A PARAMETRIC METHOD FOR PITCH ESTIMATION OF PIANO TONES

Valentin EMIYA, Bertrand DAVID, Roland BADEAU

Ecole Nationale Supérieure des Télécommunications - Département TSI
46 rue Barrault, 75634 PARIS cedex 13 - France

ABSTRACT

The efficiency of most pitch estimation methods declines when the analyzed frame is shortened and/or when a wide fundamental frequency (F_0) range is targeted. The technique proposed herein jointly uses a periodicity analysis and a spectral matching process to improve the F_0 estimation performance in such an adverse context: a 60ms-long data frame together with the whole, $7^1/_4$-octaves, piano tessitura. The enhancements are obtained thanks to a parametric approach which, among other things, models the inharmonicity of piano tones. The performance of the algorithm is assessed, is compared to the results obtained from other estimators and is discussed in order to characterize their behavior and typical misestimations.

***Index Terms*—** audio processing, pitch estimation

1. INTRODUCTION

Numerous methods dedicated to fundamental frequency (F_0) estimation of periodic signals try to extract the signal self-similarities by maximizing a function of time or frequency. In this manner, they measure a degree of internal resemblance in the waveform (ACF [1, 2], AMDF [3, 4], cepstrum [5]) or in the spectrum [6]. When processing real world musical sounds, these techniques are confronted to deviations from the theoretical model, such as the presence of noise, which can be both stationary and non stationary, or the possibly non-uniform distribution of the harmonics.

The development and applications of the quoted methods often deal with an extension to subband processing [2, 7], to an optimization of the main function [4, 7] or to the joint use of both time and frequency domains [8]. Typical errors that usually occur give a general idea of the difficulties the F_0 estimation task must cope with. Temporal or spectral methods tend to make sub-octave or octave errors respectively. Both of them come up against difficulties like a large F_0 search range (*e.g.* 27-4200 Hz for the piano), non-regular spectral envelopes and inharmonic deviations of the frequency components [6, 9]. In addition, a short analysis frame prevents spectral methods from resolving components for low F_0 values whereas the uniformely-distributed discrete time scale used by temporal methods makes the estimation fail above some F_0 limit.

The new F_0 estimation algorithm we describe aims at enhancing F_0 estimation results in the case of a short analysis window and a large F_0 search range. We will focus on piano sounds since they present all the listed difficulties and usually cause one of the worst estimation error rates per instrument (*e.g.* see [8]). The pitch of a harmonic or quasi-harmonic sound is an attribute that only depends on the sinusoidal components of the signal. Thus a F_0 estimator only requires the parameters of components such as frequency, amplitude, damping factor and initial phase. So far, the other part of the sound, including the ambient noise, transients, etc. has not been used in the F_0 estimation task, as far as the authors know. Therefore, the preliminary task in the F_0 estimation method we present consists in extracting the parameters of components. The F_0 estimator then includes a spectral function and a temporal function. The parametric approach allows to take into account the inharmonicity of sounds both in time and frequency domains and to optimize the precision of the F_0 numeric estimation.

The F_0 estimation system is described in section 2. Evaluation results and comparisons with other algorithms are then detailed in section 3 and conclusions are finally presented in section 4.

2. PITCH ESTIMATION SYSTEM

2.1. High Resolution analysis

The N_a-length analyzed waveform is modeled by:

$$s(t) \quad = \quad \sum_{k=1}^{K} \alpha_k z_k^t + w(t) \qquad (1)$$

defined for $t \in [\![0, N_a - 1]\!]$ and composed of a sum of K exponentially-modulated sinusoids $\alpha_k z_k^t, k \in [\![1, K]\!]$ with complex amplitudes $\alpha_k = A_k e^{i\Phi_k} \in \mathbb{C}^*$, ($A_k$ being the real, positive amplitude and Φ_k the initial phase), and distinct poles $z_k = e^{d_k + i2\pi f_k}$ (f_k being the frequency and d_k the damping factor), plus an additive colored noise $w(t)$. This section details how the signal is preprocessed, how poles z_k are then estimated via the ESPRIT (Estimation of Signal Parameters via Rotational Invariance Techniques) algorithm [10], and how amplitudes α_k are finally extracted.

Preprocessing. A two-step preprocessing stage is applied to the signal sampled at 32 kHz:

1. The cubic computational cost of the ESPRIT algorithm is reduced when the number of poles to be estimated is low. This is achieved by using a filter bank. The signal is splitted into $D = 32$ subbands with width 500-Hz by using cosine-modulated filters [11]. The order of magnitude of the computational cost drops from N_a^3 to N_a^3/D^2 (N_a^3/D^3 per band) leading to a satisfactory processing time for the analysis bloc.
2. Components of piano sounds are particularly well represented by the exponential sinusoidal plus noise model introduced in (1). However, the ESPRIT algorithm only applies to the restrictive case of white noise. Thus, the second preprocessing step consists in whitening the noise in each subband thanks to an AR filter estimated on the smoothed spectrum of the signal.

ESPRIT algorithm. The signal in each preprocessed subband is a sum of exponentially-modulated sinusoids plus white noise. Assuming the number of poles is known, the ESPRIT algorithm [10]

The research leading to this paper was supported by the French GIP ANR under contract ANR-06-JCJC-0027-01, Décomposition en Éléments Sonores et Applications Musicales - DESAM

gives an estimation of those poles. The method is based on a subspace projection on the so-called signal subspace and benefits from the rotational invariance property of this signal subspace.

Estimation of the number of poles. In the current application, the number of poles in each subband is not known a priori and thus must be estimated. The ESTER [12] algorithm establishes a criterion $J(p)$ that provides an estimation of the number of poles as $\text{argmax}_{p\in P}\left(J(p) > \delta_J\right)$, P being the set of candidates for the number of poles and δ_J a threshold tuned to $\delta_J = 10$ in the current study. The result obtained by this method is either correctly estimated, or slightly over-estimated. As shown in [12], the latter case is not disturbing for the ESPRIT analysis, and weak amplitudes are estimated for the spurious poles.

Estimation of amplitudes. Once the poles extracted, amplitudes are estimated by a least squares algorithm applied to the subband signal. The effects of the preprocessing stage on the amplitudes in each subband are corrected by applying the inverse filters of the various preprocessing steps – whitening, filter bank and preemphasis filter series –, leading to the estimation of the amplitudes α_k, $k \in [\![1, K]\!]$.

2.2. Pitch estimation

A temporal method and a spectral method are first introduced. Although each one could account for a F_0 estimator, they are jointly used in the same manner as in [8] to obtain the whole, more efficient estimator detailed in the last part.

2.2.1. Temporal method

Periodicity is often analyzed by assuming the signal is an observation of a real, wide-sense stationary (WSS) process y and by estimating its autocovariance function $R_y(\tau) = \mathbb{E}\left[y(t)y(t+\tau)\right]$. When the signal is periodic, the maxima of $R_y(\tau)$ are located at $\tau = 0$ and at every multiple of the period. Let us consider a real, WSS process y composed of K undamped sinusoids with frequencies ν_k, real amplitudes $2a_k$, initial phases φ_k, which are assumed to be independant and uniformely distributed along $[0, 2\pi[$. The autocovariance function of y is $R_y(\tau) = \sum_{k=1}^{K} 2a_k^2 \cos\left(2\pi\nu_k\tau\right) + \delta(\tau)\sigma_{w_y}^2$. Therefore we can define a temporal function $R(\tau)$ for F_0 estimation from the parameters estimated by the high resolution analysis:

$$R(\tau) = \sum_{k=1}^{K} p_k \cos\left(2\pi f_k \tau\right) \tag{2}$$

$$p_k = \begin{cases} \left|\alpha_k\right|^2 & \text{if } |z_k| = 1 \\ \frac{|\alpha_k|^2}{N_a}\frac{1-|z_k|^{2N_a}}{1-|z_k|^2} & \text{otherwise} \end{cases} \tag{3}$$

where $\tau > 0$, $f_k = \frac{\arg(z_k)}{2\pi}$ is the normalized frequency of component k, and the instantaneous power p_k is an estimate of coefficient $2a_k^2$ over the analysis frame.

In the case of a slightly inharmonic sound, the frequency deviation weakens or even removes the maxima of $R(\tau)$ at the multiples of the period. The inharmonicity law [13] for a piano tone of fundamental frequency f_0 causes partial h not to be located at frequency hf_0 but at $hf_0\sqrt{1+\beta(h^2-1)}$, β being the inharmonicity coefficient of the note. As illustrated in fig. 1, this frequency stretching may be inversed by remapping the set of estimated frequencies $\{f_k, k \in [\![1, K]\!]\}$ to a set of frequencies $\{g_{f_0,k}, k \in [\![1, K]\!]\}$:

$$g_{f_0,k} = \frac{f_k}{\sqrt{1+\beta\left(f_0\right)\left(h^2\left(f_0, f_k\right)-1\right)}} \tag{4}$$

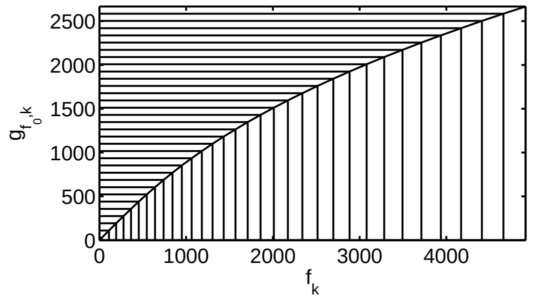

Fig. 1. At any given F_0, the frequencies f_k are remapped to $g_{f_0,k}$, leading to a harmonic distribution for the actual F_0. One theoretical partial over 5 is represented with $f_0 = 27.5$Hz and $\beta = 2.54e-4$.

where $\beta\left(f_0\right)$ is an approximative inharmonicity coefficient for fundamental frequency f_0 averaged from the results presented in [13, pp. 365]. The assumed partial order $h\left(f_0, f_k\right)$ associated to frequency f_k is extracted from the inharmonicity law:

$$h^2\left(f_0, f_k\right) = \frac{\sqrt{\left(1-\beta\left(f_0\right)\right)^2 + 4\beta\left(f_0\right)\frac{f_k^2}{f_0^2}} - 1 + \beta\left(f_0\right)}{2\beta\left(f_0\right)} \tag{5}$$

As the remapping process causes the remapped frequencies $g_{f_0,k}$ of the partials to be perfect multiples of the actual fundamental frequency f_0, we replace f_k with $g_{\frac{1}{\tau},k}$ in (2) to obtain a temporal function $R_{\text{inh}}(\tau)$ for piano tones which is maximum for $\tau = \frac{1}{f_0}$:

$$R_{\text{inh}}(\tau) = \sum_{k=1}^{K} p_k \cos\left(2\pi g_{\frac{1}{\tau},k}\tau\right) \tag{6}$$

2.2.2. Spectral method

A parametric amplitude spectrum is designed from the estimates of frequencies f_k and energies E_k of components $k \in [\![1, K]\!]$. It is composed of a sum of K gaussian curves centered in f_k with constant standard deviation σ, weighted by the square root of the component energies as average amplitudes:

$$S(f) = \sum_{k=1}^{K} \frac{\sqrt{E_k}}{\sqrt{2\pi}\sigma} e^{-\frac{(f-f_k)^2}{2\sigma^2}} \tag{7}$$

σ is set to $f_{0\min}/4$ where $f_{0\min}$ is the lower bound of the F_0 search range in order to prevent overlap between successive partials.

Our spectral estimator $U(f)$ relies on maximizing a scalar product between the parametric amplitude spectrum and theoretical harmonic unitary patterns of F_0 candidates:

$$U(f) = \sum_{h=1}^{H_f} w_{f,h} S\left(hf\right) \tag{8}$$

where H_f is the maximum number of partials possible for fundamental frequency f and $\{w_{f,h}, h \in [\![1, H_f]\!]\}$ is the pattern associated to f. The choice of the pattern is based on an approximative logarithmic spectral decrease of components. The slope p of a linear

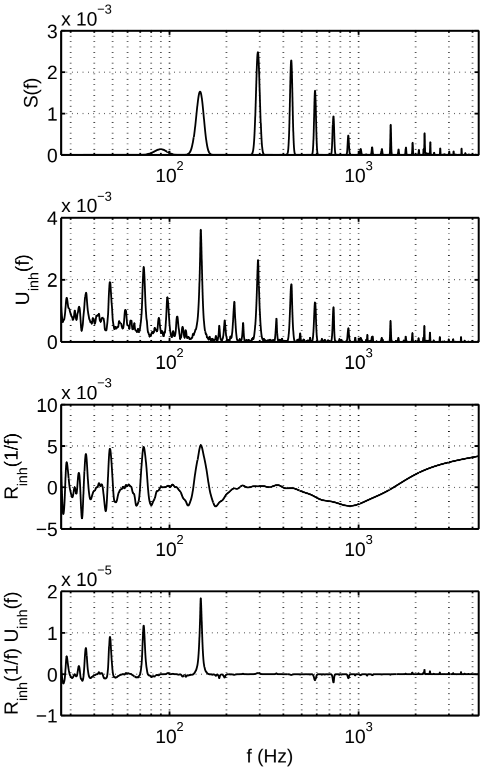

Fig. 2. From top to bottom, on a logarithmic frequency scale: parametric spectrum, spectral estimation function $U_{\text{inh}}(f)$, remapped temporal estimation function $R_{\text{inh}}\left(\frac{1}{f}\right)$, joint F_0 estimation function. Functions result from the 60 ms analysis of a D3 piano note.

regression between $\log(\sqrt{E_k})$ and f_k is extracted and weights $w_{f,h}$ are then defined as:

$$w_{f,h} \quad = \quad w_0 e^{phf} \qquad (9)$$

where $w_0 = \left(\sum_{h=1}^{H_f} e^{2phf}\right)^{-\frac{1}{2}}$ is a normalizing term such that $\sum_{h=1}^{H_f} w_{f,h}^2 = 1$.

The spectral estimator is then adapted to piano tones by selecting the values of the spectrum on an inharmonic stretched scale instead of a harmonic scale:

$$U_{\text{inh}}(f) \quad = \quad \sum_{h=1}^{H_f} w_{f,h} S\left(hf\sqrt{1+\beta(f)\left(h^2-1\right)}\right) \qquad (10)$$

Finally, the estimator efficiency can be improved by ignoring all frequencies and weights below a cut-off frequency of 100 Hz since the impedance at the piano bridge [13] causes a significant deviation of low frequencies from the inharmonicity law and the highest weigths $w_{f,h}$ of patterns are allocated to those frequencies.

2.2.3. *Pitch estimator*

As mentioned in the introduction, sub-harmonic and harmonic error trends are opposed in temporal and spectral methods. A way to benefit from this phenomenon is described in [8]. It consists in multiplying a temporal and a spectral function on a common F_0 scale in order to preserve common peaks from both functions and to remove or attenuate other peaks (see fig. 2). Thus, the pitch is estimated by maximizing the product of the methods $R_{\text{inh}}\left(\frac{1}{f}\right)$ and $U_{\text{inh}}(f)$:

$$\hat{f}_0 \quad = \quad \text{argmax}_f \left(R_{\text{inh}}\left(\frac{1}{f}\right) U_{\text{inh}}\left(f\right)\right) \qquad (11)$$

Thanks to the analytic expressions (6) and (10), $R_{\text{inh}}\left(\frac{1}{f}\right)$ and $U_{\text{inh}}(f)$ can be directly evaluated for any f value. As the F_0 distribution of an equal-tempered musical scale is logarithmic, the F_0-scale support is set to N_f points logarithmically spaced in the F_0-search range. This unconstrained choice is a key advantage of the method since the logarithmic F_0 distribution is not offered by many methods (see [4, 8]). Actually, temporal methods have a linearly distributed time scale, which results in a lack of precision in high frequency and too much resolution in low frequency, whereas Fourier-based spectral methods have a linear F_0 distribution. In those cases, the estimation function must often be interpolated to achieve the required precision and may still suffer from this.

In a Matlab implementation on a 2.4GHz-CPU, the overall processing of a 60ms frame averages 6.5s. About 1s is necessary for the analysis. About 95% of the remaining time is required by the spectral F_0 estimator and may be optimized and written in C for a computationally-efficient implementation.

3. EVALUATION

The algorithm has been evaluated on isolated piano tones from various sources: 3168 notes from three pianos of RWC database [14], 270 notes from five pianos of a PROSONUS database and 264 notes from a Yamaha upright piano of a private database. All recordings include several takes of all the 88 notes of piano range (except PROSONUS in which notes are spaced by fourth) with a varying loudness. RWC samples also offer various play modes (normal, staccato, with pedal). The F_0 search scale is composed of $N_f = 8192$ values logarithmically distributed between $f_{0\text{min}} = 26.73$ Hz and $f_{0\text{max}} = 4310$ Hz. The estimation is performed after the analysis of a single 60 ms or 93 ms frame: 60 ms is quite a challenging frame length since it is below twice the period of lowest notes while 93 ms is a well spread duration for this kind of evaluation. Each estimated F_0 is associated to the closest note in the equal tempered scale with A4 tuned to 440 Hz. Errors are then defined as incorrect note estimations. The method is compared to two estimators. The first one is as similar to our estimator as possible, replacing the ESPRIT analysis stage with a classical analysis: the ACF is estimated from the signal by the formula $r(\tau) = \frac{N_a}{N_a-\tau}\text{DFT}^{-1}\left[\left|\text{DFT}\left[s\right]\right|^2\right]$, the factor $\frac{N_a}{N_a-\tau}$ being a correction of the bias; the spectral estimator $U_{\text{inh}}(f_0)$ is computed by replacing the parametric spectrum with the modulus of the DFT of the signal, using a zero-padding on $8N_f$ points; $r(\tau)$ is mapped to the frequency scale by interpolation as described in [8]; the pitch is finally estimated by maximizing the product between the spectral function and the remapped $r(\tau)$. The second method is the YIN algorithm [4] which is considered as a very efficient monopitch estimator. We used the code available on the authors' website.

Evaluation results are reported in fig. 3. At the target window length of 60 ms, the global error rate of our estimator is around 4.4% which is at least twice better than the other estimators. This is due to a low error rate on a large F_0 range (1.1% in the F_0 range $65-2000$ Hz) and slowly increasing values at the very bass and treble limits. In comparison, the non-ESPRIT based estimator achieves

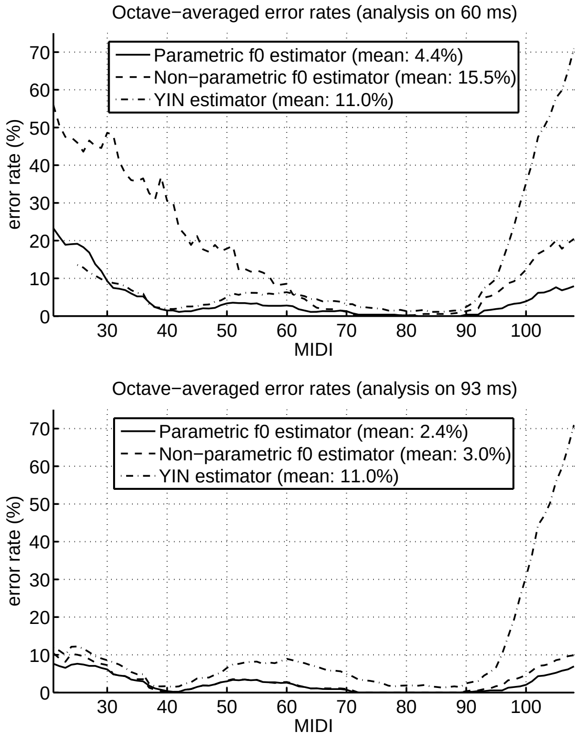

Fig. 3. Octave-averaged error rates per note with two different frame lengths, for the parametric F_0 estimator and two other methods: a similar but non-parametric algorithm and the YIN estimator

a 1.1% error rate in the range $240 - 2000$ Hz. Its low efficiency outside this range shows how the F_0 estimation is improved by both the high resolution analysis and the handling of parametric, analytic formulas. The YIN algorithm is slightly less efficient in the medium range than our estimator and has similar results in the bass range (for the first octave both curves should be at the same level, but our estimator results seem to be worse since they include the lowest four note error rates that cannot be estimated by the YIN algorithm with a 60 ms window length). In the high range, it shows a quite high error rate, which is a typical behavior of temporal methods. Global results are improved with a 93 ms frame length. Nevertheless, the high resolution analysis does not enhance significatively the F_0 estimation even if its error rate remains the lowest.

Typical errors are now discussed, in the 60 ms analysis case. As expected, usual errors are under-estimations of high f0s and over-estimations of low f0s. Around 18% of errors made by each algorithm are octave and suboctave errors. In the case of our algorithm, the remaining error intervals are of all kinds, with only 5% that are half-tone errors, whereas this rate reaches 10% for the other two algorithms. The YIN algorithm makes a high proportion of sub-harmonic errors (13% are sub-octaves, 8% are sub-nineteenth). Thus, even if our algorithm makes a reduced number of harmonic/subharmonic errors, those errors remain difficult to avoid. Half-tone error rates show the efficiency of our method while the other algorithms suffer from a lack of precision of temporal estimators for high F_0. Finally, the inharmonicity management contributes to lower the global error rate, from 4.9 to 4.4% in the 60-ms frame case. As expected, the improvement is localized in the lowest F_0 range: the error rate in the MIDI range $[\![21, 37]\!]$ decreases from 16.6 to 14.1%.

4. CONCLUSIONS

The F_0 estimator designed in this paper allows to address typical error trends in a short frame analysis and a wide F_0-range context. It is based on a preliminary extraction of the parameters of components and on the design of temporal and spectral parametric function. Satisfying performances have been obtained and a large part was allocated to the discussion on typical errors and the way to avoid them.

5. REFERENCES

[1] L. Rabiner, "On the use of autocorrelation analysis for pitch detection," *IEEE Trans. on Acoustics, Speech, and Signal Processing*, vol. 25, no. 1, pp. 24–33, 1977.

[2] Ray Meddis and Michael J. Hewitt, "Virtual pitch and phase sensitivity of a computer model of the auditory periphery. I: Pitch identification," *JASA*, vol. 89, no. 6, pp. 2866–2882, 1991.

[3] M. Ross, H. Shaffer, A. Cohen, R. Freudberg, and H. Manley, "Average magnitude difference function pitch extractor," *IEEE Trans. on Acoustics, Speech, and Signal Processing*, vol. 22, no. 5, pp. 353–362, 1974.

[4] A. de Cheveigne and H. Kawahara, "YIN, a fundamental frequency estimator for speech and music," *JASA*, vol. 111, no. 4, pp. 1917–1930, 2002.

[5] A. Michael Noll, "Cepstrum pitch determination," *JASA*, vol. 41, no. 2, pp. 293–309, 1967.

[6] A.P. Klapuri, "Multiple fundamental frequency estimation based on harmonicity and spectral smoothness," *IEEE Trans. on Speech and Audio Processing*, vol. 11, no. 6, pp. 804–816, November 2003.

[7] A.P. Klapuri, "A perceptually motivated multiple-f0 estimation method," in *Proc. of WASPAA*, New Paltz, NY, USA, October 2005, IEEE, pp. 291– 294.

[8] G. Peeters, "Music pitch representation by periodicity measures based on combined temporal and spectral representations," in *Proc. of ICASSP 2006*, Paris, France, May 14-29 2006, IEEE, vol. 5, pp. 53–56.

[9] S. Godsill and M. Davy, "Bayesian computational models for inharmonicity in musical instruments," in *Proc. of WASPAA*, New Paltz, NY, USA, October 2005, IEEE, pp. 283– 286.

[10] R. Roy, A. Paulraj, and T. Kailath, "ESPRIT–a subspace rotation approach to estimation of parameters of cisoids in noise," *IEEE Trans. on Acoustics, Speech, and Signal Processing*, vol. 34, no. 5, pp. 1340–1342, 1986.

[11] P. P. Vaidyanathan, *Multirate systems and filter banks*, Englewoods Cliffs, NJ, USA: Prentice Hall, 1993.

[12] R. Badeau, B. David, and G. Richard, "A new perturbation analysis for signal enumeration in rotational invariance techniques," *IEEE Trans. on Signal Processing*, vol. 54, no. 2, pp. 450–458, February 2006.

[13] N. H. Fletcher and T. D. Rossing, *The Physics of Musical Instruments*, Springer, 1998.

[14] T. Nishimura M. Goto, H. Hashiguchi and R. Oka, "RWC music database: Music genre database and musical instrument sound database," in *Proc. of ISMIR*, Baltimore, Maryland, USA, 2003, pp. 229–230.