AN EFFICIENT PITCH-TRACKING ALGORITHM USING A COMBINATION OF FOURIER TRANSFORMS

Sylvain Marchand

SCRIME - LaBRI, Université Bordeaux 1
351, cours de la Libération, F-33405 Talence cedex, France
`sm@labri.u-bordeaux.fr`

ABSTRACT

In this paper we present a technique for detecting the pitch of sound using a series of two forward Fourier transforms. We use an enhanced version of the Fourier transform for a better accuracy, as well as a tracking strategy among pitch candidates for an increased robustness. This efficient technique allows us to precisely find out the pitches of harmonic sounds such as the voice or classic musical instruments, but also of more complex sounds like rippled noises.

1. INTRODUCTION

Determining the evolutions with time of the pitch of sound is an important problem. This is indeed extremely useful for controlling synthesizers from this pitch information and absolutely necessary for pitch-synchronous algorithms such as PSOLA techniques [1].

Various methods have been proposed for the determination of the pitch as a function of time (pitch tracking). They use either the autocorrelation factor [2], other physical [3, 4] or geometric [5] criteria, least-square fitting [6], pattern recognition [7] or even neural networks [8]. Arfib and Delprat use in [9] the inverse FFT of the sound spectrum modulus limited to the positive frequency. In this article, we propose a new composition of two Fourier transforms, thus introducing the "Fourier of Fourier" transform of great interest for pitch extraction.

After a brief introduction to sounds and their pitches in Section 2, we introduce in Section 3 our new transform. This transform allows us to extract accurate pitch candidates. We present in Section 4 an efficient and accurate pitch-tracking algorithm based on this transform. We show how to choose the right pitch candidate most of the time in order to reach an acceptable level of robustness. Finally, we give some results – in terms of performance, accuracy, and robustness – in Section 5.

2. SOUNDS AND PITCHES

Pitch is not a physical parameter, but a perceptive one. There is a close link with frequency, but this relation is rather complex. For a single sinusoid, Equation 1 gives the relation between the frequency F and the pitch P in the harmonic scale:

$$P(F) = P_{\mathrm{ref}} + O \log_2\left(\frac{F}{F_{\mathrm{ref}}}\right) \tag{1}$$

where P_{ref} and F_{ref} are, respectively, the pitch and the corresponding frequency of a tone of reference. In the remainder of this paper we will use the values $P_{\mathrm{ref}} = 69$ and $F_{\mathrm{ref}} = 440$ Hz. The constant O is the division of the octave. An usual value is $O = 12$, leading to the classic dodecaphonic musical scale. With these values, P is the MIDI pitch, where 69 corresponds to the `A3` note, 70 to `A#3`, etc.

2.1. Harmonic Sounds

For an harmonic sound, the perceived pitch corresponds to a kind of greatest common divisor (*gcd*) of the frequencies of the harmonics, that is the fundamental. The fundamental coincides with the frequency of the first harmonic. But this first harmonic may be missing, or "virtual".

2.2. About Noise

For a narrow-band noise, the pitch corresponds to the frequency of the middle of the band. For a rippled noise, the pitch corresponds to the *gcd* of the peaks in the spectral envelope, even if the first peak is missing.

3. "FOURIER OF FOURIER" TRANSFORM

In our FT^n analysis method [10, 11], we proposed to take advantage of two Fourier transforms computed in parallel. The resulting analysis precision [12] has recently been used for accurate pitch detection [13]. We show here that the use of two Fourier transforms in sequence is of great interest too.

More precisely, we consider the magnitude spectrum of the Fourier transform of the magnitude spectrum – limited to positive frequencies – of the Fourier transform of the signal. Let us denote by "Fourier of Fourier transform" this combination of the two Fourier transforms. Note that this transform is not the same as the well-known "cepstrum", which is the (inverse) Fourier transform of the logarithm of the spectrum resulting from the Fourier transform.

This transform is well-suited for pitch-tracking, that is for computing the fundamental frequency of the sound, even if it is missing or "virtual". For example, if we consider an harmonic sound, its Fourier transform has a series of peaks in its magnitude spectrum corresponding to the harmonics of the sound, at frequencies close to multiples of the fundamental frequency F. Some harmonics may be missing, even the fundamental itself. Anyway, the Fourier of Fourier transform of an harmonic sound shows a series of peaks, and the first and most prominent one corresponds to the fundamental frequency F of the harmonic sound, and its amplitude is the sum of the amplitudes of the harmonics of the sound. Figure 1 illustrates this.

In the spectrum resulting from the first Fourier transform (FT), the index of a bin i_{FT} is related to the analyzed frequency f. More

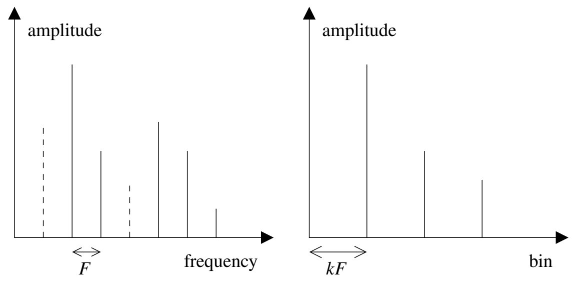

Figure 1: *The power spectrum of an harmonic sound (left) together with the power spectrum resulting from the Fourier transform of this first spectrum (right). There might be missing harmonics (dashed).*

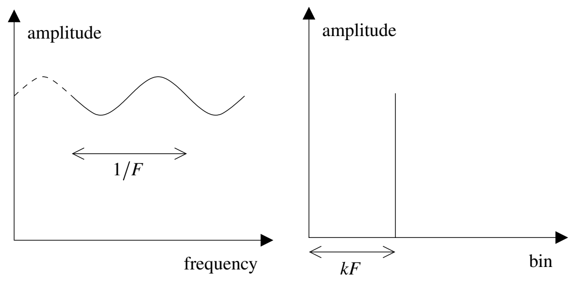

Figure 2: *The power spectrum of a rippled noise (left) together with the power spectrum resulting from the Fourier transform of this first spectrum (right). There might be missing ripples (dashed).*

precisely, if F_s is the sampling rate and N the size of the Fourier transform, we have:

$$i_{\mathrm{FT}} = Nf/F_s \tag{2}$$

When considering an harmonic sound whose fundamental is F, the magnitude spectrum shows a series of uniformly-spaced peaks (unless some harmonics are missing). The distance between two consecutive harmonics is F, which corresponds to a period of Δ bins where:

$$\Delta = NF/F_s \tag{3}$$

In the spectrum resulting from the Fourier transform of the magnitude spectrum of the first Fourier transform (FT(FT)), the greatest local maximum of magnitude (apart from the one corresponding to bin 0) is located at the bin corresponding to index:

$$i_{\mathrm{FT(FT)}} = N/(2\Delta) \tag{4}$$

In Equation 4 we consider that the size of the second Fourier transform is again N. This is no mandatory though. It is then possible to recover the fundamental frequency from the value of this index:

$$F = \frac{F_s/2}{i_{\mathrm{FT(FT)}}} \tag{5}$$

The same reasoning also works for single sinusoids or rippled noises (even if some ripples are missing). Figure 2 illustrates this. As a consequence, the Fourier of Fourier transform turns out be be extremely well-suited for determining the pitch of these sounds, as well as their volume. We have also verified this for natural sounds, as shown in Figure 3. It is important to note that the amplitude corresponding to the $i_{\mathrm{FT(FT)}}$ index is close to the sum of the amplitudes of the harmonics constituting the sound. One can also obtain instead a good approximation of the RMS (Root Mean Square) amplitude, by replacing the amplitudes by their squares in the magnitude spectrum prior to the second Fourier transform, and by replacing the amplitudes by their square roots in the magnitude spectrum resulting from this second transform (see [14]). The result must be scaled by a $1/\sqrt{2}$ factor though.

4. PITCH-TRACKING ALGORITHM

We have seen previously that the Fourier of Fourier transform – the magnitude spectrum of the Fourier transform of the magnitude spectrum of the Fourier transform of the signal – is well-suited for pitch tracking, that is for computing the fundamental frequency of the sound, even if it is missing or “virtual”.

4.1. Using the Order-1 Fourier Transform

We propose to use the Fourier of Fourier transform to perform the detection of the pitch. A very important feature is that we may use the FT^n method [10, 11] for $n = 1$ – also called the order-1 Fourier transform or simply the “derivative algorithm” – instead of the classic Fourier transform for a better accuracy for the pitch detection.

More precisely, if we want to determine the pitch at a certain time t, then we consider a small portion of temporal signal centered at t. This temporal frame is multiplied by the Hann analysis window, and then analyzed using the order-1 Fourier transform. With this transform, the spectral peaks are extracted with an enhanced precision in comparison to the classic Fourier transform.

With this technique, the short-term magnitude spectrum has then to be reconstructed from the spectral peaks prior to the second Fourier transform. In fact, this is done by a simple sampling of the spectrum. For a greater accuracy, a convolution of the peaks with the spectrum of the Hann window can be used as a preliminary. After that, the classic Fourier transform is used, and the spectral peaks are extracted. The resulting n spectral peaks corresponds to frequencies (see Equation 5) that are pitch candidates.

4.2. Pseudo-partial Tracking

We have seen that the fundamental frequency of the sound is given – in theory – by the greatest local maximum of magnitude (apart from the one corresponding to bin 0) in the spectrum resulting from the Fourier of Fourier transform. As a consequence, the pitch should be the frequency of the pitch candidate with the greatest amplitude.

The problem is that for some sounds this maximum of energy is detected at the wrong place from time to time. This often leads to jumps among octaves and results in a poor robustness. We propose to apply a peak-tracking strategy similar to partial tracking (see [12]), except that this time we deal with “pseudo-partials”, that is partials detected in the spectrum resulting from the Fourier of Fourier transform. When obtain a set of partials, as shown in

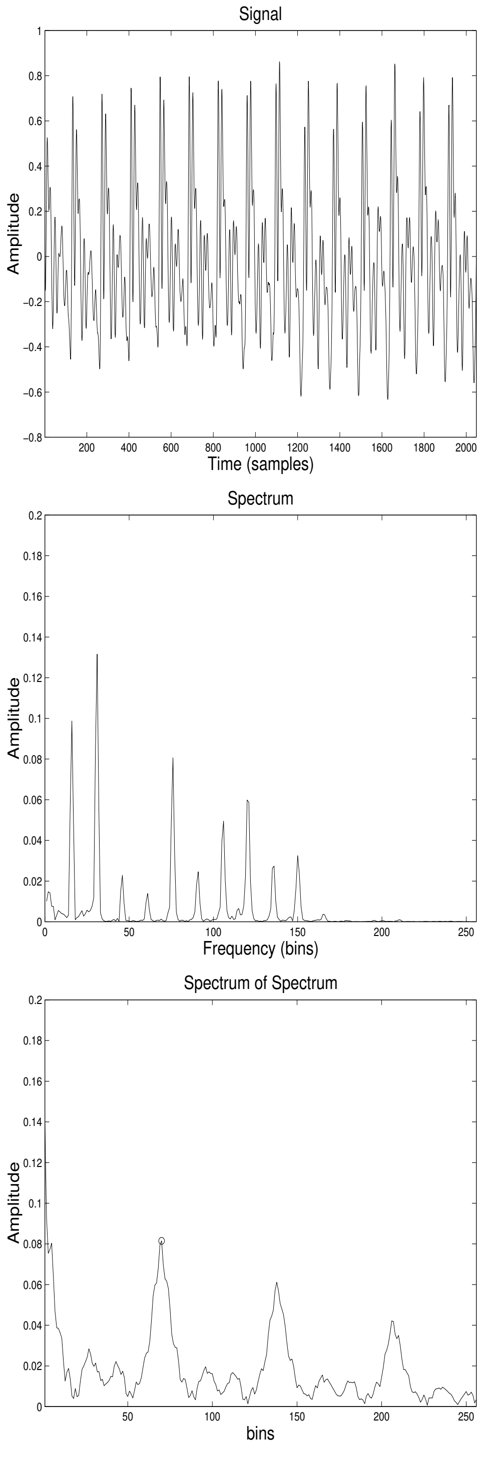

Figure 3: *Fourier of Fourier. From top to bottom are the original signal (singing voice, sampled at $F_s = 44100$ Hz), its magnitude spectrum, and the magnitude spectrum resulting from the Fourier transform of the previous magnitude spectrum ($N = 2048$, but only the first 256 bins are displayed). One can clearly see in this spectrum the prominent peak corresponding to the fundamental frequency of the original sound.*

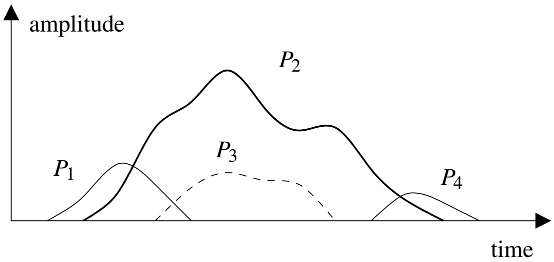

Figure 4: *The strongest partial (P_2) among the dominant partials (P_1, P_2, and P_4). P_3 is dominated by P_2.*

Figure 4. Each partial corresponds to a certain pitch candidate, and contains the evolutions in time of its frequency and amplitude parameters. In order to detect the right pitch, we have to choose the right partial in this set.

When two partials overlap at a certain time t – such as P_1 and P_2 in Figure 4 – the partial with the greatest amplitude is said to be dominating. If this partial is longer and louder than the other, we forget the dominated partial. In Figure 4, we remove P_3 because it is always dominated by P_2. Once all dominated partials have been removed, we consider the strongest partial, which is the partial who is dominating for the longest period. In Figure 4, P_2 is the strongest partial. The frequency of the strongest partial gives the evolutions in time of the fundamental frequency of the initial sound.

5. RESULTS

We have implemented the above algorithm in our *InSpect* analysis software package [15]. This implementation is made of three main parts (see Figure 5). The first part (dashed box on this figure) is a short-term analysis module: the Fourier of Fourier transform, which computes the magnitude of the Fourier transform of the magnitude of the Fourier transform of the sound signal. The local maxima (peaks) in the resulting short-term "spectra" are then tracked from frame to frame using a classic partial-tracking algorithm (second part). The third part consists in selecting the strongest partial (see Section 4) among all these tracks. The evolution in time of the frequency of this partial coincides with the pitch – as a function of time – of the initial sound.

5.1. Performance

This algorithm is much faster than the well-known autocorrelation method. Arfib and Delprat use in [9] the real part of the inverse FFT of the sound spectrum modulus limited to the positive frequency. This is strictly equivalent to the autocorrelation of the windowed part of the signal, but much faster. Our method is as fast as this one. Both methods require the computation of two Fourier transforms.

5.2. Accuracy

Perhaps surprisingly, our method is more accurate than the one used by Arfib and Delprat. Let F_{ref} be the exact fundamental frequency and F its measured value. The relative error e is given by:

$$e = |F - F_{\mathrm{ref}}| / F_{\mathrm{ref}} \tag{6}$$

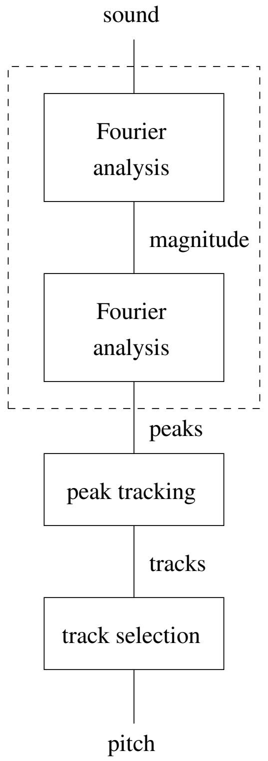

Figure 5: *Algorithm overview.*

Since our algorithm – as many others – fails in the case of a single sinusoid, let us take as a reference for our tests the sound consisting of the fundamental (with amplitude 0.75) and its first harmonic (with amplitude 0.25), with a sampling rate of $F_s = 44100$ Hz. The number of samples per analysis frame is $N = 1024$. Figure 6 shows that the relative error for the Fourier of Fourier transform goes from approximatively 1% to 6% for fundamental frequencies between 440 Hz to 1660 Hz. With the method used by Arfib and Delprat, we have measured that the relative error goes from approximatively 5% to 12% for the same frequency interval. The difference between the two methods may seem quite small. But even this small difference of 6% corresponds to approximately one half-tone...

The accuracy of the Fourier of Fourier transform can be increased by using the order-1 Fourier transform instead of the first Fourier transform (see Section 4). It is then possible to tune the accuracy (or, on the contrary, the performance) by adjusting the size of the second Fourier transform.

However, if we consider the relative error measured on a single sinusoid with the classic Fourier transform (see Figure 7), we notice that this error is lower than for the Fourier of Fourier transform for frequencies above approximatively 1000 Hz. It might be wiser to use the classic Fourier transform instead of the Fourier of Fourier transform in order to detect high pitches. Moreover, if we consider the same relative error measured for the order-1 Fourier transform (see Figure 7), we clearly see that this error is very low, even for low frequencies. This opens up new horizons for other pitch-detection algorithms.

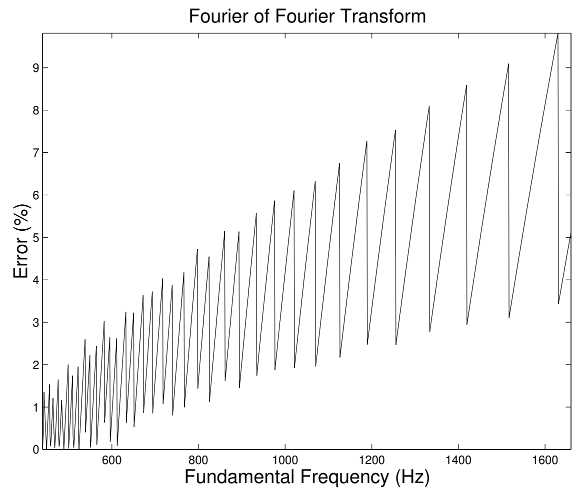

Figure 6: *Accuracy of the Fourier of Fourier transform. The relative error in percents is given for fundamental frequencies between 440 Hz and 1660 Hz (2 octaves).*

5.3. Robustness

By considering the peak with the greatest amplitude in the Fourier of Fourier transform, it is possible to perform the pitch detection in real time. The problem is that the resulting algorithm is not robust.

The technique consisting in constructing partials and selecting the strongest of them (see Section 4) has proven to be a very robust way to obtain the pitch of the sound. We have successfully recovered the pitches of many natural sounds like saxophones, guitars or singing voice for example. With this technique, there are no more jumps among octaves. The problem is that the resulting pitch-detection algorithm does not work in real time anymore.

6. CONCLUSION AND FUTURE WORK

In this article, we have presented a method for pitch detection based on a combination of two Fourier transforms. We have proposed a way to enhance the accuracy of the detected pitch – by using the order-1 Fourier transform – as well as a way to improve the robustness of the detection algorithm – by selecting the strongest pitch candidate. We have implemented the above algorithm in our *InSpect* analysis software package [15], and it has proven to be very accurate and robust in practice on natural sounds (voice, classic musical instruments, and even some kinds of noise).

During this research, we have identified the need for a standard set of tests in order to compare the numerous pitch-tracking algorithms. Further research should include the generalization of the pitch-detection methods for polyphonic sounds, thus leading to the extraction of multiple pitches, which is of great musical interest.

7. ACKNOWLEDGMENTS

This research was carried out in the context of the SCRIME (*Studio de Création et de Recherche en Informatique et Musique Electroacoustique*) and was supported by the *Conseil Régional d'Aquitaine*, the *Ministère de la Culture*, the *Direction Régionale des Actions Culturelles d'Aquitaine*, and the *Conseil Général de la Gironde*.

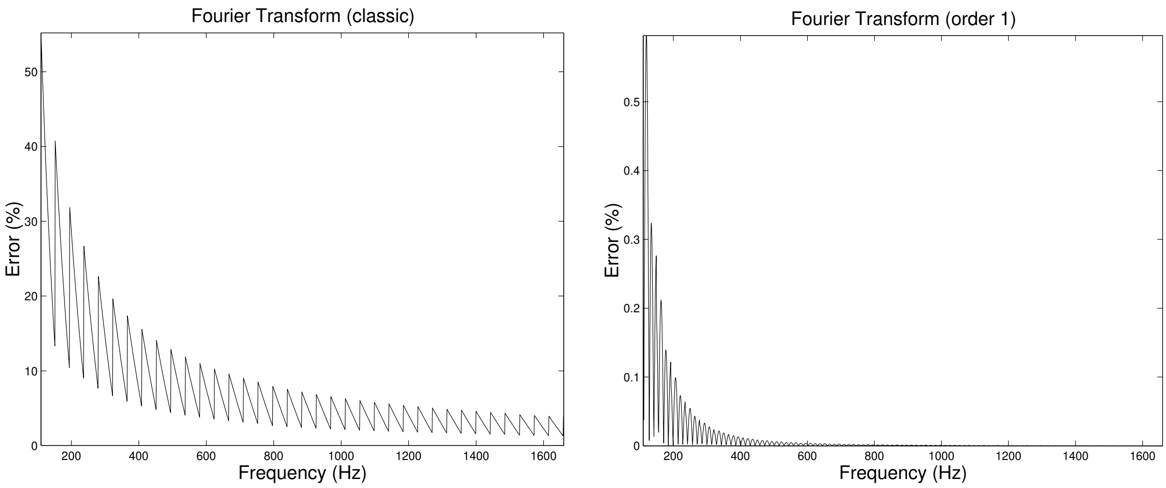

Figure 7: *Accuracy of the classic Fourier transform (left) and the order-1 Fourier transform (right). The relative error in percents is given for frequencies between 110 Hz and 1660 Hz (4 octaves).*

I would like to thank Florian Keiler and Giuliano Monti for the fruitful discussions we had during the COST short-term mission on pitch detection held in Bordeaux at the beginning of July 2001. Some pieces of code developed in common during this meeting were also used in this article in order to measure the error rates of the different pitch-detection methods.

8. REFERENCES

[1] Geoffroy Peeters, "Analyse-Synthèse des sons musicaux par la méthode PSOLA," in *Proceedings of the Journées d'Informatique Musicale (JIM)*, Toulon, 1998, In French.

[2] Lawrence R. Rabiner, "On the Use of Autocorrelation Analysis for Pitch Detection," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 25, no. 1, pp. 24–33, February 1977.

[3] J. C. Brown and M. S. Puckette, "A High Resolution Fundamental Frequency Determination Based on Phase Changes of the Fourier Transform," *Journal of the Acoustical Society of America*, vol. 94, no. 2, pp. 662–667, 1993.

[4] John E. Lane, "Pitch Detection Using a Tunable IIR Filter," *Computer Music Journal*, vol. 14, no. 3, pp. 46–59, Fall 1990.

[5] David Cooper and Kia C. Ng, "A Monophonic Pitch-Tracking Algorithm Based on Waveform Periodicity Determinations Using Landmark Points," *Computer Music Journal*, vol. 20, no. 3, pp. 70–78, Fall 1996.

[6] Andrew Choi, "Real-Time Fundamental Frequency Estimation by Least-Square Fitting," *IEEE Transactions on Speech and Audio Processing*, vol. 5, no. 2, pp. 201–205, March 1997.

[7] J. C. Brown, "Musical Fundamental Frequency Tracking Using a Pattern Recognition Method," *Journal of the Acoustical Society of America*, vol. 92, no. 3, pp. 1394–1402, September 1992.

[8] Hajime Sano and B. Keith Jenkins, "A Neural Network Model for Pitch Perception," *Computer Music Journal*, vol. 13, no. 3, pp. 41–48, Fall 1989.

[9] Daniel Arfib and Nathalie Delprat, "Alteration of the Vibrato of a Recorded Voice," in *Proceedings of the International Computer Music Conference (ICMC)*, Beijing, China, October 1999, International Computer Music Association (ICMA), pp. 186–189.

[10] Sylvain Marchand, "Improving Spectral Analysis Precision with an Enhanced Phase Vocoder Using Signal Derivatives," in *Proceedings of the Digital Audio Effects (DAFx) Workshop*, Barcelona, Spain, November 1998, Audiovisual Institute, Pompeu Fabra University and COST (European Cooperation in the Field of Scientific and Technical Research), pp. 114–118.

[11] Myriam Desainte-Catherine and Sylvain Marchand, "High Precision Fourier Analysis of Sounds Using Signal Derivatives," *Journal of the Audio Engineering Society*, vol. 48, no. 7/8, pp. 654–667, July/August 2000.

[12] Rasmus Althoff, Florian Keiler, and Udo Zölzer, "Extracting Sinusoids From Harmonic Signals," in *Proceedings of the Digital Audio Effects (DAFx) Workshop*, Trondheim, Norway, December 1999, Norwegian University of Science and Technology (NTNU) and COST (European Cooperation in the Field of Scientific and Technical Research), pp. 97–100.

[13] Damien Cirotteau, Dominique Fober, Stéphane Letz, and Yann Orlarey, "Un pitchtracker monophonique," in *Proceedings of the Journées d'Informatique Musicale (JIM)*, Bourges, June 2001, IMEB, pp. 217–223, In French.

[14] Sylvain Marchand, *Sound Models for Computer Music (analysis, transformation, synthesis)*, Ph.D. thesis, University of Bordeaux 1, LaBRI, December 2000.

[15] Sylvain Marchand, "InSpect+ProSpect+ReSpect Software Packages," Online. URL: `http://www.scrime.u-bordeaux.fr`, 2000.