Master M2 - DataScience

Audio and music information retrieval

Lecture on
Pitch and Multipitch estimation

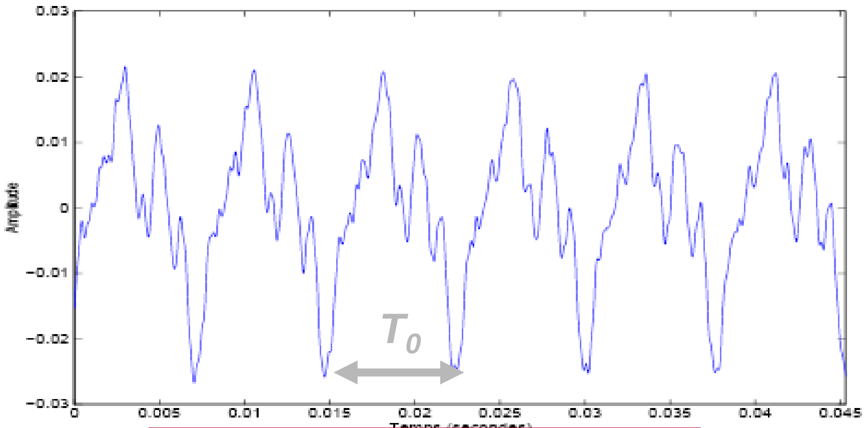
Gaël RICHARD
Télécom Paris
March 2022

Fundamental frequency detection

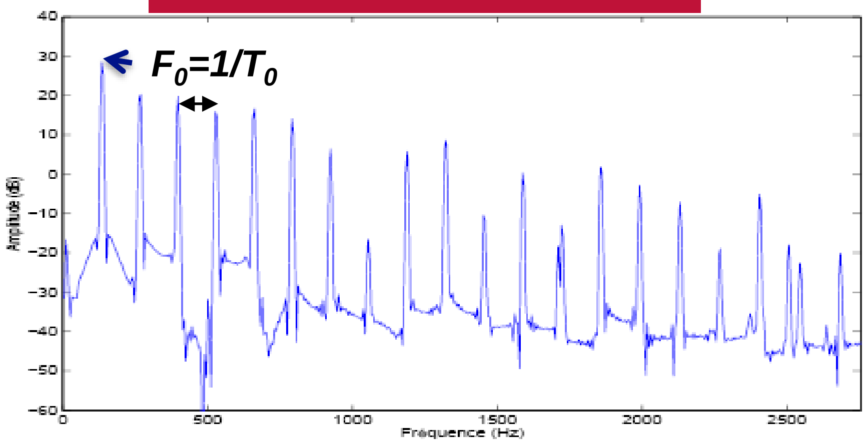
Content

- **Introduction**
  - Quasi-periodic sounds
  - Quasi-periodic model
- **Time-domain methods**
- **Spectral domain methods**
- **Extension to mutipitch (e.g. multiple fundamental frequencies) estimation**

A quasi-periodic sound



A piano sound (C3)



Spectrum of a piano sound

How can we estimate the height (pitch) of a note

or

How to estimate the **fundamental periode** (T_0)
or **frequency** (F_0) ?

Signal Model

- $$x(n) = \sum_{k=1}^{H} 2A_k \cos(2\pi k f_0 n + \phi_k) + w(n)$$
- $f_0 = \dfrac{1}{T_0}$ **normalised fundamental frequency**
- **H is the number of harmonics**
- **Amplitudes $\{A_k\}$ are real numbers > 0**
- **Phases $\{\phi_k\}$ are independant r.v. uniform on $[0, 2\pi[$**
- ***w* is a centered white noise of variance σ^2, independent of phases $\{\phi_k\}$**
- **x(n) is a centered second order process with autocovariance**

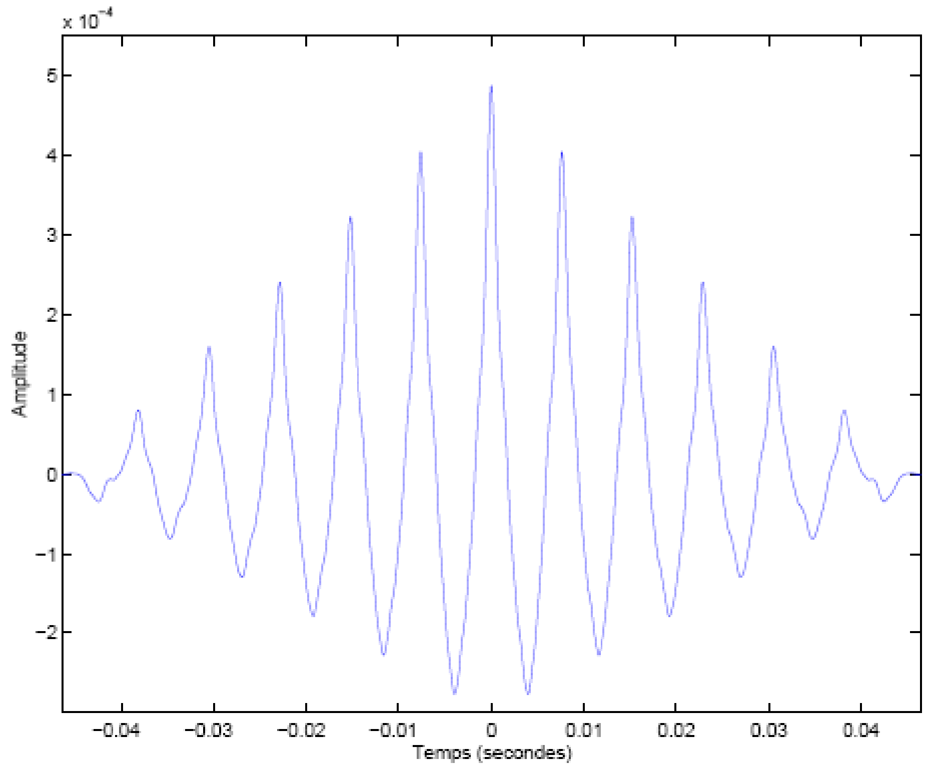
$$r_x(m) = \sum_{k=1}^{H} [2A_k^2 \cos(2\pi k f_0 m)] + \sigma^2 \delta[m]$$

Time domain methods

- **Autocovariance estimation (biased)**

$$\frac{1}{N}\sum_{n=0}^{N-1-m} x[n]\,x[n+m] \text{ si } m \geq 0$$

$$\mathbf{E}(\hat{r}_x[m]) = \frac{N-|m|}{N}\, r_x[m] \qquad\qquad |\hat{r}_x[m]| \leq \hat{r}_x[0]$$

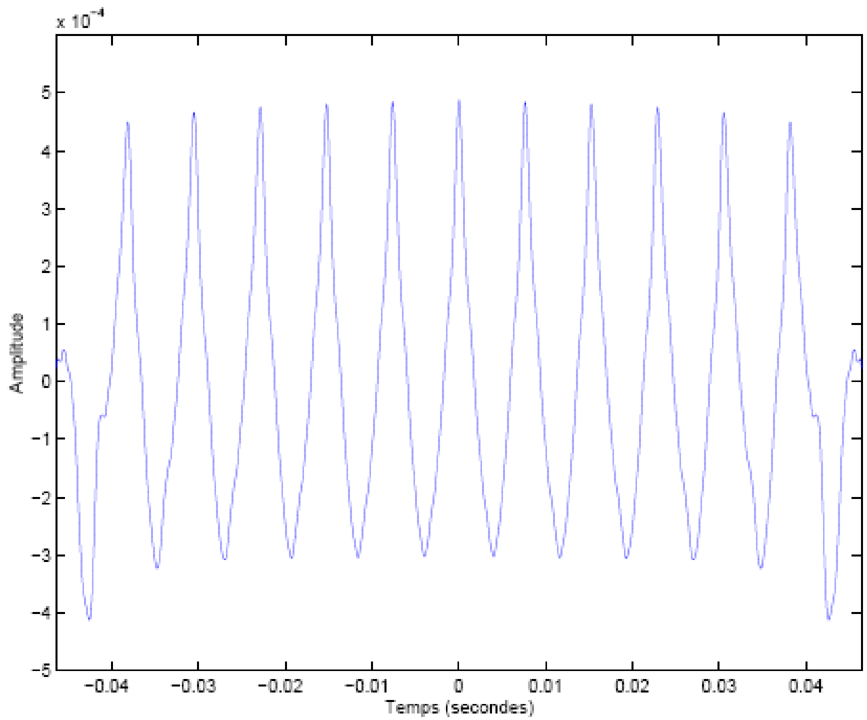


Time domain methods

- **Autocovariance estimation (unbiased)**

$$\tilde{r}_x[m] = \frac{1}{N-m} \sum_{n=0}^{N-1-m} x[n]\, x[n+m] \text{ si } m \geq 0$$

$$\mathbf{E}(\tilde{r}_x[m]) = r_x[m] \qquad \mathrm{Var}(\tilde{r}_x[m]) = (\frac{N}{N-m})^2\, \mathrm{Var}(\hat{r}_x[m])$$



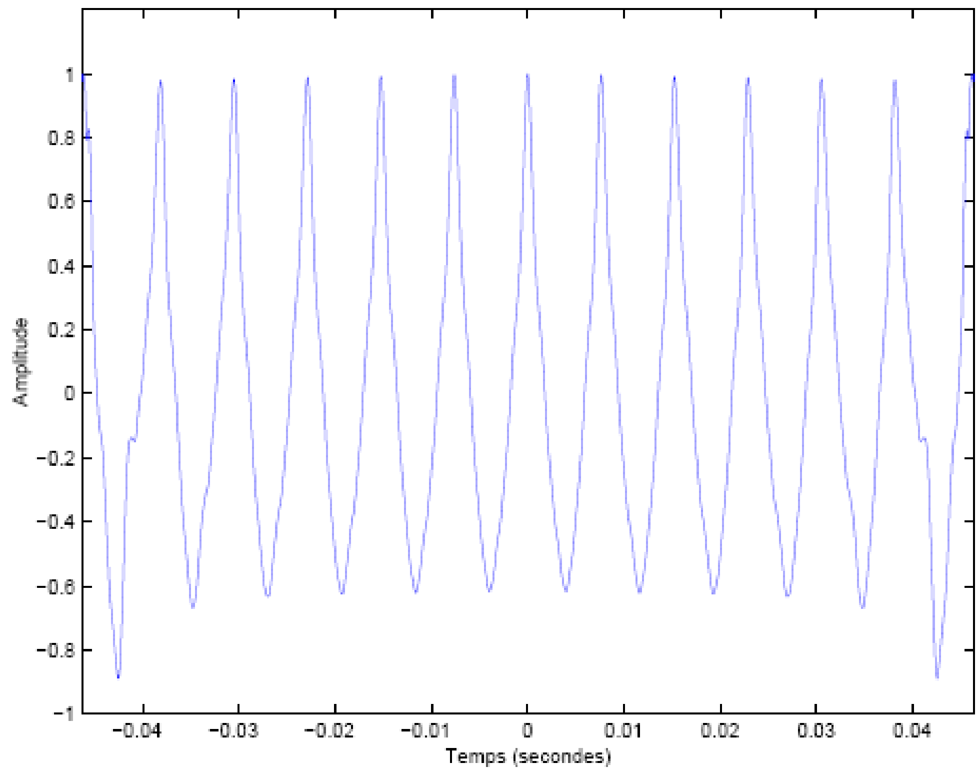
$$|\tilde{r}_x[m]| \not\leq \tilde{r}_x[0]$$

Time domain methods

- **Autocorrelation**

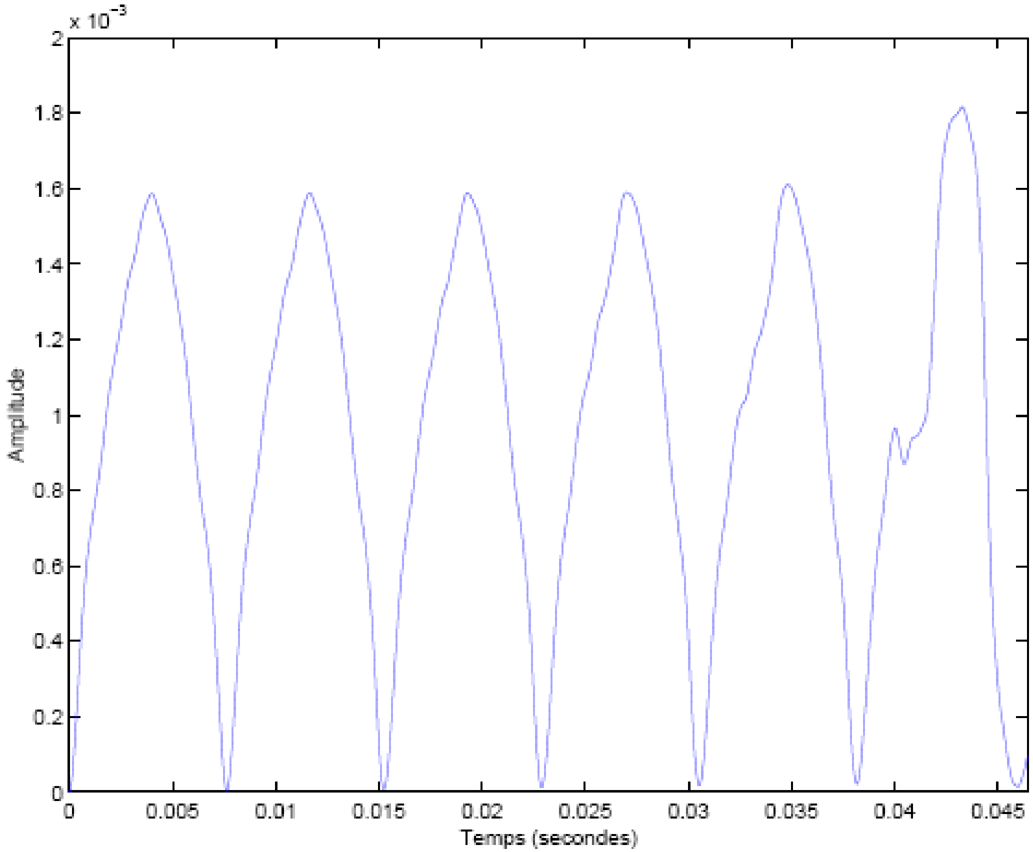
$$\bar{r}_x[m] = \frac{\sum_{n=0}^{N-1-m} x[n]\, x[n+m]}{\sqrt{\sum_{n=0}^{N-1-m} x[n]^2}\sqrt{\sum_{n=0}^{N-1-m} x[n+m]^2}} \text{ si } m \geq 0$$

$$|\bar{r}_x[m]| \leq \bar{r}_x[0] = 1 \qquad |\bar{r}_x[m]| = 1 \text{ ssi les vecteurs sont colinaires}$$



Average square difference function (ASDF)

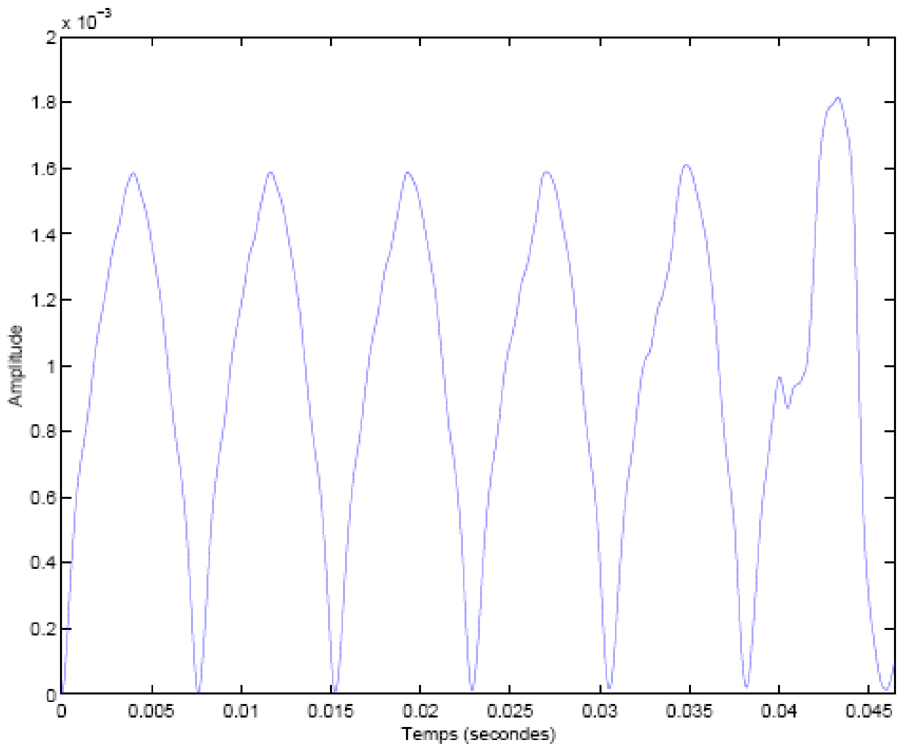
$$\mathrm{ASDF}[m] = \frac{1}{N-m}\sum_{n=0}^{N-1-m}(x[n]-x[n+m])^2$$



Average square difference function (ASDF)

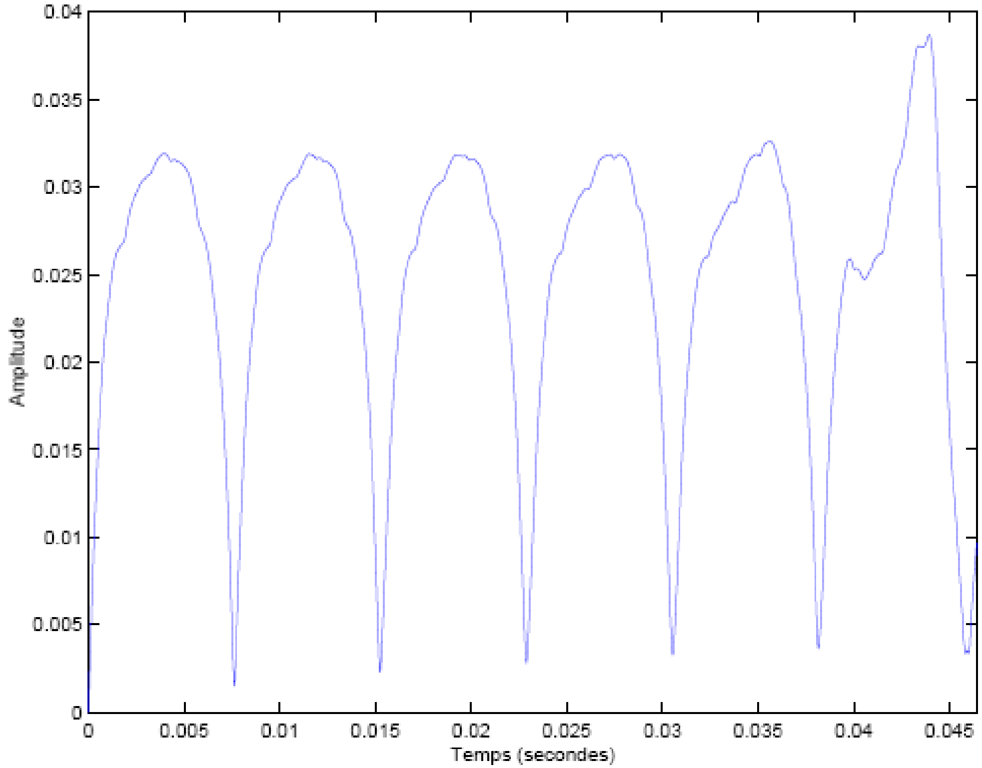
- The period T_0 can be estimated in looking at teh minimum of the square difference between $x(n)$ and $x(n-m)$:

$$\mathbf{E}[\mathrm{ASDF}[m]] = 2(r_x[0] - r_x[m])$$



Average magnitude difference function (AMDF)

$$\mathrm{AMDF}[m] = \frac{1}{N-m} \sum_{n=0}^{N-1-m} |x[n] - x[n+m]|$$



An efficient time-domain algorithm: Yin

(Thanks to V. Emiya for additionnal slides)

- **H. Kawahara A. de Cheveigné, *YIN, a fundamental frequency estimator for speech and music*,, JASA, 111(4), 2002**

- **Initial method: Autocorrelation method (ACF)**
- **Successive improvements:**
  - Use of ASDF
  - Normalisation
  - Threshold
  - Interpolation
  - Local minimisation in time

Version	Gross error (%)
Step 1	10.0
Step 2	1.95
Step 3	1.69
Step 4	0.78
Step 5	0.77
Step 6	0.50

YIN (2)

- **ASDF used:**

$$\mathrm{d}_n[m] = \sum_{k=0}^{N-1} (x_n[k] - x_n[k+m])^2$$

- **Links with autocorrelation**

$$\mathrm{d}_n[m] = r_n(0) + r_{n+m}(0) - 2r_n(m)$$

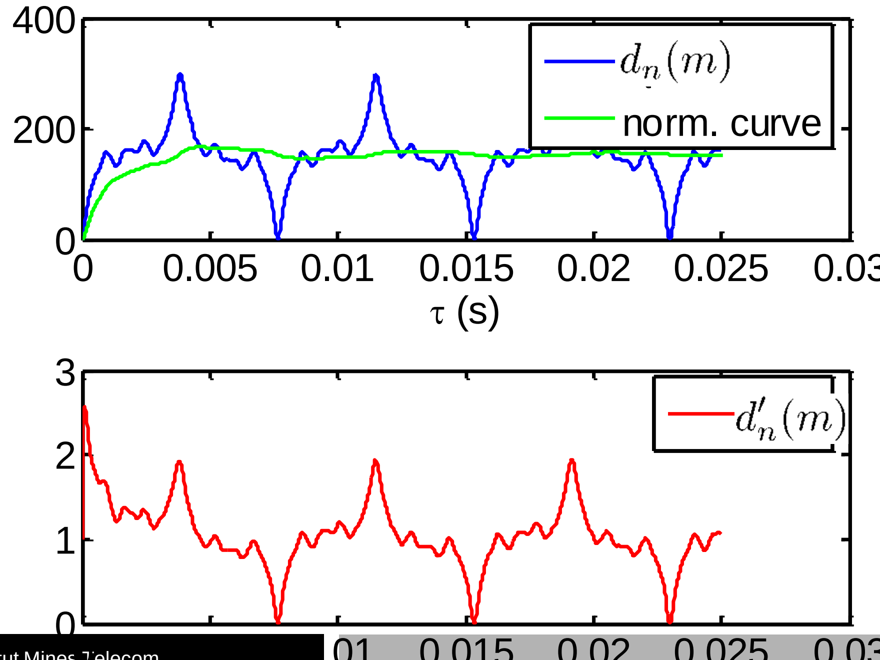
- **Performance increase : ASDF is less sensitive to amplitude variations** (e.g. ACF is sensitive to even harmonics accentuation)

YIN (3)

- **Normalisation by the « cumulative mean »**

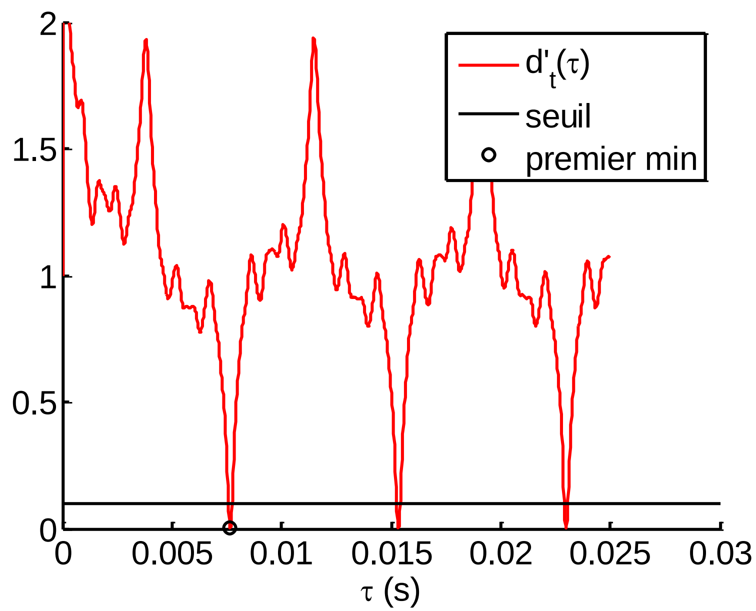
$$d'_n(m)(= \begin{cases} 1 & \text{si} \quad m = 0 \\ \dfrac{d_n(m)}{\frac{1}{m}\sum_{k=1}^{m} d_n(k)} & \text{sinon} \end{cases}$$

- **Performance increase: suppression of the main lobe at 0**



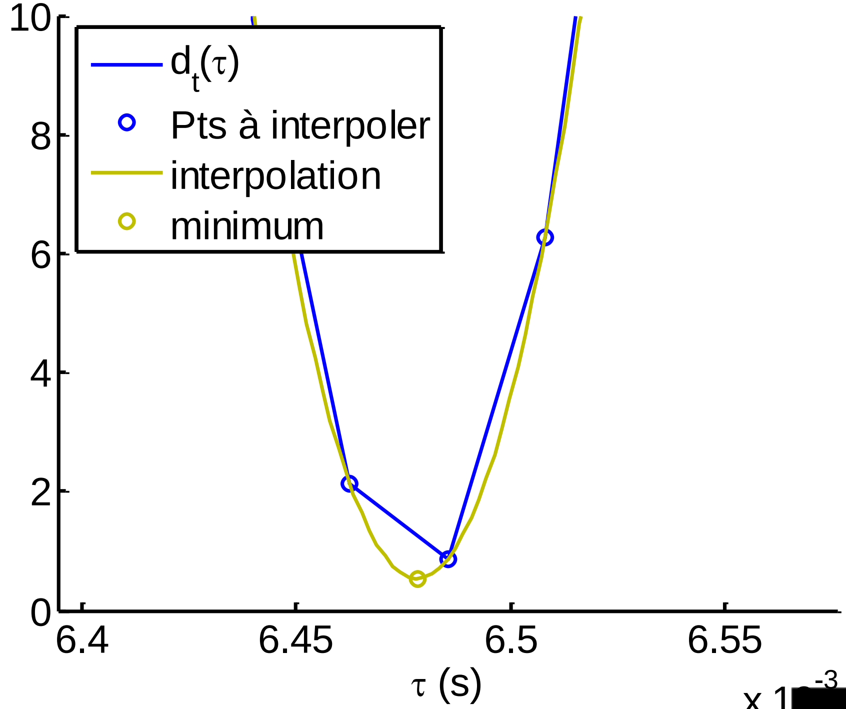
YIN (4)

- **Absolute threshold**
  - The smallest period below the threshold is chosen
  - If no period is below the threshold, the global minimum is chosen



YIN (4)

- **Parabolic interpolation around the minimum**
  - ⇨ Applied on $d_n(m)$ (i.e before normalisation)
  - ⇨ Performance increase: precision on F0



YIN (5)

- **Local minimisation in time**

$$T_n = argmin_n(d'_n(m))$$

  - **Minimisation around timeT_θ:** $argmin_\theta(d'_\theta(T_\theta))$ **with**

$$t - T_{max} \quad < \theta \quad < t + T_{max}, \qquad\qquad T_{max} = 25ms$$

$$0.8T_n \quad < T_\theta \quad < 1.2T_n$$

- **Performance increase in case of fluctuation (it is a kind of smoothing, a bit similar to median filtering)**

YIN: Evaluation

- **On four speech databases, automatically annotated by YIN (from the laryngograph signal) then manually checked**

	Gross error (%)					
Method	DB1	DB2	DB3	DB4	Average	(low/high)
pda	10.3	19.0	17.3	27.0	**16.8**	(14.2/2.6)
fxac	13.3	16.8	17.1	16.3	**15.2**	(14.2/1.0)
fxcep	4.6	15.8	5.4	6.8	**6.0**	(5.0/1.0)
ac	2.7	9.2	3.0	10.3	**5.1**	(4.1/1.0)
cc	3.4	6.8	2.9	7.5	**4.5**	(3.4/1.1)
shs	7.8	12.8	8.2	10.2	**8.7**	(8.6/0.18)
acf	0.45	1.9	7.1	11.7	**5.0**	(0.23/4.8)
nacf	0.43	1.7	6.7	11.4	**4.8**	(0.16/4.7)
additive	2.4	3.6	3.9	3.4	**3.1**	(2.5/0.55)
TEMPO	1.0	3.2	8.7	2.6	**3.4**	(0.53/2.9)
YIN	0.30	1.4	2.0	1.3	**1.03**	(0.37/0.66)

Fundamental frequency estimation using a signal model: Maximum likelihood approach

- **Signal model:** $x(n) = a(n) + w(n)$
  - *a* is a deterministic model of period T_0
  - *w* is a Gaussian white noise with variance σ^2

- **Observation likelihood**

$$p(x|T_0, a, \sigma^2)) = \frac{1}{(2\pi\sigma^2)^{\frac{N}{2}}} e^{-\frac{1}{2\sigma^2}\sum_{n=0}^{N-1}(x(n)-a(n))^2}$$

- **Log-likelihood**

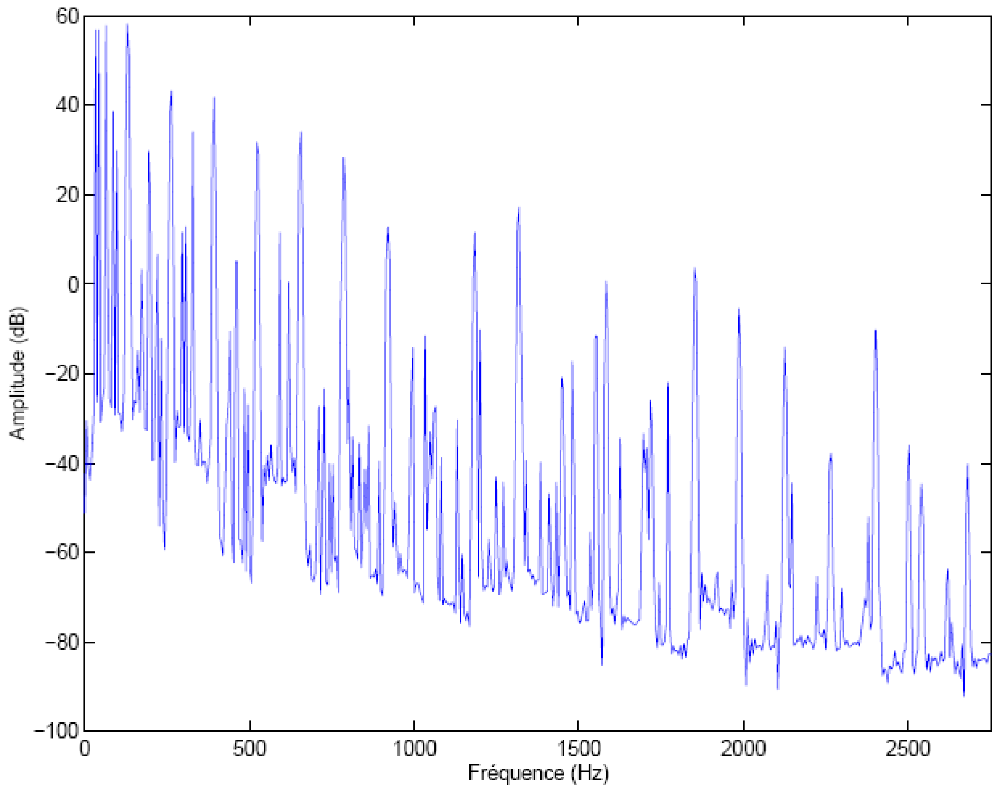
$$L(T_0, a, \sigma^2) = -\frac{N}{2}\ln(2\pi\sigma^2) - \frac{1}{2\sigma^2}\sum_{n=0}^{N-1}(x(n) - a(n))^2$$

- **Method : maximise iteratively L with respect to a, then σ^2 and then T_0**

Maximum likelihood approach

- **It can be shown that maximisation of L with respect to $F_0 = \frac{m}{N}$ is equivalent to maximise the spectral sum**

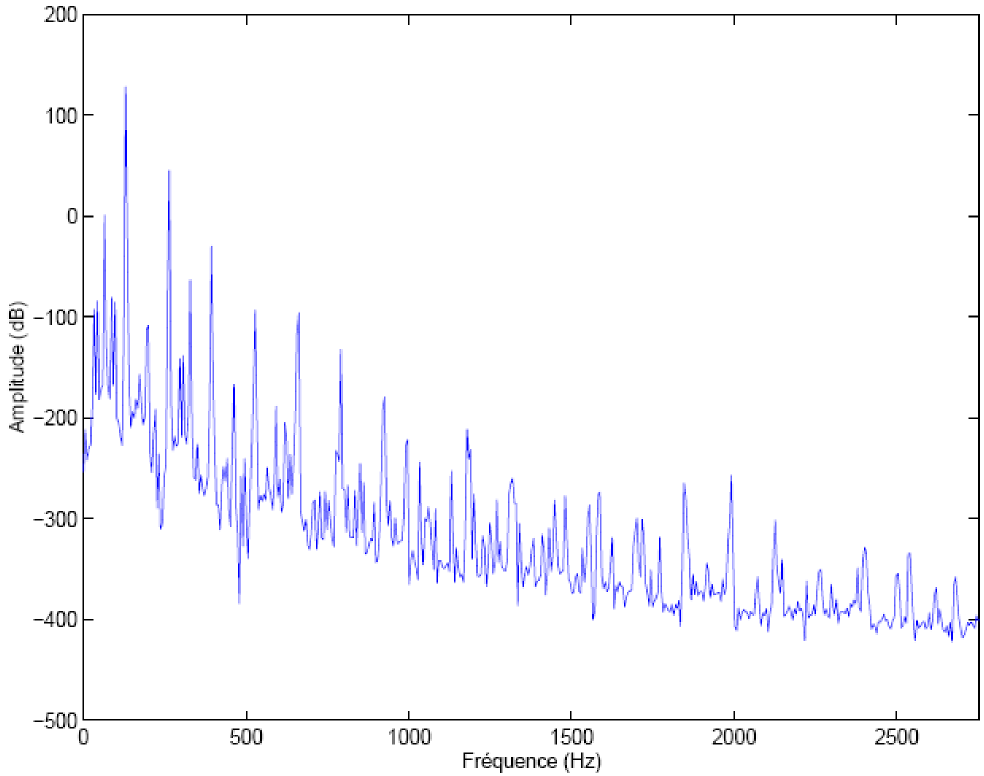
$$S(e^{j\,2\pi\,\frac{m}{N}}) = \sum_{k=1}^{H} \hat{R}_x(e^{j\,2\pi k\,\frac{m}{N}})$$



Spectral product

- **By analogy to spectral sum (often more robust)**

$$P(e^{j\,2\pi\,\frac{m}{N}}) = \prod_{k=1}^{H} \hat{R}_x(e^{j\,2\pi k\,\frac{m}{N}})$$



Multiple fundamental frequencies detection

Multiple fundamental frequencies detection

- **Objective: to estimate all musical notes of a polyphonic recording**

- **Problem: notes can be played in harmony (often the case in music …!!)**

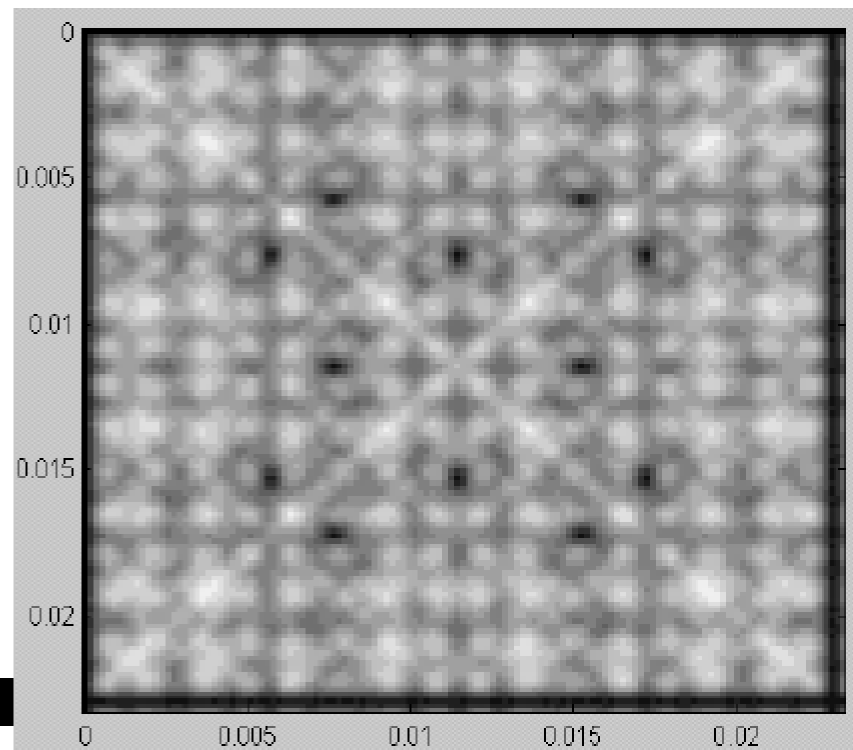
- **Sometimes: necessity to take into account the non-harmonicity of played notes**

Multiple fundamental frequencies detection

- **DMDF (*Double Magnitude Difference Function*)**

$$DMDF(k_1, k_2) = \frac{1}{N - k_1 - k_2} \sum_{n=0}^{N-k_1-k_2-1} \left| d[n] - d[n+k_1] - d[n+k_2] + d[n+k_1+k_2] \right|$$

- ✓ **piano sound**
- ✓ addition of two notes
  T1=0.0076s
  T2=0.0057s



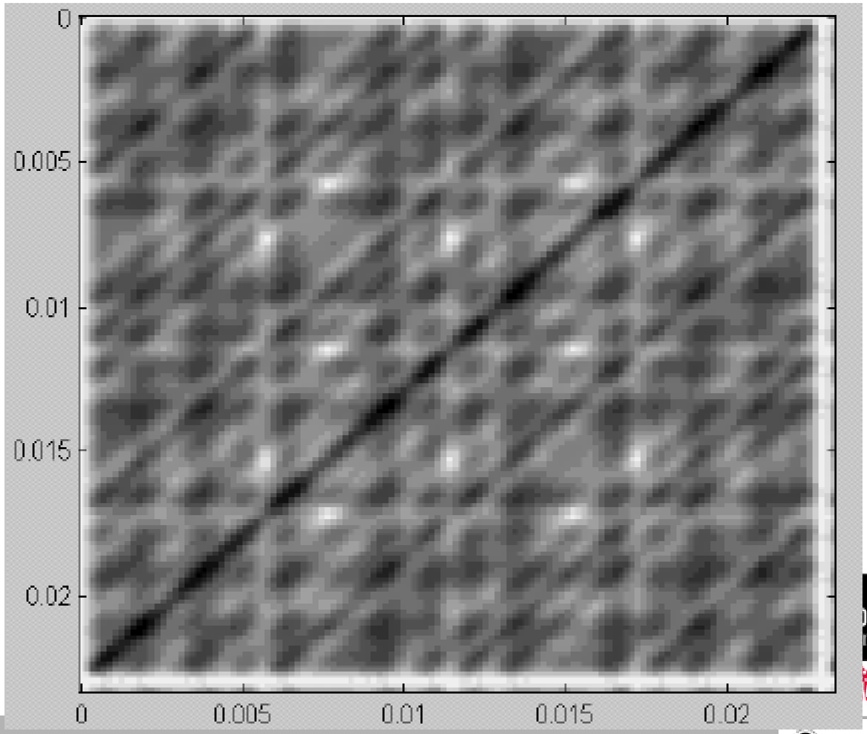
Multiple fundamental frequencies detection

- **Bi-dimensional correlation**

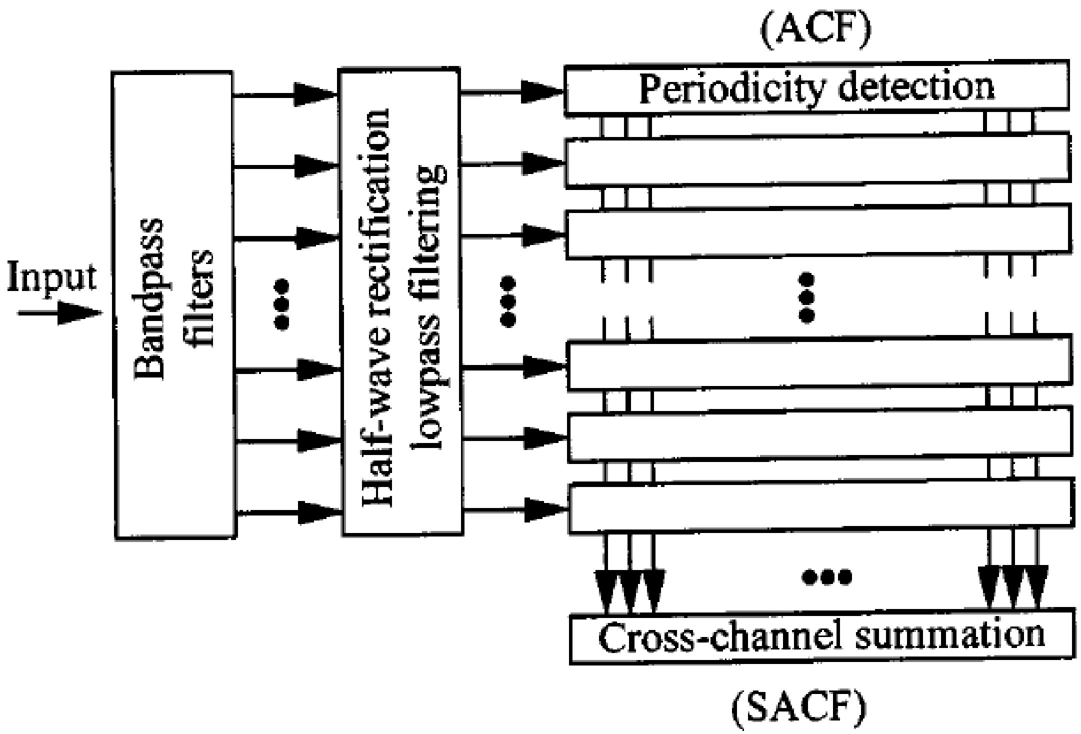
$$\bar{r}(k_1,k_2)=\frac{\sum_{n=0}^{N-k_1-k_2-1} d[n]\left(d[n+k_1]+d[n+k_2]-d[n+k_1+k_2]\right)}{\left(\sum_{n=0}^{N-k_1-k_2-1} d[n]^2\right)^{1/2}\left(\sum_{n=0}^{N-k_1-k_2-1}\left(d[n+k_1]+d[n+k_2]-d[n+k_1+k_2]\right)^2\right)^{1/2}}$$

Measures the « similarity » between

- d(n) et
- d(n+k1) + d(n+k2)-d(n+k1+k2)

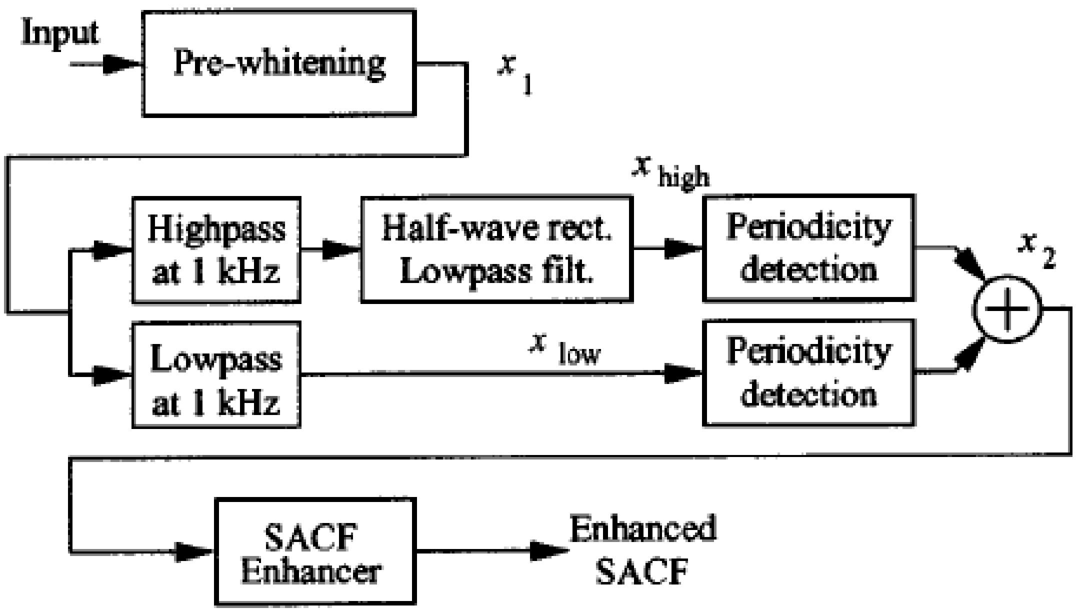


A filter banc approach



- R. Meddis and M. Hewitt, “Virtual pitch and phase sensitivity of a computer model of the auditory periphery—I: Pitch identification,” *J. Acoust. Soc. Am.*, vol. 89, pp. 2866–2882, June 1991.

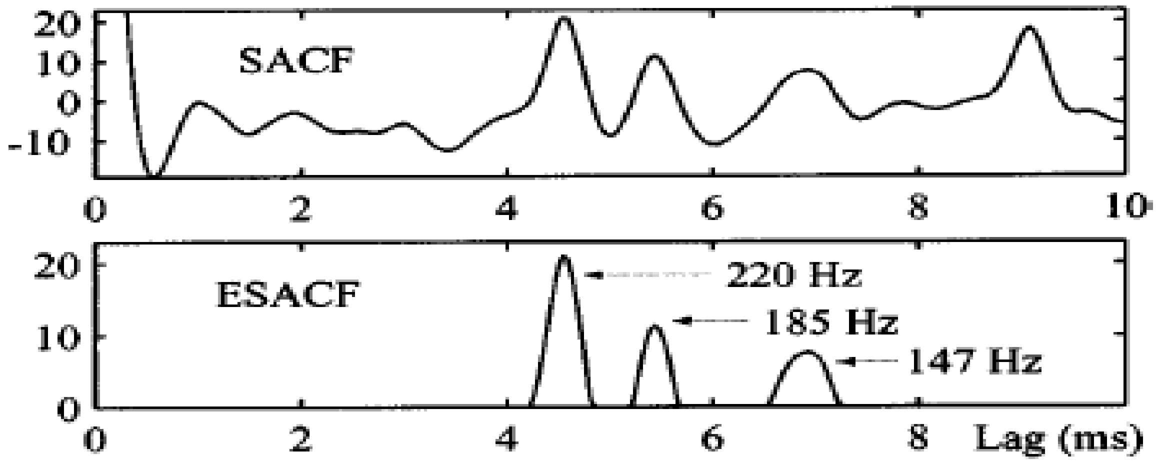
A simpler approach (inspired by the previous method)



- T. Tolonen and M. Karjalainen, "A computationally efficient multipitch analysis model," *IEEE Trans. On Speech and Audio Processing*, vol. 8, no. 6, pp. 708–716, 2000.

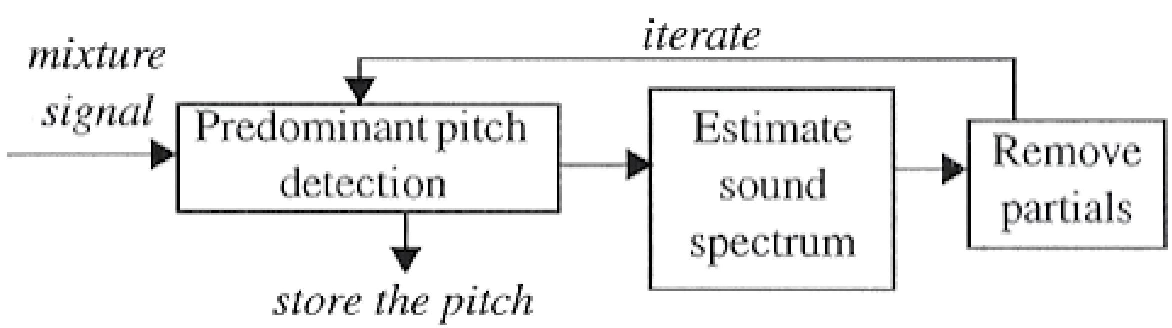
Enhanced Summary ACF

- **Several steps:**
  - **Half wave rectification**
    - We only keep positive values
  - **Slowed down twice (or more) and deduced from rectified SACF**
    - allows to suppress double pics



An iterative approach

- **Estimate each note one after the other …**
  - First, detect the most prominent note …
  - Subtract this note from the polyphony
  - Then, detect the next most prominent note
  - Subtract this note from the polyphony
  - Etc… until all notes are found

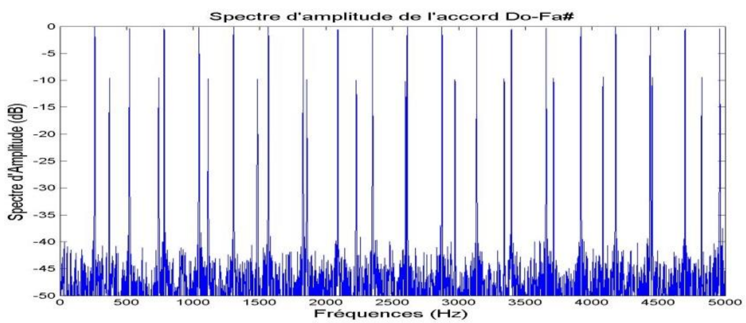


Anssi P. Klapuri, *Multiple Fundamental Frequency Estimation Based on Harmonicity and Spectral Smoothness*, IEEE Trans. On Speech and Sig. Proc., 11(6), 2003

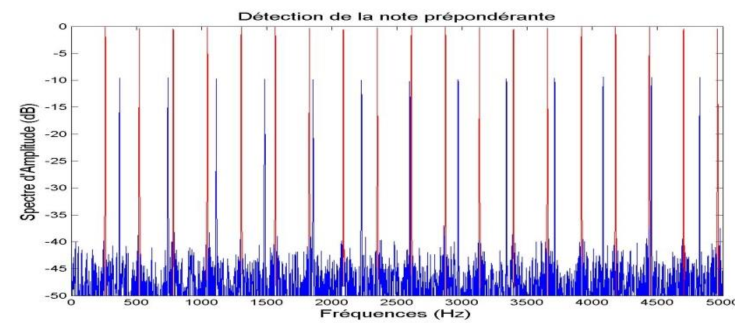
Anssi P. Klapuri "Multipitch Analysis of Polyphonic Music and Speech Signals Using an Auditory Model", IEEE Trans. On ASLP, Feb. 2008

Iterative multipitch estimation

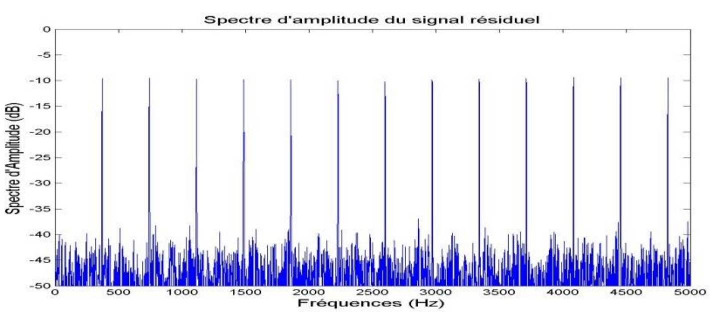
Chord of two synthetic notes C – F#



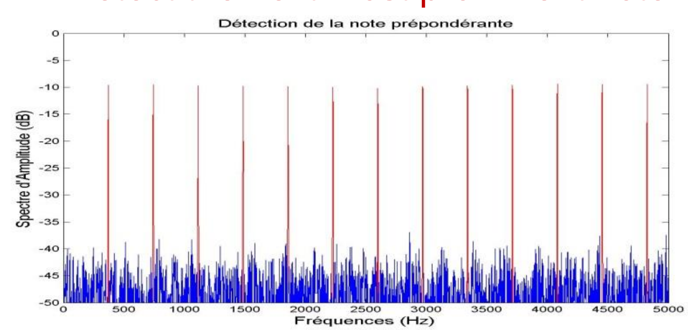
Detect the most prominent note (in red)



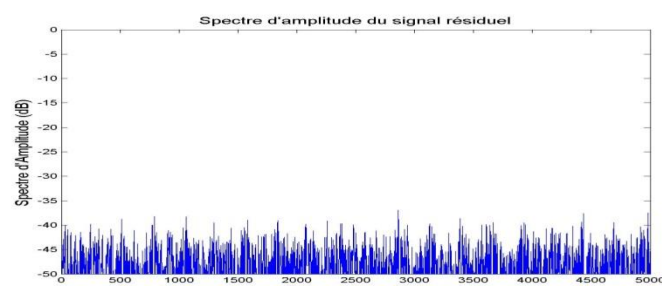
Subtract the detected note



Detect the next most prominent note



There is no more notes….chord C – F# is recognized

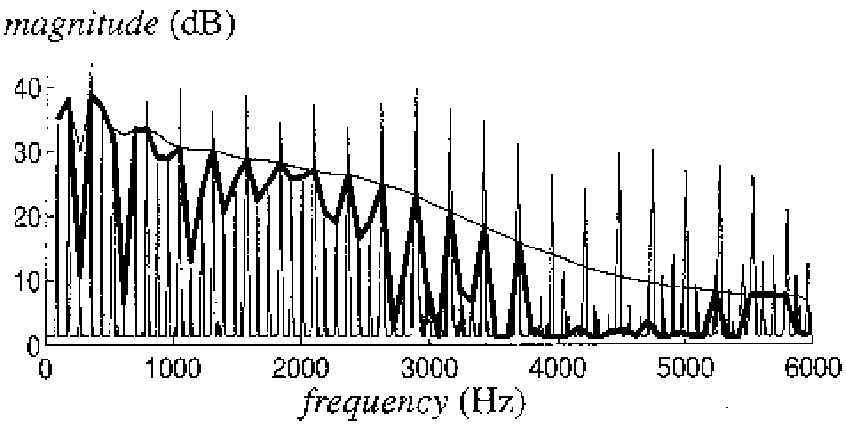


Iterative multipitch estimation

Spectral smoothing: towards subtracting only the current note

- $a_h = min(a_h, m_h)$

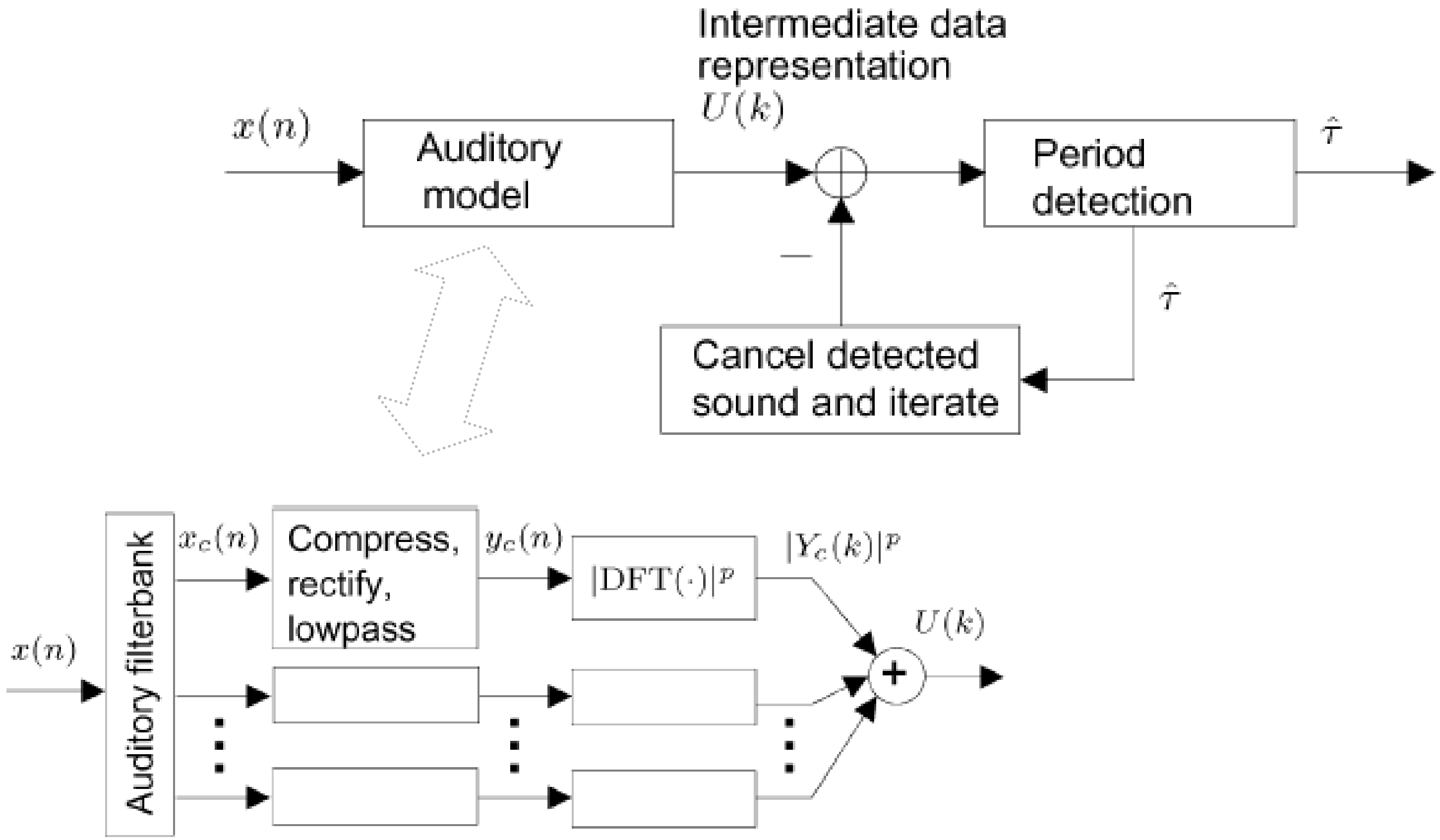
where m_h is the mean on a spectral window (*one octave wide*) around the current harmonic.



Anssi P. Klapuri, *Multiple Fundamental Frequency Estimation Based on Harmonicity and Spectral Smoothness*, IEEE Trans. On Speech and Sig. Proc., 11(6), 2003

Anssi P. Klapuri "Multipitch Analysis of Polyphonic Music and Speech Signals Using an Auditory Model", IEEE Trans. On ASLP, Feb. 2008

Improvement using a perceptual model



- **Anssi P. Klapuri "Multipitch Analysis of Polyphonic Music and Speech Signals Using an Auditory Model", IEEE Trans. On ASLP, Feb. 2008**

Multiple frequency estimation

- **Many other approaches**
  - **Bayesian methods**
  - **Factorisation methods (NMF for example)**
  - **Neural networks, Deep neural networks**

Non-Negative Factorization methods or NMF

- Use of non-supervised decomposition methods (for example Non-Negative Factorization methods or NMF)

- **Principle of NMF :**

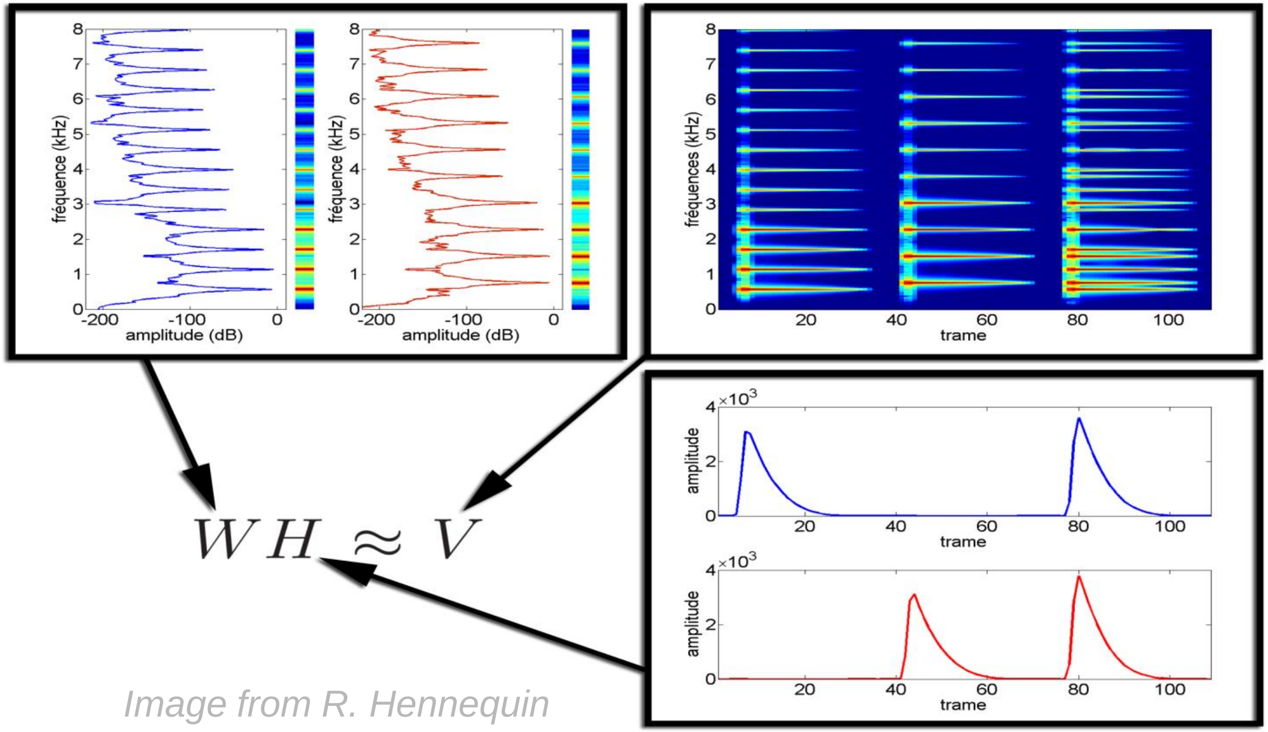
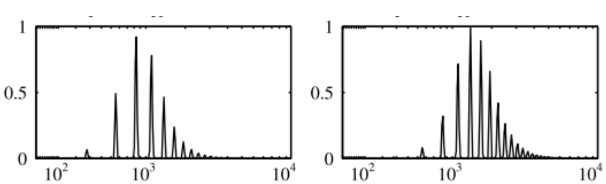


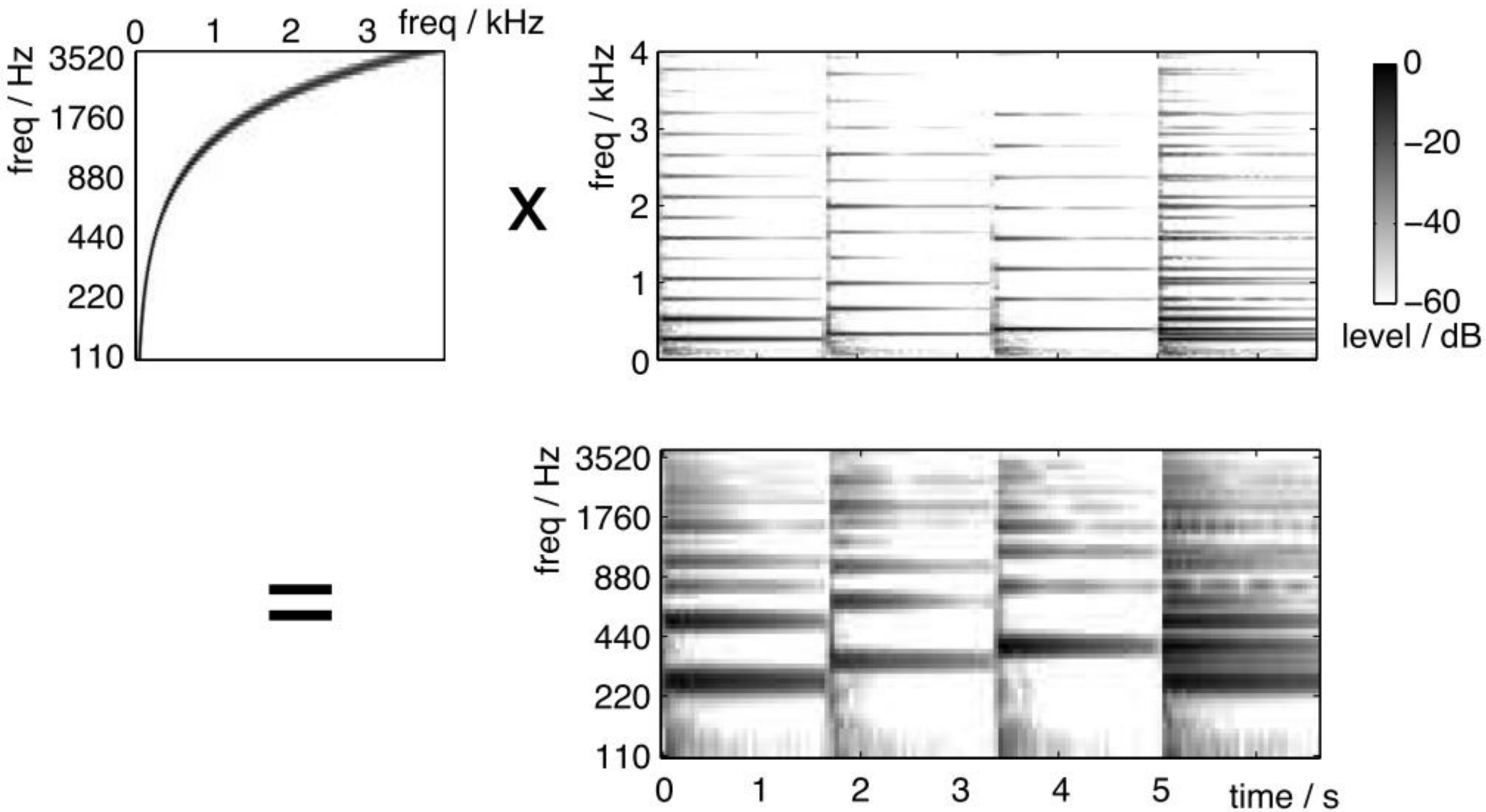
Image from R. Hennequin

Non-Negative Factorization methods or NMF

- **Use in multipitch estimation:**
  - Important to introduce *a priori* (probabilist approach) or constraints (déterminist approach)
  - Constraint examples (after Vincent & al, 2010):
    - NMF classic: $$Y_{ft} = \sum_{i=1}^{I} A_{it} S_{if}$$
    - NMF with pitch dependant templates: $$Y_{ft} = \sum_{p=p_{\mathrm{low}}}^{p_{\mathrm{high}}} \sum_{j=1}^{J_p} A_{pjt} S_{pjf}$$
    - … and template constraints $$S_{pjf} = \sum_{k=1}^{K_p} E_{pjk} N_{pkf}$$
    - Ex. With "local" envelopes



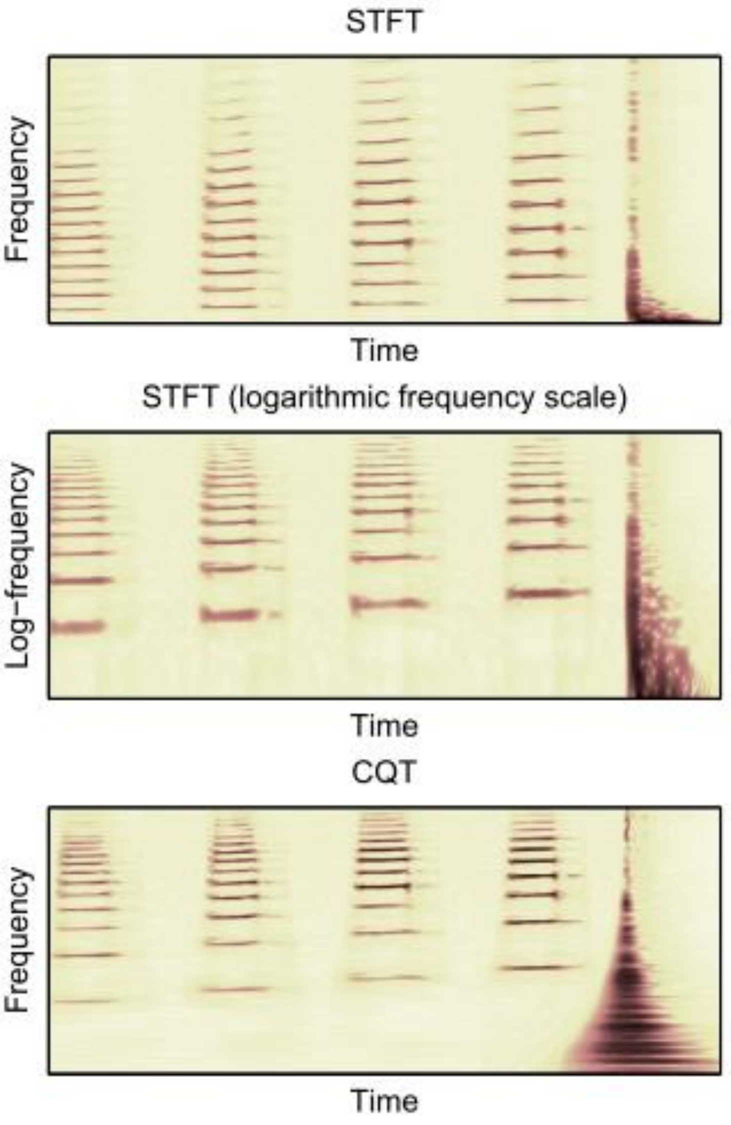
Use of a constant Q transform



D'après M. Mueller & al. « Signal Processing for Music Analysis, IEEE Trans. On Selected topics of Signal Processing, oct. 2011

Utilisation en estimation multipitch

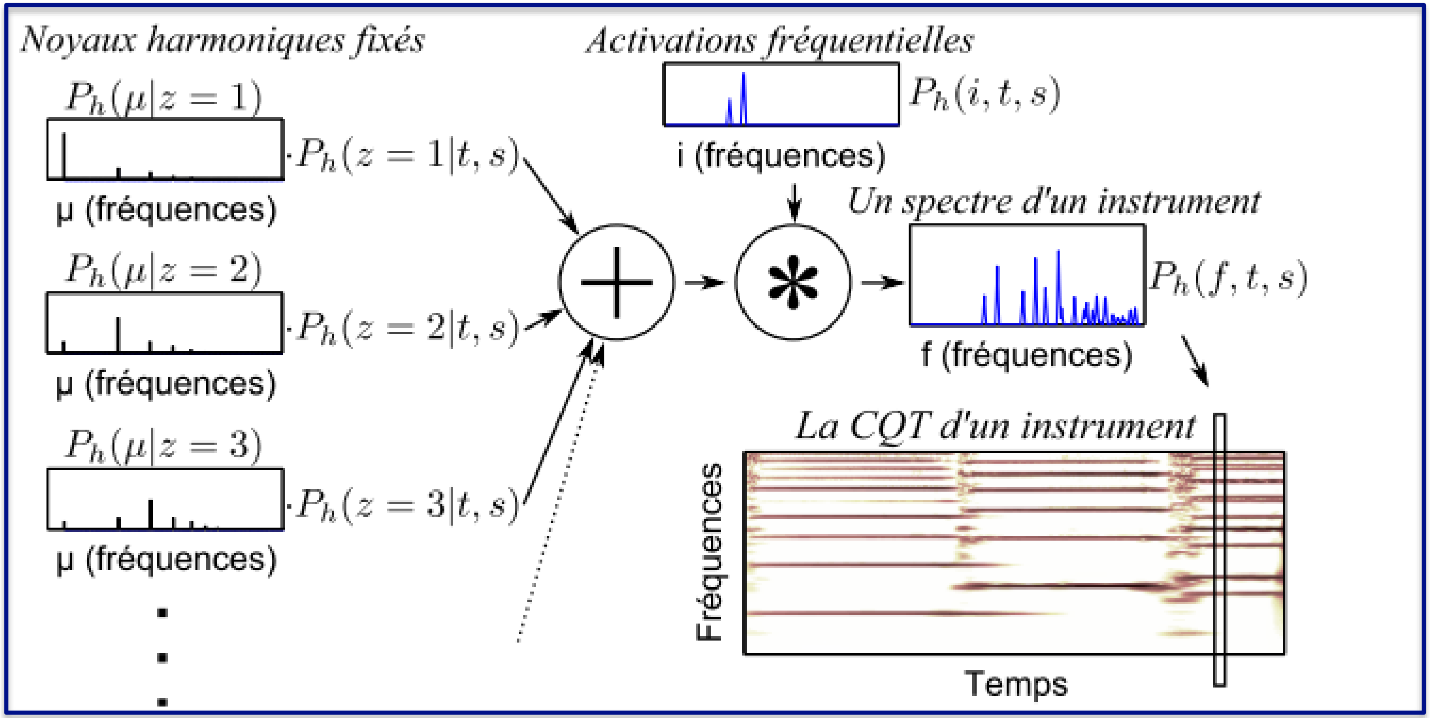
- **On a constant Q transform**
  - **A difference in pitch corresponds to a translation in frequency**
  - **Towards "Shift invariant PLCA (v. smaragdis2008 et Fuentes & al. 2011)**



A PLCA model example

- **The HALCA model (Fuentes & al.)**

$$P(f,t) = P(c=h)\boxed{P_h(f,t)} + P(c=b)P_b(f,t)$$

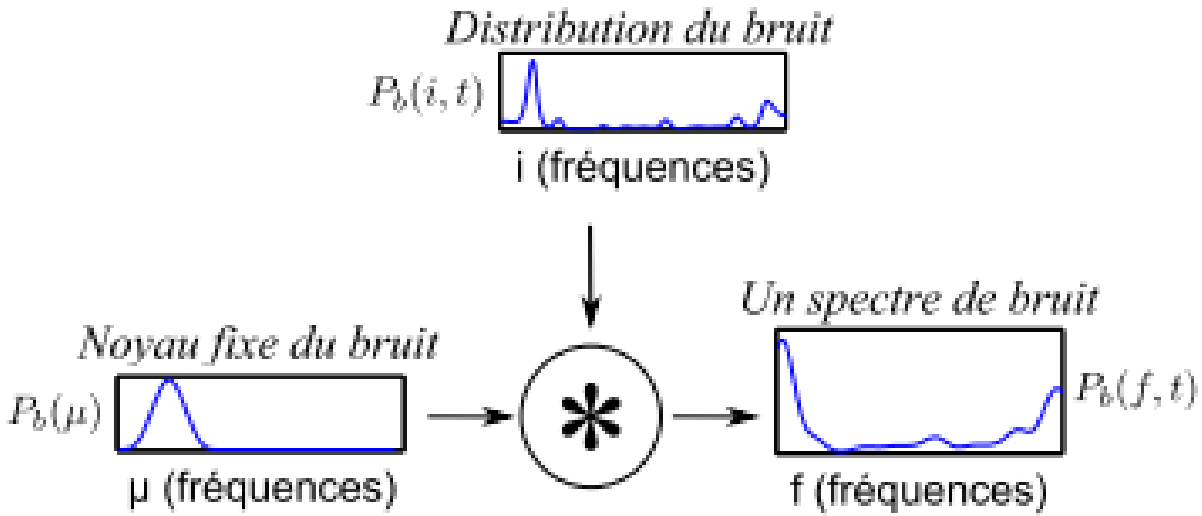


B. Fuentes, R. Badeau, and G. Richard, "Harmonic Adaptive Latent Component Analysis of Audio and Application to Music Transcription" IEEE Trans. On ASLP, 2013.

A PLCA model example

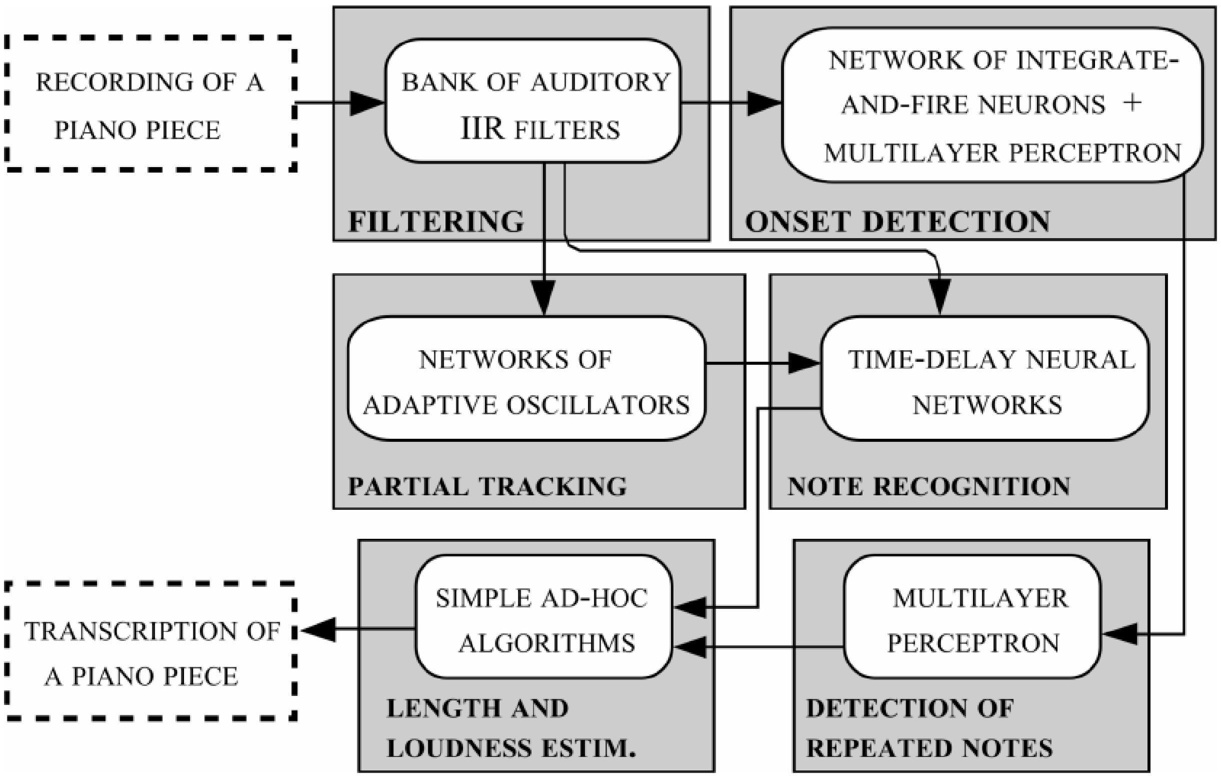
- **The HALCA model (Fuentes & al.)**

$$P(f,t) = P(c=h)P_h(f,t) + P(c=b)\boxed{P_b(f,t)}$$



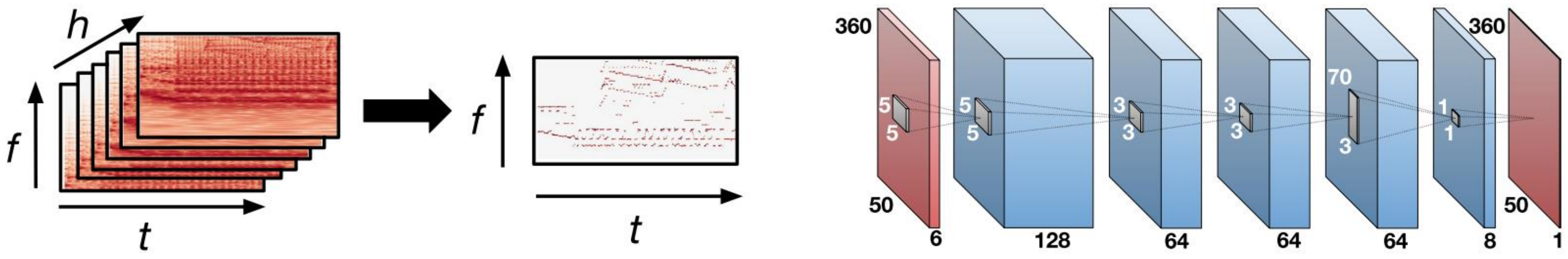
Multipitch estimation using neural networks

- **An early example by M. Marolt (2004) for piano sounds**

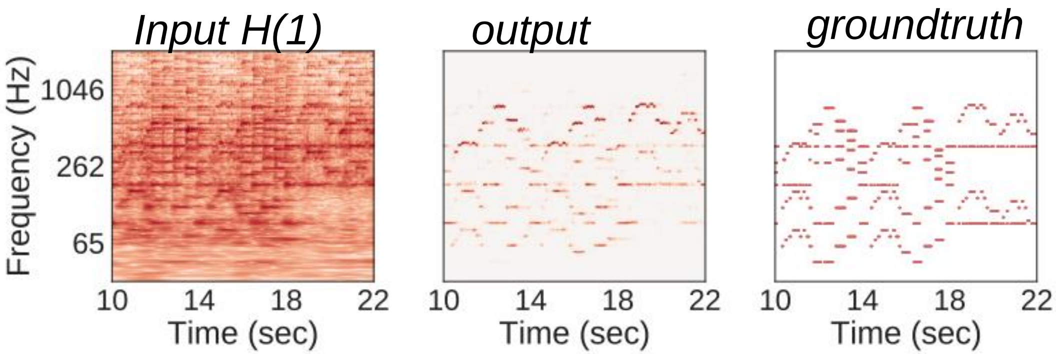


Marolt, Matija. (2004). A Connectionist Approach to Automatic Transcription of Polyphonic Piano Music. Multimedia, IEEE Transactions on. 6. 439 - 449. 10.1109/TMM.2004.827507.

Multipitch estimation using neural networks



- Use of a specific input representation: the harmonic-CQT
- CNN architecture with Relu ; Last layer with sigmoid
- The predicted saliency map can be interpreted as a likelihood score of each time-frequency bin belonging to an f0 contour.



Bittner, Rachel & McFee, Brian & Salamon, Justin & Li, Peter & Bello, Juan. (2017). Deep Salience Representations for f0 Estimation in Polyphonic Music.

Multipitch estimation using neural networks

Other neural approaches

- Deep spiking networks in (Qian 2019)
- Multi-resolution spectrogram as input with LSTM networks (Böck & al. 2012)
- Use of a kind of “language model” in Neural Autoregressive Distribution Estimator, also known as NADE (*similar to wavenet architecture*) in (Sigtia, 2016)
- A succession of 2 bi-LSTM networks (for note onset detection and note duration estimation), in (Hawthorne & al. 2018)

- An interesting reading: (Benetos &al. 2019)

« *Yet, despite these [...] limitations, NMF-based methods remain competitive or even exceed the results achieved using NNs.”*

E. Benetos, S. Dixon, Z. Duan and S. Ewert, "Automatic Music Transcription: An Overview," in *IEEE Signal Processing Magazine*, vol. 36, no. 1, pp. 20-30, Jan. 2019, doi: 10.1109/MSP.2018.2869928.

C. Hawthorne, E. Elsen, J. Song, A. Roberts, I. S. C. Raffel, J. Engel, S. Oore, and D. Eck, “Onsets and frames: Dual-objective piano transcription,” in Proc. Int. Society Music Information Retrieval Conf., 2018, pp. 50–57.

S. Sigtia, E. Benetos, and S. Dixon, “An end-to-end neural network for polyphonic piano music transcription,” IEEE/ACM Trans. Audio, Speech, Language Process., vol. 24, no. 5, pp. 927–939, 2016.

S. Böck and M. Schedl, “Polyphonic piano note transcription with recurrent neural networks,” in Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing, 2012, pp. 121–124.

Qian, Hanxiao et al. “Robust Multipitch Estimation of Piano Sounds Using Deep Spiking Neural Networks.” 2019 IEEE Symposium Series on Computational Intelligence (SSCI) (2019): 2335-2341.