly, while the other instruments performed well. One would think that the inharmonicity of stringed instruments as well as the low frequency range of the cello played a part. An analysis of the distribution of samples for each instrument was conducted and this revealed that indeed the cello had a distribution that occupied a significantly lower range than the other instruments. This is likely to have played a larger role than inharmonicity, since the algorithm had no problem dealing with the violin samples.

This reveals a primary weakness of the proposed algorithm. It does not seem to deal well with lower F0s. This is most likely due to inadequate frequency resolution for F0s below the 3rd octave. A higher upsampling rate in the interpolation stage may mitigate this somewhat, but this would increase computational complexity.

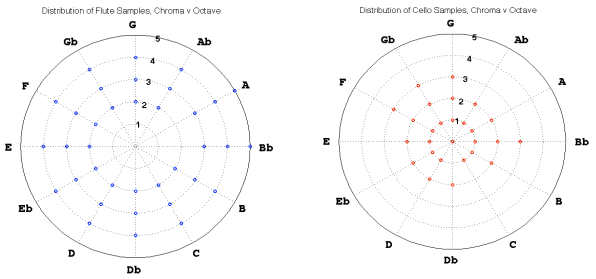

Figure 6. Polar comparison of flute (left) and cello (right) sample distributions, by chroma (angle) vs. octave (radius).

5. CONCLUSIONS AND FUTURE WORK

In conclusion, we have shown here that it may be possible to perform multiple F0 estimation entirely in the transform-domain. We have adapted a state of the art algorithm to work in the transform-domain, which includes a model of human pitch perception. We have shown that upsampling and interpolation of MDCT coefficients is a viable strategy for mitigating the inadequate frequency resolution of frame sizes native to perceptual audio coders. However, we have also found that F0s in the lower octaves still remain a problem due to limited frequency resolution.

For the purposes of comparing this algorithm against other multiple F0 estimation algorithms, it would be useful to use a MIREX database [12] for future test material. This would provide more reliable grounds on which to make comparisons. The test material used here was intended to be as close as possible to that used in the reference method. Furthermore, while a large effort was made to accurately implement the reference algorithm, mistakes will always be made because limited publication space inevitably causes some details to be left out.

Future work should also include a more detailed decoder model, as well as further experimentation with different upsampling factors and filterbank designs. The MDCT spectra used for this investigation, while fine for a starting point on this research, are certainly not an exact representation of what we would see coming from an actual decoder. Strategies will need to be developed to deal with the limitations of more realistic representations of MDCT coefficients in perceptually coded files. While this work does not address these tedious details, it lays the groundwork for an evolution in that direction.

6. REFERENCES

[1] E. Ravelli, G. Richard, and L. Daudet: "Fast MIR in a sparse transform domain," *Proceedings of the International Symposium on Music Information Retrieval*, pp. 527-532, 2008.

[2] G. Tzanetakis and P. Cook: "Sound analysis using MPEG compressed audio," *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, pp. 761-764, 2000.

[3] S. Merdjani and L. Daudet: "Direct estimation of frequency from MDCT-encoded files," *Proceedings of the 6th International Conference on Digital Audio Effects,* 2003.

[4] A. Klapuri: "A perceptually motivated multiple-F0 estimation method," *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, 2005.

[5] A. Klapuri: "Signal processing methods for the automatic transcription of music," *PhD Thesis*, Tampere University of Technology, 2004.

[6] L. Paz: "Multiple-F0 estimation using auditory models," *M. Sci. Thesis,* University of Miami, 2008.

[7] R. Meddis: "Virtual pitch and phase sensitivity of a computer model of the auditory periphery. 1. Pitch idenficitaion," *Journal of the Acoustical Society of America*, vol. 86, no. 6, pp. 2886-2882, 1991.

[8] T. Tolonen and M. Karjalainen: "A computationally efficient multipitch analysis model," *IEEE Trans. On Speech and Audio Processing,* vol. 8, no. 6, 2000.

[9] M. Bosi, et al.: "ISO/IEC Advanced Audio Coding," *Journal of the Audio Engineering Society*, vol. 45, no. 10, pp. 789-811, 1997.

[10] C. Cheng: "Method for Estimating Magnitude and Phase in the MDCT Domain," *Audio Engineering Society (AES) 116th Convention*, Berlin, Germany, 2004.

[11] M. Slaney: "An Efficient Implementation of the Patterson Holdsworth Auditory Filter Bank," Perception Group, Apple Computer, Tech. Rep. 35, 1993.

[12] J. Downie: "The Music Information Retrieval Evaluation Exchange: A window into music information retrieval research," *Acoustical Science and Technology*, vol. 29, no. 4, pp. 247-255, 2008.