Scale-invariant probabilistic latent component analysis

Romain Hennequin, Bertrand David, Roland Badeau

Scale-invariant probabilistic latent component analysis

Analyse probabiliste en composantes latentes invariante par homothétie

Romain Hennequin
Bertrand David
Roland Badeau

2011D003

Février 2011

Département Traitement du Signal et des Images
Groupe AAO : Audio, Acoustique et Ondes

Scale-invariant probabilistic latent component analysis

Analyse probabiliste en composantes latentes invariante par homothétie

Romain Hennequin, Bertrand David and Roland Badeau
Institut Télécom, Télécom ParisTech; CNRS-LTCI
46 rue Barrault 75683, Paris CEDEX 13
romain.hennequin@telecom-paristech.fr

***Résumé*—Dans cette article, nous présentons une nouvelle méthode de décomposition de spectrogrammes musicaux. Cette méthode vise à transposer les decompositions invariantes par translation qui permettent de décomposer des spectrogrammes à Q-constant (avec une résolution fréquentielle logarithmique) a des spectrogrammes standard issues de transformées de Fourier à court terme (avec une résolution fréquentielle linéaire). Cette technique a l'avantage de permettre facilement une reconstruction des signaux latents par filtrage de Wiener, ce qui peut être utilisé par exemple dans des applications de séparation de sources.**

***Abstract*—In this paper, we present a new method to decompose musical spectrograms. This method transposes shift-invariant probabilistic latent component analysis (PLCA) which permits to decompose constant Q spectrograms (with a logarithmic frequency resolution) to standard short time Fourier transform spectrograms (with a linear frequency resolution). This makes it possible to easily use the method reconstruct the latent signals (which can be useful for source separation).**

***Index Terms*—Non-negative decomposition, Non-negative matrix factorization, probabilistic latent component analysis, shift-invariant decomposition.**

I. Introduction

Non-negative decompositions are widely used for audio spectrograms processing: non-negative matrix factorization (NMF) [5] and PLCA [8], [9] are both used to decompose spectrograms with applications such as source separation [15], [11] and automatic transcription [10], [6]. Shift-invariant decomposition [7], [6], [12] permits to decompose constant-Q spectrograms [1] with a single frequency template for each harmonic instrument: with a log-frequency resolution, a frequency shift corresponds to a transposition. Then each note of a single instrument can be modeled as a base template shifted to the right pitch.

Unfortunately, constant-Q transforms (CQT) are difficult to use for sound source separation: they are difficult to inverse [4], and the variable resolution of the decomposition makes it difficult to apply time-frequency masking. Attempts were made to use shift-invariant decomposition of CQT for source separation [3], [2] using a mapping between log-frequency and linear frequency resolution to avoid CQT inversion. This paper presents a new decomposition method inspired by shift-invariant decompositions but which is designed to directly decompose STFT spectrograms: in a constant-Q spectrogram, a change of fundamental frequency approximately corresponds to a translation of the spectral template. In a standard STFT spectrogram, one can model such a change with an homothety on the spectral template. This approximation is only valid for small transformations since:

- The model of transposition (same set of harmonic amplitudes for all notes) is only valid for a few electronic instruments (this approximation is also used in shift-invariant decomposition).
- Harmonics (or partials) are not Dirac functions in the frequency domain and have a width (given by the size and the type of the analysis window used in the STFT) which is the same for all partials, but an homothety will widen (or slim) this partials.

We call the new decomposition *scale-invariant PLCA*. This scale-invariant model presents some new issues that were not encountered with shift-invariant models: an homothety on a set of integers does not

yield integers. We propose a solution to this issue. In section II, we remind the principle of standard PLCA and shift-invariant PLCA. We then present, in section III, the new scale-invariant model and derive an algorithm to estimate the parameters. Some examples are presented in section IV and we propose an application of single notes repitching in a polyphonic signal. Finally, we draw conclusions in section V.

II. Probabilistic latent component analysis

The model that we used is inspired by shift-invariant PLCA [12] which is a probabilistic drawing model. In PLCA decompositions [9], a non-negative spectrogram $\mathbf{V}_{ft}$ is considered as an histogram obtained from a structured draw of a frequency random variable $f \in \{1, 2, \ldots, F\}$ and a time random variable $t \in \{1, 2, \ldots, T\}$ which follow the joint distribution $P(f, t)$. One can design $P(f, t)$ in different ways, which lead to different decompositions.

A. Standard PLCA

Standard PLCA (non shift-invariant) [9] leads to a decomposition very similar to NMF. The draw is structured with a latent (hidden) random variable z which corresponds to a "component", assuming that f and t are independent conditionally to z:

$$P(f, t|z) = P(f|z)P(t|z),$$

then:

$$P(f, t) = \sum_{z=1}^{Z} P(z)P(f|z)P(t|z).$$

The histogram $\mathbf{V}_{ft}$ is thus assumed to be obtained in the following way: first z is drawn following $P(z)$ and then f and t are drawn following respectively $P(f|z)$ and $P(t|z)$.

In an NMF framework, $P(f|z)$ corresponds to the spectral templates and $P(t|z)$ corresponds to the activations of each component. $P(z)$ is the relative weight of each component and can be computed.

B. Shift-invariant PLCA

Shift-invariant PLCA introduces another latent random variable $\tau \in \mathbb{Z}$ which describes a transposition and another random variable $f' \in \{1, 2, \ldots, F'\}$ which corresponds to the base frequency. f' and t are assumed independent conditionally to z, and τ and f' are also independent conditionally to z (but t and τ are not). f is obtained by a transposition of the base template: $f = f' + \tau$. $P(f, t)$ takes the form:

$$P(f, t) = \sum_{z=1}^{Z} P(z) \sum_{f'=1}^{F'} P_K(f'|z)P_I(f - f', t|z).$$

P_K is called the kernel distribution: it corresponds to the base spectral templates which are shifted by the impulse distribution P_I.

III. Scale-invariant PLCA

A. Model

In standard short-time Fourier transform spectrograms (with a linear frequency resolution), transposition is no longer a shift: we model it with a multiplication by a scalar $\lambda \in \mathbb{R}^+\backslash\{0\}$. Let X be a discrete random variable taking its values in $\{0, 1, 2, ...K\}$ and λ a continuous positive random variable with a density function p. The density of $u = \lambda X$ is then:

$$p_{\lambda X}(u) = \sum_{k=1}^{K} \frac{P(X = k)p(\frac{u}{k})}{k} + \delta(u)P(X = 0) \tag{1}$$

where $\delta(u)$ is a Dirac delta function.

In our model, one assumes that the frequency random variable $f_c \in \mathbb{R}$ is obtained by multiplying the base frequency $f' \in \{0, 1, \ldots, F'\}$ (which is independent of t conditionally to z) with the transposition factor $\lambda \in \mathbb{R}^+\backslash\{0\}$ (which depends on t but not on f' conditionally to z). The random variable f_c is continuous, but the observed random variable $f \in \{0, 1, \ldots, F\}$ is discrete. Then we will suppose that:

$$P(f, t) = \int_{f-\frac{1}{2}}^{f+\frac{1}{2}} P(f_c, t)df_c.$$

Using (1), we get:

$$P(f_c, t|z) = \sum_{f'=1}^{F'} \frac{P_K(f'|z)}{f'} P_I\left(\frac{f_c}{f'}, t|z\right) + \delta(u)P_K(0|z).$$

We use the notation P_K for the kernel distribution and P_I for the impulse distribution, as for shift-invariant PLCA. However they do not represent the same object.

In this paper, we consider that $P_K(0|z) = 0$ to avoid the singularity of the null frequency. As we will see later, there still can be energy in the frequency channel 0 by scaling down the frequency template.

We then get:

$$P(f_c,t)=\sum_{z=1}^{Z}P(z)\sum_{f'=1}^{F'}\frac{P_K(f'|z)}{f'}P_I\left(\frac{f_c}{f'},t|z\right).$$

Consequently:

$$\forall f\in\{0,1,\dots,F\},\forall t\in\{1,\dots,T\}$$
$$P(f,t)=\sum_{z,f'}\frac{P(z)P_K(f'|z)}{f'}\int_{f-\frac{1}{2}}^{f+\frac{1}{2}}P_I\left(\frac{f_c}{f'},t|z\right)df_c$$

The parameters to be estimated are then: $\theta=\{P(z),P_K(f'|z),P_I(\lambda,t|z)\}$.

For practical purposes, one needs to discretize P_I (which is a continuous density function with respect to λ) in some way, in order to estimate θ. We propose to perform this discretization by parameterizing P_I assuming that $\lambda\mapsto P_I(\lambda,t|z)$ is piece-wise constant for all t and z. We select a family $\{\lambda_k\}_{k\in\{1,\dots,K\}}$ (which does not depend on t and z). In this paper, we choose $\lambda_k=2^{\frac{k-k_0}{12n_{\text{st}}}}$: his exponential discretization is chosen to fit a transposition scale in subdivisions of the tone (n_{st} corresponds to the number of discretized values of λ for each semitone). We assume that P_I is given by:

$$\forall\lambda\in[\lambda_k2^{-\frac{1}{24n_{\text{st}}}},\lambda_k2^{\frac{1}{24n_{\text{st}}}}],\quad P_I(\lambda,t|z)=P_I(\lambda_k,t|z).$$

Moreover, we assume that P_I is zero outside these intervals. The values $P_I(\lambda_k,t|z)$ (for all k, t et z) then completely describe P_I.

Then:

$$\int_{f-\frac{1}{2}}^{f+\frac{1}{2}}P_I(\frac{f_c}{f'},t|z)df_c=f'\sum_{k=k_{\text{min}}^{f,f'}}^{k_{\text{max}}^{f,f'}}P_I(\lambda_k,t|z)\delta\lambda_k^{f,f'}$$

where $k_{\text{min}}^{f,f'}$ is chosen so that $\lambda_{k_{\text{min}}^{f,f'}}2^{-\frac{1}{24n_{\text{st}}}}<\frac{f-\frac{1}{2}}{f'}\leq\lambda_{k_{\text{min}}^{f,f'}}2^{\frac{1}{24n_{\text{st}}}}$ and $k_{\text{max}}^{f,f'}$ is chosen so that $\lambda_{k_{\text{max}}^{f,f'}}2^{-\frac{1}{24n_{\text{st}}}}\leq\frac{f+\frac{1}{2}}{f'}<\lambda_{k_{\text{max}}^{f,f'}}2^{\frac{1}{24n_{\text{st}}}}$ (with the following constraints: $1\leq k_{\text{min}}^{f,f'}\leq K$ and $1\leq k_{\text{max}}^{f,f'}\leq K$):

$$\begin{aligned}k_{\text{min}}&=\left\lceil k_0-\frac{1}{2}+12n_{\text{st}}\log_2(\frac{f-\frac{1}{2}}{f'})\right\rceil\\k_{\text{max}}&=\left\lfloor k_0+\frac{1}{2}+12n_{\text{st}}\log_2(\frac{f+\frac{1}{2}}{f'})\right\rfloor\end{aligned}$$

where $\lceil.\rceil$ is the ceiling function and $\lfloor.\rfloor$ the floor function.

Thus $\delta\lambda_k^{f,f'}$ is given by $\delta\lambda_k^{f,f'}=\min(\lambda_k2^{\frac{1}{24n_{\text{st}}}},\frac{f+\frac{1}{2}}{f'})-\max(\lambda_k2^{-\frac{1}{24n_{\text{st}}}},\frac{f-\frac{1}{2}}{f'})$. When $\delta\lambda_k^{f,f'}$ is not limited by constraints on f and f', we will denote $\delta\lambda_k=\lambda_k2^{\frac{1}{24n_{\text{st}}}}-\lambda_k2^{-\frac{1}{24n_{\text{st}}}}$.

The parameters are then: $\theta=\{P(z),P_K(f'|z),P_I(\lambda_k,t|z)|z\in\{1,...,Z\},f'\in\{1,...,F'\},k\in\{1,...,K\},t\in\{1,...,T\}\}$.

Remark: Other parameterizations of P_I are possible (for instance, piecewise affine functions). Then, in order to keep calculation as general as possible, the parametrization of P_I will only appear at the end of the calculus.

B. Expectation-Maximization algorithm

We intend to estimate the value of the parameter θ that maximizes the log-likelihood of observing $\mathbf{V}_{ft}$:

$$L((\bar{f},\bar{t})|\theta)=\sum_{i\in I}\log P(f_i,t_i),\tag{2}$$

where $\bar{f}$ and $\bar{t}$ correspond to the draws of f and t (draws are indexed by $i\in I=\{1,...,N\}$ where N is the total number of draws). As the number of draws that leads to the value (f,t) is V_{ft}, the log-likelihood can be rewritten:

$$L((\bar{f},\bar{t})|\theta)=\sum_{f=1}^{F}\sum_{t=1}^{T}\mathbf{V}_{ft}\log P(f,t)$$

The estimation will be performed with the Expectation-Maximization (EM) algorithm with latent variables z and f' (it would be equivalent to consider z and λ as latent variables since $f=\lambda f'$).

The completed log-likelihood is:

$$\begin{aligned}L((\bar{f},\bar{t},z,f')|\theta)&=\sum_{i\in I}\log P(f_i,t_i,z,f')\\&=\sum_{f=1}^{F}\sum_{t=1}^{T}\mathbf{V}_{ft}\log P(f,t,z,f')\end{aligned}$$

Moreover, since

$$P(f,t,z,f')=\int_{f-\frac{1}{2}}^{f+\frac{1}{2}}P(f_c,t,z,f')df_c,$$

and

$$P(f_c,t,z,f')=P(z)\frac{P_K(f'|z)}{f'}P_I(\frac{f_c}{f'},t|z),$$

then:

$$P(f,t,z,f')=P(z)\frac{P_K(f'|z)}{f'}\int_{f-\frac{1}{2}}^{f+\frac{1}{2}}P_I(\frac{f_c}{f'},t|z)df_c.$$

Consequently:

$$L((\bar{f},\bar{t},z,f')|\theta) = \sum_{f,t}\mathbf{V}_{ft}\Big\{\log P(z) + \log P_K(f'|z) + \log\left(\int_{f-\frac{1}{2}}^{f+\frac{1}{2}} P_I(\frac{f_c}{f'},t|z)df_c\right)\Big\} + c$$

where c is a constant that does not depend on θ.

The completed log-likelihood expectation is then:

$$\begin{aligned}Q(\theta|\theta^{(c)}) &= \mathbb{E}_{z,f'|f,t,\theta^{(c)}}(L((\bar{f},\bar{t},z,f')|\theta)) \\ &= \sum_{f',z,f,t}\mathbf{V}_{ft}P(z,f'|f,t,\theta^{(c)})\Big\{\log P(z) \\ &+ \log P_K(f'|z) + \log\left(\int P_I(\frac{f_c}{f'},t|z)df_c\right)\Big\} + c\end{aligned} \tag{3}$$

where $\theta^{(c)}$ is the current value of the parameter.

We can get an expression for $P(z,f'|f,t,\theta^{(c)})$ with respect to $\theta^{(c)}$ using the Bayes theorem (E step):

$$\begin{aligned}P(z,f'|f,t,\theta^{(c)}) &= \frac{P(f,t,f'|z)P^{(c)}(z)}{P^{(c)}(f,t)} \\ &= \frac{P_K^{(c)}(f'|z)P^{(c)}(z)\int P_I^{(c)}(\frac{f_c}{f'},t|z)df_c}{f'P^{(c)}(f,t)}.\end{aligned} \tag{4}$$

The notation $(.)^{(c)}$ refers to values computed from the current parameter: $\theta^{(c)} = \{P^{(c)}(z), P_K^{(c)}(f'|z), P_I^{(c)}(\lambda,t|z)\}$.

The completed expectation (3) will be maximized (M step) with respect to θ ($\theta^{(c)}$ being fixed). As θ is made up of probabilities that must sum to 1, the maximization is constrained. Thus we use Lagrange multipliers with the Lagrangian:

$$\begin{aligned}H(\theta|\theta^{(c)}) = &\sum_{f,z,f',t}\mathbf{V}_{ft}P(z,f'|f,t,\theta^{(c)})\Big\{\log P(z) \\ &+ \log P_K(f'|z) + \log\left(\int P_I(\frac{f_c}{f'},t|z)df_c\right)\Big\} \\ &+\mu\left(\sum_z P(z) - 1\right) \\ &+\sum_z \rho_z\left(\sum_{f'} P_K(f'|z) - 1\right) \\ &+\sum_z \tau_z\left(\sum_t\int_{\lambda_{\min}}^{\lambda_{\max}} P_I(\lambda,t|z) - 1\right)\end{aligned}$$

where μ, ρ_z and τ_z are Lagrange multipliers.

1) Update of $P(z)$: $\frac{\partial H(\theta|\theta^{(c)})}{\partial P(z)} = 0$ leads to the update rule of $P(z)$ (details of the calculation are given in appendix A):

$$P(z) \leftarrow \frac{\sum_{f,f',t}\mathbf{V}_{ft}P(z,f'|f,t,\theta^{(c)})}{\sum_{z,f,f',t}\mathbf{V}_{ft}P(z,f'|f,t,\theta^{(c)})} \tag{5}$$

2) Update of $P_K(f'|z)$: In a similar way, we obtain the update rule of $P_K(f'|z)$:

$$P_K(f'|z) \leftarrow \frac{\sum_{f,t}\mathbf{V}_{ft}P(z,f'|f,t,\theta^{(c)})}{\sum_{f,f',t}\mathbf{V}_{ft}P(z,f'|f,t,\theta^{(c)})} \tag{6}$$

3) Update of $P_I(\lambda_k,t|z)$: Because of the expression of the Lagrangian $H(\theta|\theta^{(c)})$ with respect to $P_I(\lambda_k,t|z)$, the update rule of $P_I(\lambda_k,t|z)$ is more complex to derive.

We consider the following "fixed-point" update rule (iterated several times) which hopefully will converge to a zero of $\frac{\partial H}{\partial P_I(\lambda_k,t|z)}$ (see appendix B):

$$P_I(\lambda_k,t|z) \leftarrow \sum_{f,f'}\frac{\mathbf{V}_{ft}P(z,f'|f,t,\theta^{(c)})P_I(\lambda_k,t|z)\mathbb{1}_{[k_{\min},k_{\max}]}(k)}{-\tau_z\delta\lambda_k\sum_{k'=k_{\min}^{f,f'}}^{k_{\max}^{f,f'}}P_I(\lambda_{k'},t|z)\delta\lambda_{k'}^{f,f'}}\delta\lambda_k^{f,f'} \tag{7}$$

In equation (7), the division by τ_z is a normalization.

We did not manage to prove convergence of P_I under several iterations of the update rule (7) to a zero of $\frac{\partial H}{\partial P_I}$. However we observed it in practice. As the constrained maximization problem considered is stricly concave with affine equality constraints, the obtained fixed point which verifies the Karush-Kuhn-Tucker conditions is necessarily the global maximum of $Q(\theta|\theta^{(c)})$ (defined in equation (3)) under the normalization constraints. Thus, $Q(\theta|\theta^{(c)})$ is effectively maximized at each iteration and the EM algorithm converges to a local minimum of the likelihood.

C. Multiplicatives updates

Update rules (5) and (6) can be rewritten in a multiplicative form, using expression (4), replacing the calculus of the denominator of each rule by a normalization:

1) Update of $P(z)$:

$$\begin{cases} P^{(i)}(z) & \leftarrow & P^{(c)}(z) \sum_{f,t,f'} \frac{\mathbf{V}_{ft}}{P^{(c)}(f,t)} P_K^{(c)}(f'|z) \\ & & \sum_{k'=k_{\min}^{f,f'}}^{k_{\max}^{f,f'}} P_I(\lambda_{k'},t|z)\delta\lambda_{k'}^{f,f'} \\ P(z) & \leftarrow & \frac{P^{(i)}(z)}{\sum_{z'} P^{(i)}(z')} \quad \text{(normalization)} \end{cases}$$

2) Update of $P_K(f'|z)$:

$$\begin{cases} P_K^{(i)}(f'|z) & \leftarrow & P_K^{(c)}(f'|z) \sum_{f,t} \frac{\mathbf{V}_{ft}}{P^{(c)}(f,t)} P^{(c)}(z) \\ & & \sum_{k'=k_{\min}^{f,f'}}^{k_{\max}^{f,f'}} P_I(\lambda_{k'},t|z)\delta\lambda_{k'}^{f,f'} \\ P_K(f'|z) & \leftarrow & \frac{P_K^{(i)}(f'|z)}{\sum_{f''} P_K^{(i)}(f''|z)} \quad \text{(norm.)} \end{cases}$$

The update rule (7) of P_I can also be rewritten in a multiplicative form:

$$\begin{cases} P_I^{(i)}(\lambda_k,t|z) \leftarrow P_I(\lambda_k,t|z) P^{(c)}(z) \sum_{f'} P_K^{(c)}(f'|z) \\ \quad \sum_f \frac{\mathbf{V}_{ft} \int P_I^{(c)}(\lambda,t|z) d\lambda \delta\lambda_k^{f,f'}}{P^{(c)}(f,t) \int P_I(\lambda,t|z) d\lambda} \mathbb{1}_{[k_{\min}^{f,f'}, k_{\max}^{f,f'}]}(k) \\ P_I(\lambda_k,t|z) \leftarrow \frac{P_I^{(i)}(\lambda_k,t|z)}{\delta\lambda_k \sum_{k',t'} P_I^{(i)}(\lambda_{k'},t'|z)} \quad \text{(norm.)} \end{cases}$$

with :

$$\int P_I^{(c)}(\lambda,t|z) d\lambda = \sum_{k'} P_I^{(c)}(\lambda_{k'},t|z)\delta\lambda_{k'}^{f,f'}$$

$$\text{and } \int P_I(\lambda,t|z) d\lambda = \sum_{k'} P_I(\lambda_{k'},t|z)\delta\lambda_{k'}^{f,f'}$$

.

D. Computational complexity of the algorithm

The main drawback of our algorithm is its important computational complexity: in opposition to shift-invariant PLCA, the computation of the parametric spectrogram $P(f,t)$ can not be done with a fast convolution algorithm. Then the computation time to get the decomposition is quite important, and the computation time is about a hundred to a thousand (depending on the parameters size) times longer than real-time on a recent standard personal computer.

IV. Examples and applications

A. Toy example

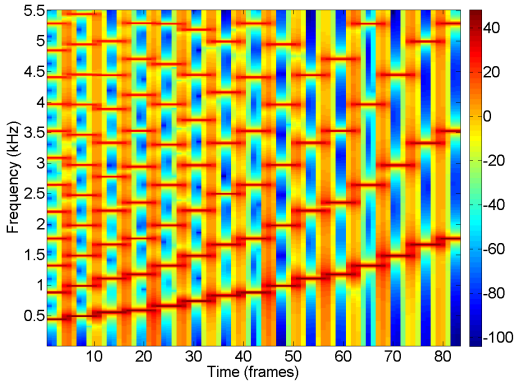
In this section, we present the decomposition provided by our algorithm of a simple spectrogram. The decomposed spectrogram is obtained by STFT (using a 1024 sample-long Hann window with 75% overlap) of a a short excerpt of synthesizer which plays the notes of a A major scale on two octave (From A4 to A6) sampled at $11025Hz$. The original spectrogram is pictured in figure 1(a). The decomposition provides the reconstructed spectrogram pictured in figure fig:reconstructedspectrogram: the reconstructed spectrogram is very similar to the original one. The difference of maximum amplitudes between original and reconstructed spectrograms come from the normalization of $P(f,t)$ (V_{ft} is not normalized), but the dynamic remains the same in both spectrogram. High frequency harmonics of the reconstructed spectrogram are slightly larger than the original one.

Obtained kernel distribution P_K is represented in figure 2(b): we can see that the factorized template is actually harmonic. For high values of the frequency index, amplitudes of the templates tend to be very small. This comes from the fact that our model consider that values of the spectrogram outside the observed frequencies are zeros whereas the model spectrogram $P(f,t)$ can take positive values outside this range. This can be solved using the approach of [14], [13]. In practice, for signals of actual acoustic instruments this not a real issue, since the harmonic content of such signal is almost entirely smothered by quantification noise from $5000Hz$: thus a sampling rate of $22050Hz$ or more permits to reduce this issue.

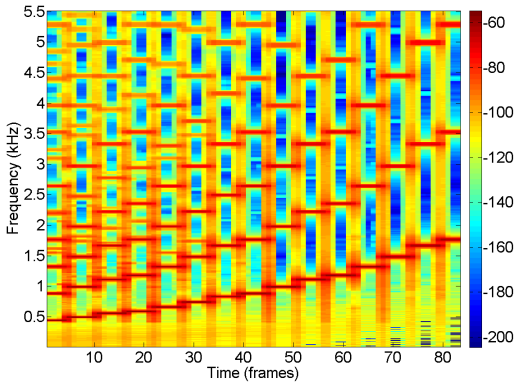
Impulse distribution P_I is represented in figure 2(a): high probabilities clearly appear at actual notes relative positions. There are also some replicas of the notes at position with high harmonic similitudes (octave, twelfth, double octave...). At onset time, P_I takes high values for many values of the homothety factor λ: this comes from the flat shape of the spectrum of onset which is matched her with several rescaled harmonic templates.

B. Real audio data

In this section, we present the SIPLCA decomposition of the spectrogram of the 10 first second of the song Because by the Beatles with a single template ($Z = 1$). The decomposed signal consists in a polyphonic harpsichord introduction recorded in real condition. The original signal was transformed in a

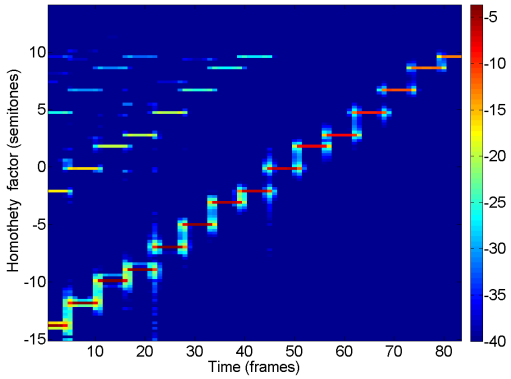


(a) Original spectrogram

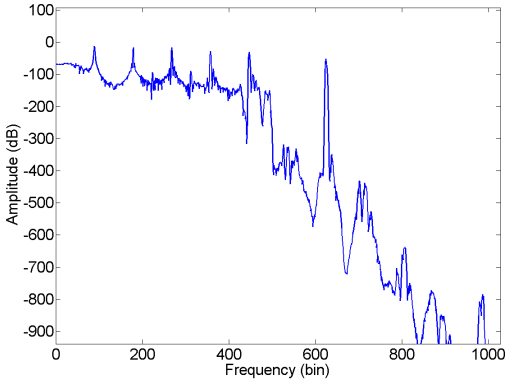


(b) Reconstructed spectrogram

Figure 1. Original and reconstructed spectrogram.



(a) Impulse distribution P_I



(b) Kernel distribution P_K

Figure 2. Scale-invariant PLCA: Kernel and impulse.

mono signal (summing both channels) and downsampled to $22050Hz$. The spectrogram was calculated with a STFT using a 2048 sample-long Hann window with 75% overlap.

Obtained impulse distribution P_I is represented in figure IV-B: actual played notes are materialized by a rectangle in the figure. We can see that in all rectangles, P_I takes high values. P_I

The impulse distribution P_I is thus very similar to the impulse distribution that can be obtained with shift-invariant decompositions. However, as our decomposition is done on linear frequency resolution spectrograms, it has an important advantage: it is possible to generate time-frequency mask that can be directly used to separate different components with Wiener filtering. Thus it is possible to isolate single notes in a polyphonic signal and to repitch them individually.

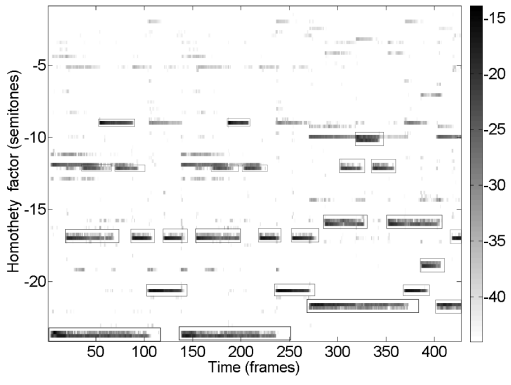


Figure 3. Impulse distribution of the introduction of the song Because.

V. Conclusion

In this paper, we proposed a new way to decompose non-negative music spectrogram: the decomposition is based on a few frequency templates which can be rescaled at each frame (which corresponds to a transposition). We presented examples of this decomposition on music spectrogram and showed how this decomposition can be used to modify individual notes in a polyphonic signal.

Future works should deal with a better representation of non-harmonic components of musical sounds, such as transients.

References

[1] Judith C. Brown. Calculation of a constant Q spectral transform. *Journal of the Acoustical Society of America*, 89(1):425–434, January 1991.

[2] Derry Fitzgerald and Matt Cranitch. Resynthesis methods for sound source separation using shifted non-negative factorisation models. In *Irish Signals and Systems Conference*, Derry, Ireland, 2007.

[3] Derry Fitzgerald, Matt Cranitch, and Eugene Coyle. Shifted non-negative matrix factorisation for sound source separation. In *IEEE conference on Statistics in Signal Processing*, pages 1132 – 1137, Bordeaux, France, July 2005.

[4] Derry Fitzgerald, Matt Cranitch, and Marcin T. Cychowsky. Towards an inverse constant Q transform. In *Audio Engineering Society Convention Paper*, Paris, France, May 2006.

[5] Daniel D. Lee and H. Sebastian Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401(6755):788–791, October 1999.

[6] Gautham J. Mysore and Paris Smaragdis. Relative pitch estimation of multiple instruments. In *International Conference on Acoustics, Speech and Signal Processing*, pages 313–316, Taipei, Taiwan, April 2009.

[7] M. Schmidt and M. Mørup. Nonnegative matrix factor 2-D deconvolution for blind single channel source separation. In *Conference on Independent Component Analysis and Blind Source Separation (ICA)*, volume 3889 of *Lecture Notes in Computer Science (LNCS)*, pages 700–707, Paris, France, April 2006. Springer.

[8] Madhusudana V. Shashanka. *Latent variable framework for modeling and separating single-channel acoustic sources*. PhD thesis, Boston University, Boston, MA, USA, 2007.

[9] M.V. Shashanka, B. Raj, and P. Smaragdis. Sparse overcomplete latent variable decomposition of counts data. In *Neural Information Processing Systems*, Vancouver, BC, Canada, December 2007.

[10] Paris Smaragdis and Judith C. Brown. Non-negative matrix factorization for polyphonic music transcription. In *Workshop on Applications of Signal Processing to Audio and Acoustics*, pages 177 – 180, New Paltz, NY, USA, October 2003.

[11] Paris Smaragdis, Bhiksha Raj, and Madhusudana Shashanka. Supervised and semi-supervised separation of sounds from single-channel mixtures. In *7th International Conference on Independent Component Analysis and Signal Separation*, London, UK, September 2007.

[12] Paris Smaragdis, Bhiksha Raj, and Madhusudana Shashanka. Sparse and shift-invariant feature extraction from non-negative data. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2069 – 2072, Las Vegas, Nevada, USA, March 2008.

[13] Paris Smaragdis, Bhiksha Raj, and Madhusudana Shashanka. Missing data imputation for spectral audio signals. In *IEEE international workshop on Machine Learning for Signal Processing (MLSP)*, Grenoble, France, September 2009.

[14] Paris Smaragdis, Bhiksha Raj, and Madhusudana Shashanka. Missing data imputation for time-frequency representations of audio signals. *Journal of Signal Processing Systems*, August 2010.

[15] Tuomas Virtanen. Monaural sound source separation by non-negative matrix factorization with temporal continuity. *IEEE Transactions on Audio, Speech and Language Processing*, 15(3):1066–1074, March 2007.

Appendix A
Update rule of $P(z)$ (calculation)

The partial derivative of $H(\theta|\theta^{(c)})$ with respect to $P(z)$ is:

$$\frac{\partial H(\theta|\theta^{(c)})}{\partial P(z)} = \sum_{f,f',t} \mathbf{V}_{ft} \frac{P(z,f'|f,t,\theta^{(c)})}{P(z)} + \mu$$

This partial derivative must be zero: $\frac{\partial H(\theta|\theta^{(c)})}{\partial P(z)} = 0$, then:

$$\sum_{f,f',t} \mathbf{V}_{ft} P(z,f'|f,t,\theta^{(c)}) + \mu P(z) = 0 \quad (8)$$

A summation on z leads to:

$$\mu = -\sum_{z,f,f',t} \mathbf{V}_{ft} P(z,f'|f,t,\theta^{(c)})$$

Then, from equation (8), we get the update rule of $P(z)$:

$$P(z) \leftarrow \frac{\sum_{f,f',t} \mathbf{V}_{ft} P(z,f'|f,t,\theta^{(c)})}{\sum_{z,f,f',t} \mathbf{V}_{ft} P(z,f'|f,t,\theta^{(c)})} \quad (9)$$

Appendix B
Update rule of $PI(\lambda_k,t|z)$ (calculation)

The partial derivative of the Lagrangian with respect to $P_I(\lambda_k,t|z)$ is:

$$\frac{\partial H(\theta|\theta^{(c)})}{\partial P_I(\lambda_k,t|z)} = \sum_{f,f'} \mathbf{V}_{ft} P(z,f'|f,t,\theta^{(c)})$$

$$\frac{\partial \log\left(\int P_I(\frac{f_c}{f'},t|z)df_c\right)}{\partial P_I(\lambda_k,t|z)} + \tau_z \frac{\partial \int_{\lambda_{\min}}^{\lambda_{\max}} P_I(\lambda,t|z)}{\partial P_I(\lambda_k,t|z)}$$

$$= \sum_{f,f'} \mathbf{V}_{ft} \frac{P(z,f'|f,t,\theta^{(c)})}{\int P_I(\frac{f_c}{f'},t|z)df_c}$$

$$\frac{\partial\left(\int P_I(\frac{f_c}{f'},t|z)df_c\right)}{\partial P_I(\lambda_k,t|z)} + \tau_z \frac{\partial \int_{\lambda_{\min}}^{\lambda_{\max}} P_I(\lambda,t|z)}{\partial P_I(\lambda_k,t|z)}$$

Using the proposed parametrization of P_I (piecewise constant function):

$$\frac{\partial H}{\partial P_I} = \sum_{f,f'} \mathbf{V}_{ft} \frac{P(z,f'|f,t,\theta^{(c)}) \mathbb{1}_{[k_{\min}^{ff'}, k_{\max}^{ff'}]}(k)}{\sum_{k'=k_{\min}^{f,f'}}^{k_{\max}^{f,f'}} P_I(\lambda_{k'},t|z)\delta\lambda_{k'}^{f,f'}} \delta\lambda_k^{ff'} + \tau_z \delta\lambda_k$$

where $\mathbb{1}_S$ denotes the indicator function of set S.

Then, we must have:

$$\sum_{f,f'} \mathbf{V}_{ft} \frac{P(z,f'|f,t,\theta^{(c)}) \mathbb{1}_{[k_{\min}^{f,f'}, k_{\max}^{f,f'}]}(k)}{\sum_{k'=k_{\min}^{f,f'}}^{k_{\max}^{f,f'}} P_I(\lambda_{k'},t|z)\delta\lambda_{k'}^{f,f'}} \delta\lambda_k^{f,f'} + \tau_z \delta\lambda_k = 0$$

which is equivalent to:

$$\sum_{f,f'} \mathbf{V}_{ft} P(z,f'|f,t,\theta^{(c)}) \frac{P_I(\lambda_k,t|z) \mathbb{1}_{[k_{\min}^{f,f'}, k_{\max}^{f,f'}]}(k)}{\sum_{k'=k_{\min}^{f,f'}}^{k_{\max}^{f,f'}} P_I(\lambda_{k'},t|z)\delta\lambda_{k'}^{f,f'}} \delta\lambda_k^{f,f'} + \tau_z \delta\lambda_k P_I(\lambda_k,t|z) = 0$$

A summation on k and t leads to:

$$\tau_z = -\sum_{k,t,f,f'} \mathbf{V}_{ft} P(z,f'|f,t,\theta^{(c)}) \frac{P_I(\lambda_k,t|z)\delta\lambda_k^{f,f'} \mathbb{1}_{[k_{\min}^{f,f'}, k_{\max}^{f,f'}]}(k)}{\sum_{k'=k_{\min}^{f,f'}}^{k_{\max}^{f,f'}} P_I(\lambda_{k'},t|z)\delta\lambda_{k'}^{f,f'}}$$

We thus get the fixed-point update rule (7).

Dépôt légal : 2011 – 1er trimestre
Imprimé à Télécom ParisTech – Paris
ISSN 0751-1345 ENST D (Paris) (France 1983-9999)

Télécom ParisTech

Institut TELECOM - membre de ParisTech

46, rue Barrault - 75634 Paris Cedex 13 - Tél. + 33 (0)1 45 81 77 77 - www.telecom-paristech.frfr

Département TSI