Time-Frequency Jigsaw Puzzle: adaptive multiwindow and multilayered Gabor expansions

Florent Jaillet, Bruno Torrésani

Time-Frequency Jigsaw Puzzle: Adaptive multiwindow and multilayered Gabor expansions

Abstract

We describe a new adaptive multiwindow Gabor expansion, which dynamically adapts the windows to the signal's features in time-frequency space. The adaptation is based upon local time-frequency sparsity criteria, and also yields as by-product an expansion of the signal into layers corresponding to different windows. As an illustration, we show that simply using two different windows with different sizes leads to decompositions of audio signals into transient and tonal layers. We also discuss potential applications to transient detection and denoising.

Index Terms

Gabor expansion, multiwindow, time-frequency concentration, adaptivity, parsimony, entropy.

I. INTRODUCTION

Time-frequency representations[1], [2], [8] provide simple and efficient representations of signals. Among time-frequency representation methods, the so-called "atomic" time-frequency representations, based upon signal expansions on a family of elementary waveforms (the atoms), generated using simple rules, have enjoyed increasing success, mainly because of their versatility and simplicity.

Florent Jaillet and Bruno Torrésani are with Laboratoire d'Analyse, Topologie et Probabilités, Université de Provence, 39 rue F. Joliot-Curie, 13453 Marseille cedex 13, France. E-mail: kowalski@cmi.univ-mrs.fr, torresan@cmi.univ-mrs.fr

Florent Jaillet is with GENESIS SA, Bâtiment Gérard MÉGIE, domaine du petit Arbois, BP 69, 13545 Aix en Provence Cedex 4, France.

Nevertheless, in some situations, such approaches turn out to suffer from an important lack of flexibility. For example, it is well known that the time-frequency representation provided by short time Fourier transformation depends strongly on the choice of the analyzing window. Indeed, different choices for the analyzing window emphasize different features of the signal. Such a property may be exploited in some cases (for example, when one can adapt the window to some specific feature of the signal to be analyzed). It may also become a drawback in more complex situations, for example when the analyzed signal contains significantly different features. In such cases, there does not exist any choice of the analyzing window that would be satisfactory for all of them.

The goal of this paper is to present a new approach for this problem, based on an automatic selection of the optimal window, *locally in the time-frequency plane*. The proposed approach is based on a paving of the time-frequency plane into rectangular "super-tiles", within which the sparsest Gabor representation (within a fixed family of Gabor transformations, using different windows) of the signal is seeked. The so-obtained multiple Gabor transformation is then inverted, yielding a first approximation of the signal, involving what we shall call sparse *layers*. This first approximation is then substracted from the signal, yielding a residual, to which the same procedure is applied. The procedure is then iterated until the residual is small enough, or other criterion is met.

Our approach presents some similarities with the family of *greedy algorithms*, such as matching pursuit and orthogonal matching pursuit in time-frequency dictionaries[13], [9], and shares some aspects of the Multiple Window Gabor expansions[5], [18]. However, it differs from the former by the criterion used to select the time-frequency atoms (a sparsity criterion rather than a matching criterion), and by the use of dictionaries (generally, an union of two Gabor frames is used) that are smaller than those generally used in matching pursuit type algorithms. It also differs from the latter by the fact that the problem is not set up as that of *reducing* a very redundant frame. The primary target application of the proposed approach is improved readability *time-frequency signal representation*. However, the so-obtained time-frequency representations turn out to be quite suitable for several post-processing tasks, as we shall show at the end of this article.

This paper is organized as follows. After this introduction, we first recall in Section II the main aspects of time frequency representations and Gabor and multiple Gabor frame theory that will be of interest to us, and introduce the sparsity measures we shall be using. We describe our approach in Section III, and numerical illustrations are provided in Section IV. Pseudocode for the proposed algorithms may be found in the appendix.

II. TIME-FREQUENCY REPRESENTATIONS, TIME-FREQUENCY LOCALIZATION, SPARSITY

A. *Time-frequency representations, Gabor systems*

Let us start with a short account of atomic time-frequency representations. We shall focus here on the case of the short-time Fourier transform (also termed continuous Gabor transform[1]). For the sake of simplicity, we shall limit our discussion to the case of finite energy signals, i.e. square integrable functions[1]. The case of discrete signals may be handled in a similar way.

Let $g \in L^2(\mathbb{R})$, $g \neq 0$, and $b, \nu \in \mathbb{R}_*^+$, and consider the set of shifted and modulated copies g_{mn} of g, defined by

$$g_{mn} = e^{2i\pi n\nu t} g(t - mb) , \quad m, n \in \mathbb{Z} . \tag{1}$$

These may be seen as resulting of a sampling of the continuous family of translates and modulates $t \to g_{(\tau,f)}(t) = e^{2i\pi ft} g(t - \tau)$, $f, \tau \in \mathbb{R}$ on the lattice $\mathcal{L} = b\mathbb{Z} \times \nu\mathbb{Z}$ in the time-frequency plane. It may be shown[3] that for $b\nu$ small enough, the family of time-frequency atoms $\{g_{mn}, m, n \in \mathbb{Z}\}$ constitute a *frame*[7] in $L^2(\mathbb{R})$: this in particular implies that to any signal $x \in L^2(\mathbb{R})$ can be associated the sequence $G_x \in \ell^2(\mathbb{Z}^2)$ of Gabor coefficients

$$G_x(m, n) = \langle x, g_{mn} \rangle . \tag{2}$$

Under these assumptions, it may be shown[3] that the sequence of coefficients G_x is (doubly) square-summable

$$\|G_x\|^2 := \sum_{m,n} |G_x(m, n)|^2 < \infty , $$

and that x may be reconstructed from the latter: there exists a *dual window* $\tilde{g}$ and a corresponding *dual* family of Gabor atoms $\tilde{g}_{mn}$, such that for all $x \in L^2(\mathbb{R})$,

$$x = \sum_{m,n} G_x(m, n) \tilde{g}_{mn} . \tag{3}$$

In fact, such a frame inversion is generally far from unique, and (unless the frame is a basis of $L^2(\mathbb{R})$), there are infinitely many expansions such as (3) for expressing x from its Gabor coefficients $G_x(m, n)$.

The sequence of Gabor coefficients also yields a time-frequency representation, by considering its square modulus, suitably normalized

$$\rho_x(m, n) = \frac{1}{\|G_x\|^2} |G_x(m, n)|^2 , \tag{4}$$

---

[1]Recall that the space of square-integrable functions $L^2(\mathbb{R}) = \{x : \mathbb{R} \to \mathbb{C}, \int_{-\infty}^{\infty} |x(t)|^2 \, dt < \infty\}$ is an inner product space, with inner product $\langle x, y \rangle = \int_{-\infty}^{\infty} x(t)\overline{y}(t) \, dt$ and norm $\|x\| = \sqrt{\langle x, x \rangle}$.

so that $\sum_{m,n} \rho_x(m,n) = 1$, i.e. ρ_x defines a discrete probability distribution. The latter (a discrete *spectrogram*) is generally used for signal *analysis*, as an alternative representation, from which some salient features of the signal are more easily accessible[1], [2], [8][2].

B. Multiple Gabor systems

Multiple Gabor systems have been proposed[18] for relaxing the dependance of the time-frequency representation on the analyzing window. The main idea is the following: given a family of R Gabor frames

$$\mathcal{F}^r = \{g_\lambda^r, \lambda \in \mathcal{L}^r\}, \quad r = 1, \ldots R,$$

with different window functions $g^1, \ldots g^R$ and associated time-frequency lattices $\mathcal{L}^1, \ldots \mathcal{L}^R$, denote by

$$G_x^r(m,n) = \langle x, g_{mn}^r \rangle \tag{5}$$

the corresponding Gabor coefficients of $x \in L^2(\mathbb{R})$. It is easy to show that the union of these Gabor frames is again a frame in $L^2(\mathbb{R})$. Hence, it may be proved that any $x \in L^2(\mathbb{R})$ may be written as

$$x = \sum_{r=1}^R \sum_{m,n} \Gamma_x^r(m,n)\tilde{g}_{mn}^r \tag{6}$$

for some set of coefficients $\Gamma_x^r(m,n)$, the functions $\tilde{g}_{mn}^r$ being the dual frame of g_{mn}^r (for example, take $\Gamma_x^r(m,n) = \langle x, g_{mn}^r \rangle / R$). Again, there are infinitely many such expansions of x in terms of the dual Gabor atoms $\tilde{g}_{mn}^r$, and the "optimal" ones may be sought.

Recently, the notion of *quilted Gabor frame*[5] has been proposed as a solution to this problem. Starting from a union of several frames, the idea is to reduce the so-obtained highly redundant frame by limiting the expansion to suitably chosen time-frequency atoms in different regions of the time-frequency domain. These selected atoms have to be chosen once for all, and must form a frame of the signal space. Therefore, such an approach is not intrinsically adaptive (although one may think of first selecting a data driven quilted frame, before expanding a signal with respect to it). We describe in Section III a fully adaptive approach, based on sparsity considerations, which we introduce below.

C. Adaptive representation, sparsity

It is natural to ask the question: does there exist an "optimal" representation for the signal. Of course, the answer depends on what is meant by "optimal". Following Wickerhauser and coworkers[17], we

---

[2]and references therein.

define the optimality in terms of *parsimony*, or *sparsity*: a time-frequency representation is "good" if it represents the signal in an economical way, i.e. by concentrating the information on a limited number of coefficients.

In the framework of Gabor expansion of a fixed signal $x \in L^2(\mathbb{R})$, the optimal window is defined as the window $g \in L^2(\mathbb{R})$ which maximizes a given sparsity measure S:

$$g_{opt} = \arg \max_g S(g) .$$

Classical choices for the sparsity measure exploit the family of Rényi entropies R_α, $0 < \alpha < 1$, defined as follows. Given a discrete probability distribution u, i.e. a vector with positive coordinates that sum up to unity, we introduce the Rényi entropy[16]

$$R_\alpha(u) = \frac{1}{1-\alpha} \log_2 \left[\sum_n u_n^\alpha \right] . \tag{7}$$

Note that R_α is nothing but a logarithmic form of some ℓ^α norm of the vector. Different values of α yield different ways of measuring sparsity. We shall not consider such issues here[11], as the choice of α does not appear crucial in the proposed method. In the limit $\alpha \to 1$, one obtains the Shannon entropy

$$R_1(u) = - \sum_n u_n \log_2 (u_n) . \tag{8}$$

It is very easy to see that the *negentropy* (negative entropy) $-R_\alpha$ is indeed a measure of sparsity, as the entropy R_α is maximal for constant vectors (up to a phase factor), and minimal for "Krönecker-like" vectors $u_n = C\delta_{n,n_0}$.

Such criteria may be used to search the "best" window for a given signal: Suppose one is given a (parametric) family of window functions $g^1, g^2, \ldots g^R$, together with time-frequency lattices $\mathcal{L}^1, \mathcal{L}^2, \ldots \mathcal{L}^R$ in the time-frequency plane, such that the corresponding families of atoms $\{g_{mn}^r, (m,n) \in \mathcal{L}^r\}$ are frames in $L^2(\mathbb{R})$, compute the corresponding Gabor transforms and associated time-frequency representations, denoted by ρ_x^r, which is by construction a probability distribution. Then the criterion to optimize takes the form

$$S(g^r) = -R_\alpha(\rho_x^r) ,$$

for some fixed $0 < \alpha \leq 1$, and the optimal window for a given signal x is the one that maximizes the criterion.

Unfortunately, it was shown [11] that although such an approach gives satisfactory results for simple synthetic signals, it generally fails as soon as the signal under consideration becomes more complex, i.e. contains features of significantly different nature. Nevertheless, it is possible to replace the above
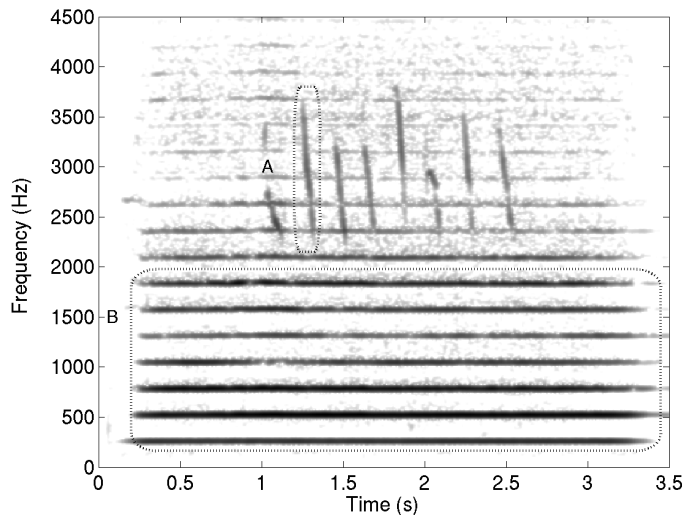
Fig. 1. Spectrogram of the synthetic mixture "flute + bird" signal, with a medium size window, and the two regions A and B on which the window adaptation is performed.

optimization with a more "local" one [11], [12], by optimizing the window with respect to specific features of the signal rather than doing so globally. This may be done thanks to suitable user interface, allowing the user to select a region of interest Ω in the time-frequency plane (from an image of the time-frequency representation), and performing the optimization on the signal obtained by partial reconstruction from this region:

$$x_{loc} = \sum_{(m,n)\in\Omega} G_x(m,n)\tilde{g}_{mn} \ .$$

In such a way, one obtains time-frequency representations that are well adapted for representing specific features of a signal. An illustration may be found in Fig. 1, which represents the spectrogram (with a medium width window) of a synthetic mixture of flute recording and a bird song. Optimizing the window size in regions A and B shown in Fig. 1 yields respectively narrow and wide window sizes. The corresponding spectrograms are shown in Figs. 2 and 3 respectively. Such local adaptations may be efficiently used as pre-processing, prior to any signal analysis/detection/parameter estimation tasks[10].

The approach descrived above is a supervized one, in the sense that it requires the selection (by the user) of a region Ω in the time-frequency plane. In the next section, we describe an unsupervised approach for dynamically adapting the window size, exploiting the ideas we just outlined.
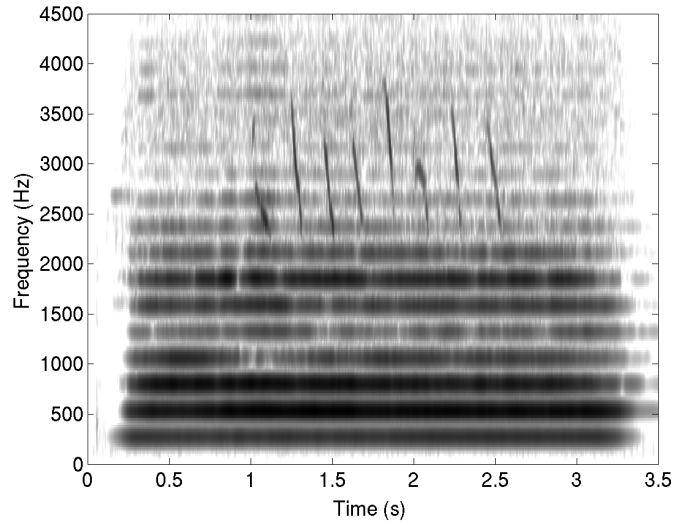
Fig. 2.   Spectrogram of the synthetic mixture "flute + bird" signal, with the window optimized within region A in Fig. 1
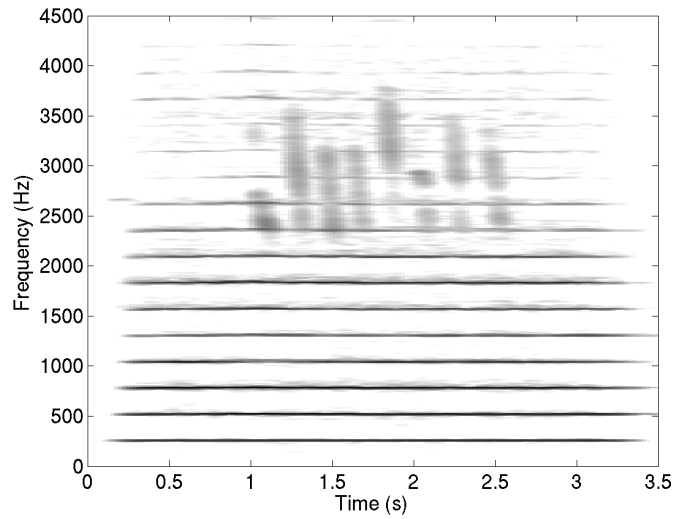

Fig. 3.   Spectrogram of the synthetic mixture "flute + bird" signal, with the window optimized within region B in Fig. 1

III. TIME-FREQUENCY JIGSAW PUZZLES

The starting point of our approach is to search for an optimal way of representing signals using Gabor atoms, with different time-frequency positions and sizes (and possibly other descriptors). Identifying these atoms with rectangular tiles with different locations and sizes in the time-frequency plane, this problem

intuitively amounts to finding good tilings of the time-frequency plane with those rectangular tiles, i.e. solving a kind of jigsaw puzzle problem. Hence the name of the proposed method.

We first describe the simplest instance of the method we propose, the version 1 of the Time Frequency Jigsaw Puzzle algorithm (TFJP1 for short), before examining "by-products" and variants. For the sake of simplicity, we limit the discussion to the case of two windows only keeping in mind the particular case of two windows of the same shape, considered at different scales (a narrow and a wide window), which we used in the numerical examples. The extension to more than two windows is straightforward, and has also been implemented numerically. We shall briefly comment on it in the conclusion section.

A. The TFJP1 method

Let us start from Gabor transforms as before, and consider several windows g^1, g^2, time-frequency lattices $\mathcal{L}^1, \mathcal{L}^2$, and corresponding canonical dual windows $\tilde{g}^1, \tilde{g}^2$, we fix $A, B \in \mathbb{R}^+$ large enough, and consider a reference tiling of the time-frequency plane into rectangular "super-tiles", denoted by $\square_{(s)}$,

$$\mathbb{R}^2 = \bigcup_s \square_{(s)} \tag{9}$$

where the super-tiles are defined by

$$\square_{(s)} = \square_{m,n} = \left[\left(m - \frac{1}{2} \right) A, \left(m + \frac{1}{2} \right) A \right) \times \left[\left(n - \frac{1}{2} \right) B, \left(n + \frac{1}{2} \right) B \right) . \tag{10}$$

The supports of the Gabor atoms associated with windows g^1 and g^2 provide different pavings of these supertiles. For each time-frequency lattice $\mathcal{L}^r, r = 1, 2$, let $\mathcal{L}^{r,s} = \mathcal{L}^r \cap \square_{(s)}$ denote the subset of $\mathcal{L}^r$ included in the super-tile $\square_{(s)}$. Illustrations of super-tiles and corresponding pavings and sub-lattices, in the case of two windows (in fact, the same window, at two different scales) may be found in Fig. 4.

To any $x \in L^2(\mathbb{R})$, any window $g^r, r = 1, 2$ and any super-tile s, associate the set of Gabor coefficients $G_x^r(m, n)$ (defined in (5)) corresponding to the paving of super-tile s with atoms of type r:

$$\mathbf{G}_{x,s}^r = \{ \langle x, g_{mn}^r \rangle, \ m, n \in \mathcal{L}^{r,s} \} .$$

Then, for each super-tile s, the optimal window $g^{\mathbf{r}(x,s)}$ is selected, and the corresponding entropy is computed, according to

$$\begin{cases} \mathbf{r}(x, s) & = \arg\min_{r=1,2} R_\alpha(\mathbf{G}_{x,s}^r) , \\ R_\alpha(x, s) & = R_\alpha(\mathbf{G}_{x,s}^{\mathbf{r}(x,s)}) . \end{cases} \tag{11}$$

Given $x \in L^2(\mathbb{R})$, let the first approximation be defined as

$$x^{(1)} = \sum_s \sum_{\lambda \in \mathcal{L}^{\mathbf{r}(x,s),s}} \langle x, g_\lambda^{\mathbf{r}(x,s)} \rangle \, \tilde{g}_\lambda^{\mathbf{r}(x,s)} \tag{12}$$
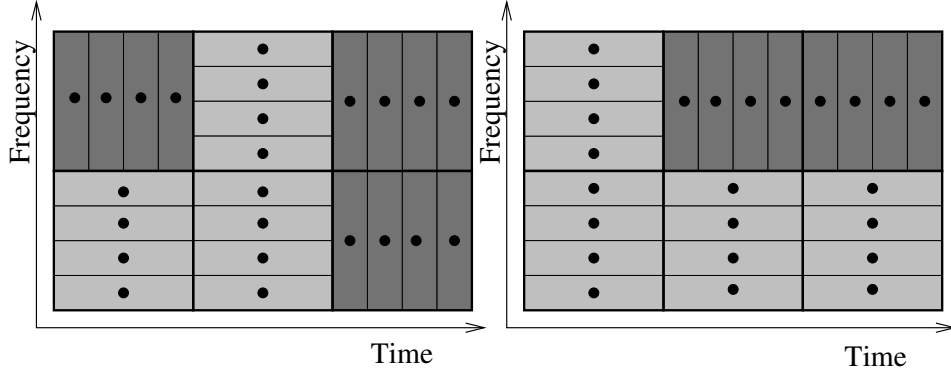
Fig. 4. Two different tilings of a region of the time-frequency plane: within a given super-tile (rectangular region with thick borders) the smaller rectangles represent the "numerical" support of the time-frequency atoms with two different windows (say, the domain within which their spectrogram exceeds some fixed threshold), and the dots represent their center, i.e. the time-frequency sampling points.

and the corresponding residual

$$\mathcal{R}^1(x) = x - x^{(1)} \ . \tag{13}$$

The same procedure may then be applied to the so-obtained residual: search for the "optimal" window within each supertile, compute the corresponding approximation to the first order residual, and a second order residual. This approximation of the first order residual may be added to $x^{(1)}$, to yield a (hopefully better) new approximation of the signal.

The procedure may in fact be applied recursively, by setting

$$x^{(k)} = \sum_{s} \sum_{\lambda \in \mathcal{L}(x,k,s)} \langle \mathcal{R}^{k-1}(x), g_\lambda^{\mathbf{r}(x,k,s)} \rangle \ \tilde{g}_\lambda^{\mathbf{r}(x,k,s)} \ , \tag{14}$$

and

$$\mathcal{R}^k(x) = \mathcal{R}^{k-1}(x) - x^{(k)} \ . \tag{15}$$

where

$$\mathbf{r}(x,k,s) = \mathbf{r}(\mathcal{R}^{k-1}(x),s)$$

is the label of the window selected at step k within the supertile s, and $\mathcal{L}(x,k,s) = \mathcal{L}^{\mathbf{r}(x,k,s),s}$ denotes the corresponding time-frequency sampling points.

At step K, we then obtain a telescopic expansion of the signal into K approximation levels and a residual

$$x = \sum_{k=1}^{K} x^{(k)} + \mathcal{R}^K(x) \ , \tag{16}$$

and the iteration stops when the residual is small enough, yielding an (approximate) expansion of the signal of the form

$$x \approx \sum_{k=1}^{K} x^{(k)} = \sum_{k=1}^{K} \sum_{s} \sum_{\lambda \in \mathcal{L}(x,k,s)} \langle \mathcal{R}^{k-1}(x), g_\lambda^{\mathbf{r}(x,k,s)} \rangle \ \tilde{g}_\lambda^{\mathbf{r}(x,k,s)} \ . \tag{17}$$

Pseudocode for this algorithm may be found in the appendix.

Remark 1: Proving the convergence of such a scheme seems to be a difficult task. Numerical experiments (see below) show that the convergence is indeed very fast, and seem to indicate exponential convergence (like matching pursuit in finite dimensional situations). Criteria ensuring that the set of selected atoms is a frame in the signal space may be obtained[6]. However, notice also that the convergence of this algorithm does not require such a property. It would be enough to prove an upper bound for the residuals.

The convergence of the algorithm is illustrated by a number of case studies. We display in Fig. 6 the evolution of the Signal to Noise Ratio

$$SNR(k) = 10 \log_{10} \left(\frac{\|x\|^2}{\|\mathcal{R}^k(x)\|^2} \right)$$

as a function of the iteration index k, for three different signals: two synthetic signals (a sum of sine waves and Dirac pulses, and a Gaussian white noise), and a real signal (the Glockenspiel signal displayed in Fig. 5). As may be seen, after about 6 iterations, the three curves are essentially parallel and straight, which seems to indicate exponential convergence, at the same speed for the three considered signals. As could be anticipated, the convergence is faster for the synthetic signal made out of sine waves and Dirac pulses, and slower for the white noise, which is not sparse for wide neither narrow windows. However, even in the worst case, the convergence is still very good, as 13 iterations are sufficient to reach a SNR equal to 100 dB, an acceptable limit for audio applications.

The speed of convergence also depends on the choices of the other parameters of the algorithm, including the widths of windows, and the sizes of supertiles. For example, we display in Fig. 7 the SNR as a function of the number of iterations for three different supertile sizes. We define a supertile to be of size $M \times N$ if it contains M time sampling points of the wide windows and N frequency sampling points of the narrow window. Fig. 7 shows that the convergence is better for large supertiles. This may be easily interpreted as a consequence of the fact that the main contributions to the residual signals $\mathcal{R}^k(x)$ originate from the boundaries between supertiles within which different windows have been chosen. When the area of the supertiles grows, the number of boundaries decreases, and the precision is better. This effect seems to be in favour of large supertiles, to insure faster convergence. However, since the original
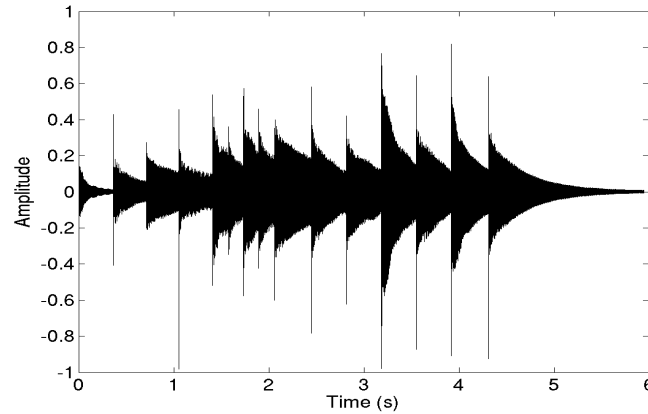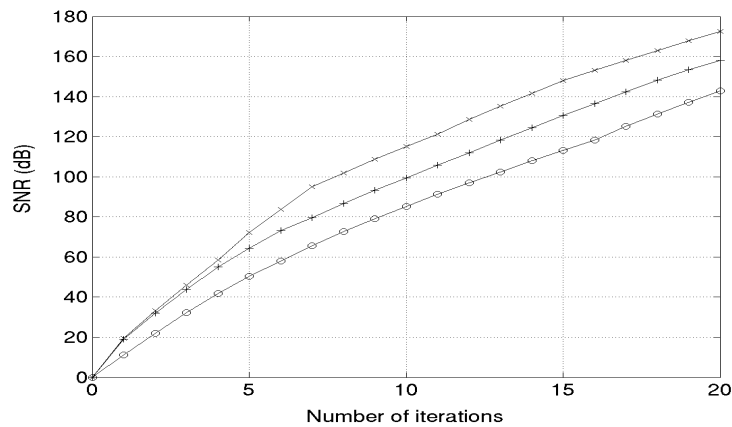
Fig. 5.   Glockenspiel test signal

Fig. 6.   Convergence of TFJP1, influence of the signal. SNR as a function of the iteration index. ○: white noise; +: glockenspiel signal; ×: sum of sine waves and Dirac pulses.

goal is to reach a good localization in small regions of the time-frequency plane, there is a trade-off to find between these two objectives.

Remark 2: For the sake of simplicity, we have chosen to illustrate the TFJP approach with the simple case of two windows. This choice is also natural in the case of windows of the same shape and different widths, because the results may be given a simple interpretation (see Section IV below). However, this is not a limitation of the method, and numerical results (not displayed here) show that the algorithm works as well with more than two windows, with similar convergence properties.
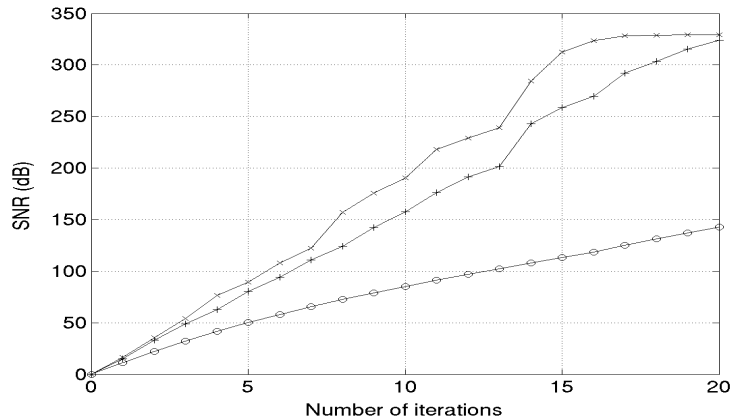
Fig. 7. Convergence of TFJP1, influence of the sizes of the supertiles. SNR as a function of the iteration index. ○: supertile of size 1×1; +: supertile of size 3×3; ×: supertile of size 5×5.

B. Multilayered time-frequency decomposition

The main idea of the method is that each window selects parts of the signal which are sparsely represented in the corresponding Gabor system. We now organize differently the multiple Gabor expansion, by summing up all contributions selected by a given window. We define r-th layer of the signal as the signal constructed by limiting the sum in (17) to Gabor atoms $\tilde{g}_{mn}^r$.

Remark 3: When it comes to numerical experiments, we shall limit ourselves to the particular case of two identical windows, at two different scales: a wide version and a narrow version. This choice is motivated by the desire of decomposing audio signals into "tonal" and "transient" layers. In this spirit, the tonal layer of a signal is defined as the "component" which admits a sparse expansion with respect to a Gabor frame with high frequency resolution (i.e. with a wide window), and the transient layer as the "component" which admits a sparse expansion with respect to a Gabor frame with high time resolution (i.e. a narrow window).

Given an expansion of the type (16), each approximation level is itself expressed as a linear combination of Gabor atoms with different window functions: for $r = 1, 2$, at a given step k, let

$$S_x^r(k) = \{s : \mathbf{r}(x, k, s) = r\}$$

denote the set of time-frequency indices for which window r has been selected, and denote by $x^{(k;r)}$ the contribution of atoms g_{mn}^r at step k:

$$x^{(k;r)} = \sum_{s \in S_x^r(k)} \sum_{\lambda \in \mathcal{L}(x,k,s)} \langle \mathcal{R}^{k-1}(x), g_\lambda^r \rangle \tilde{g}_\lambda^r . \tag{18}$$

Clearly, $x^{(k)} = x^{(k;1)} + x^{(k;2)}$, and the signal decomposition (16) may be rewritten as

$$x = \ell^1(x) + \ell^2(x) + \mathcal{R}^K(x) \ , \tag{19}$$

where the remainder $\mathcal{R}^K(x)$ is as before, and ℓ^1 and ℓ^2 are <u>layers</u> of x, defined according to

$$\ell^r(x) = \sum_{k=1}^{K} x^{(k;r)} \ . \tag{20}$$

The layer $\ell^1(x)$ (resp. $\ell^2(x)$) basically represents the "component" of the signal x which is "well represented" (i.e. sparsely represented) by the Gabor frame $\mathcal{F}^1$ (resp. $\mathcal{F}^2$). Actually, if the two windows have sufficiently different characteristics (in particular, time-frequency localization properties), the different layers do indeed represent significantly different components of the signal. Applications of this technique to the decomposition of audio signals into transient and tonal layers will be discussed in Section IV below.

Remark 4: The approach described above treats the two layers equally, in the sense that at each iteration, the construction of the time-frequency puzzle is followed directly by the estimation of corresponding contributions to all layers. However it turns out that in such a scheme, the estimate of the second layer may be perturbed by the first one and vice versa, as shown by the following simple example. Consider the simple case of the sum of a sine wave and a Dirac pulse. The sine wave is expected to be well represented by a Gabor frame with a wide window, and the Dirac pulse by a Gabor frame with a narrow window. Indeed, numerical calculations show that wide windows are selected in the neighborhood (in the frequency domain) of the frequency of the sine wave, and narrow ones in the (time) neighborhood of the location of the Dirac pulse. However, near the "intersection" of these two neighborhoods in the time-frequency domain, the poor frequency localization of the narrow windows implies that the latter "captures" a part of the energy of the sine wave. The sine wave is then accounted for twice, which biases the decomposition. A similar effect can also be observed on the Dirac pulse. To better resolve such situations, a slight modification may be done on the algorithm, described below.

C. A simple variant: TFJP2

To avoid the shortcomings mentioned in Remark 4, it is possible to modify slightly the algorithm, and only estimate a single layer at each iteration. This is variant 2 of the Time Frequency Jigsaw Puzzle method (TFJP2 for short). More precisely, assume for the sake of simplicity that two windows are given.

Given a signal x, the first step is still given in the same way as (11)

$$\begin{cases} \mathbf{r}(x,s) & = \arg\min_{r=1,2} R_\alpha(\mathbf{G}^r_{x,s}) \ , \\ R_\alpha(x,s) & = R_\alpha(\mathbf{G}^{\mathbf{r}(x,s)}_{x,s}) \ , \end{cases} \tag{21}$$

but the first estimate only takes into account the first window

$$x^{(1;1)} = \sum_{s:\mathbf{r}(x,s)=1} \sum_{\lambda \in \mathcal{L}^{1,s}} \langle x, g^1_\lambda \rangle \, \tilde{g}^1_\lambda \tag{22}$$

which also defined the corresponding residual

$$\mathcal{R}^{\frac{1}{2}}(x) = x - x^{(1;1)} \ . \tag{23}$$

The second window is then used to estimate the contribution to the second layer: (21) is used again, and the second estimate reads

$$x^{(1;2)} = \sum_{s:\mathbf{r}(\mathcal{R}^{\frac{1}{2}}(x),s)=2} \sum_{\lambda \in \mathcal{L}^{2,s}} \langle \mathcal{R}^{\frac{1}{2}}(x), g^2_\lambda \rangle \, \tilde{g}^2_\lambda \ . \tag{24}$$

The residual is then

$$\mathcal{R}^1(x) = \mathcal{R}^{\frac{1}{2}}(x) - x^{(1;2)} = x - x^{(1,1)} - x^{(1;2)} \ . \tag{25}$$

Again, the procedure may be iterated, taking the residuals $\mathcal{R}^k$ as inputs: replacing x with $\mathcal{R}^k$ in Equations (21) to (24) yields similarly

$$x^{(k;1)} = \sum_{s:\mathbf{r}(\mathcal{R}^{k-1}(x),s)=1} \sum_{\lambda \in \mathcal{L}^{1,s}} \langle \mathcal{R}^{k-1}(x), g^1_\lambda \rangle \, \tilde{g}^1_\lambda \ , \tag{26}$$

$$x^{(k;2)} = \sum_{s:\mathbf{r}(\mathcal{R}^{k-\frac{1}{2}}(x),s)=1} \sum_{\lambda \in \mathcal{L}^{2,s}} \langle \mathcal{R}^{k-\frac{1}{2}}(x), g^2_\lambda \rangle \, \tilde{g}^2_\lambda \ , \tag{27}$$

and residuals

$$\mathcal{R}^{k-\frac{1}{2}}(x) = \mathcal{R}^{k-1}(x) - x^{(k,1)} \ , \quad \mathcal{R}^k(x) = \mathcal{R}^{k-\frac{1}{2}}(x) - x^{(k;2)} \ . \tag{28}$$

As a result, one obtains a telescopic series

$$x = \sum_{k=1}^K \left(x^{(k,1)} + x^{(k,2)} \right) + \mathcal{R}^K(x) \tag{29}$$

as well as two layers

$$\ell^1(x) = \sum_{k=1}^K x^{(k,1)} \ , \quad \ell^2(x) = \sum_{k=1}^K x^{(k,2)} \ , \tag{30}$$

with the same interpretation as before. Pseudocode for this algorithm may be found in the appendix.

As mentioned above, this variant has the advantage of better avoiding boundary effects between adjacent super-tiles in which different windows are chosen. It also yields slightly better convergence, as may be
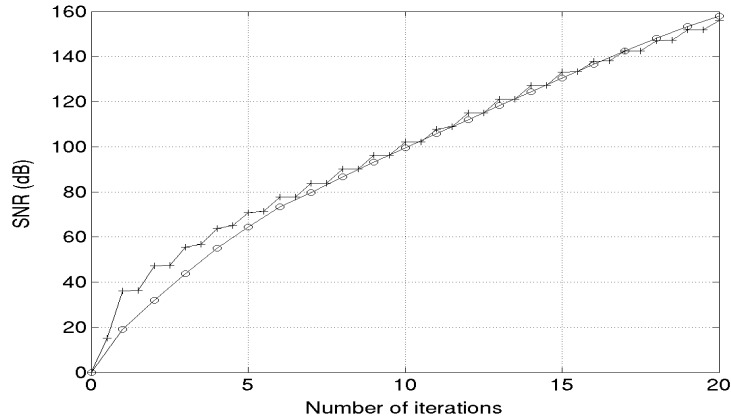
Fig. 8. Convergence of TFJP2, comparison with TFJP1, influence of the sizes of the supertiles. SNR as a function of the iteration index. ○: TFJP1; +: TFJP2.

seen in Fig. 8. Notice however that the difference between the two curves essentially lies in the first iterations. After a few iterations, the two curves essentially coincide.

Clearly, the multilayer expansion of signals described in Section III-B above may be performed as well using TFJP2.

D. Introducing significance test for sparsity: TFJP1b

The main idea of the above algorithms is to choose, within each super-tile s, the window such that the resulting entropy is minimal. However, the minimal entropy for a given super-tile may happen to be quite large, meaning that for that particular super-tile, even the "best" window was unable to yield a sufficiently sparse description. In such situations, it does not necessarily make sense to include the contribution of the considered super-tile in one of the layers, an alternative being to keep it inside the residual. We describe below this new approach (TFJP1b) in the framework of the TFJP1 algorithm (the modifications needed to adapt it to the TFJP2 algorithm are straightforward).

For a given super-tile s, and corresponding values of entropies $R_\alpha(\mathbf{G}_{x,s}^r)$, one has to decide whether or not those values are significant (i.e. correspond to actual significant signal component.) To avoid possible non-significant values, we decide that the optimal window defined in (11) is accepted only when the corresponding entropy is below some threshold value. Given such a threshold $\tau \in \mathbb{R}^+$, we simply

replace (12) and (13) with

$$x_\tau^{(1)} = \sum_{s:R_\alpha(x,s)\leq\tau} \sum_{\lambda\in\mathcal{L}^{\mathbf{r}(x,s),s}} \langle x, g_\lambda^{\mathbf{r}(x,s)} \rangle \, \tilde{g}_\lambda^{\mathbf{r}(x,s)} \tag{31}$$

$$\mathcal{R}_\tau^1(x) = x - x_\tau^{(1)} , \tag{32}$$

and similarly for the rest of the algorithm. The multilayered Gabor expansion may also be adapted accordingly, within the scheme depicted previously. This now produces a decomposition of the signal into three layers: the two previous ones, and a residual.

Pseudocode for this algorithm may be found in the appendix.

Remark 5: choice of the threshold. In some specific applications, the threshold τ may of course be chosen by the user. In a more general context, it may be desirable to choose the value(s) of the threshold on statistical grounds, which is however difficult, as it would require characterizing the distribution of Shannon's entropies computed from restrictions of Gabor transforms to super-tiles.

In our numerical experiments to be discussed below, we used the following procedure. The distribution of the entropies was estimated (numerically) from Gabor coefficients of a white noise reference signal. τ was then adjusted to a given significance level (for example, 5%). In other words, at each step of the iterative algorithm, super tiles were rejected (and kept in the residual signal) when the corresponding value of entropy was too likely to have been produced by a Gaussian white noise (considered the worst case signal, as far as sparsity is concerned). In such situations, the residual has no reason to converge to zero, and may even contain interesting signal which simply cannot be sparsely represented by the considered systems of time-frequency atoms.

IV. Applications

The approach we have presented in this article provides alternative representations for signals. While the Time-Frequency Jigsaw Puzzle approach does not solve signal processing problems by itself, the signal representations it provides facilitates further tasks. We give below a couple of case studies to illustrate the soundness of the approach.

A. *Application to tonal/transient separation in audio signals*

Audio signals often feature significantly different "components", (partials, transients, chirps, noise,...), which are sometimes associated with different physical processes. The TFJP approach offers a way of separating such components, via a suitable choice of the analyzing windows.
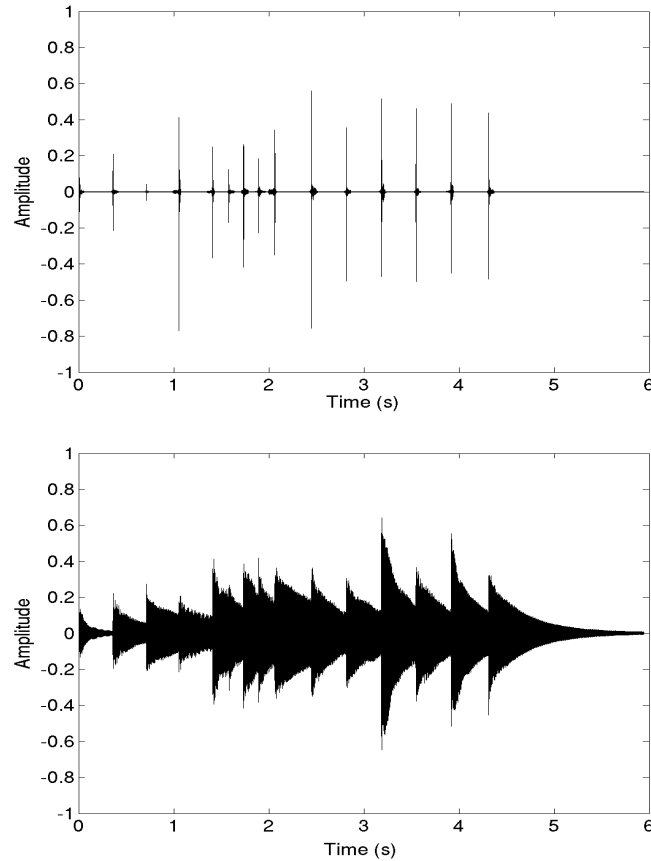
Fig. 9.   Glockenspiel test signal: transient layer (top) and tonal layer (bottom).

We illustrate this fact by an example of separation of transient components (i.e. the attacks, or onsets) and tonal components (i.e. resonances) from the glockenspiel signal already presented in Fig. 5. This problem is of interest in various contexts, including audio coding, where it may be efficient to encode the two layers separately[4], [15], or sinusoidal/harmonic signal modeling, where the estimation of sinusoidal components is obviously more precise when performed on the tonal layer rather than the complete signal[10].

In that case, two Gaussian windows with different width (a wide window and a narrow window) were used in the framework of TFJP2[3], and the two corresponding layers ℓ^1 and ℓ^2 were reconstructed. The resulting waveforms are shown in Fig. 9, and it appears clearly that the two components have been very neatly separated.

---

[3]TFJP2 was prefered to TFJP1 because of the point raised in Remark 4.

Time-frequency representations of these two layers are shown in Fig. 10. The separation of the two components appears clearly there too.
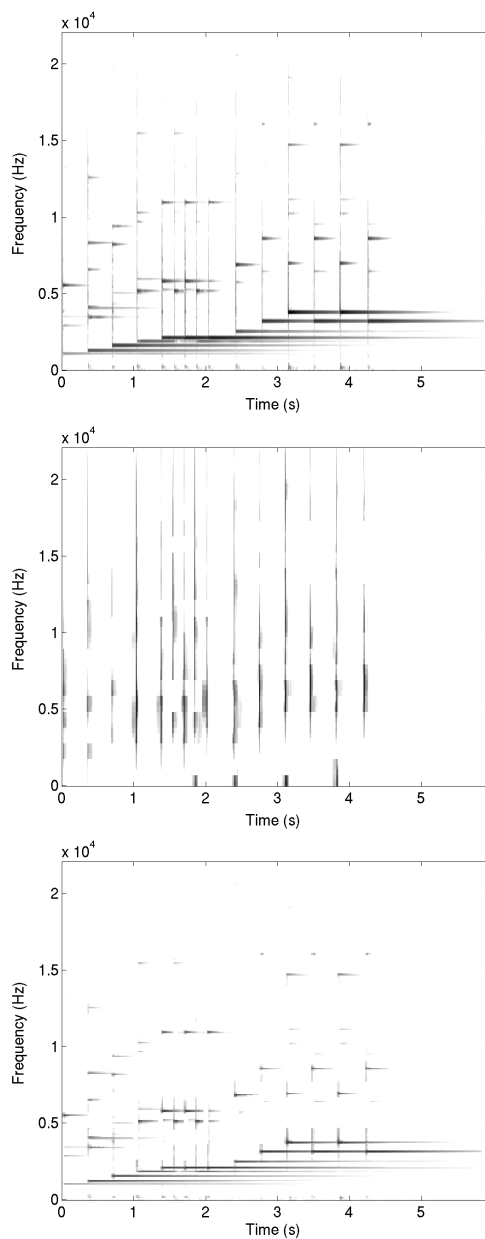

Fig. 10. Multilayered decomposition of the "Glockenspiel" signal, obtained using TFJP2. From top to bottom: time-frequency representations (spectrograms) of the original signal (with a "medium size" window), the estimated transient layer (with the narrow window) and the estimated tonal layer (with the wide window).

Real signals often feature noise, or stochastic-like components. In such situations, TFJP1b is to be
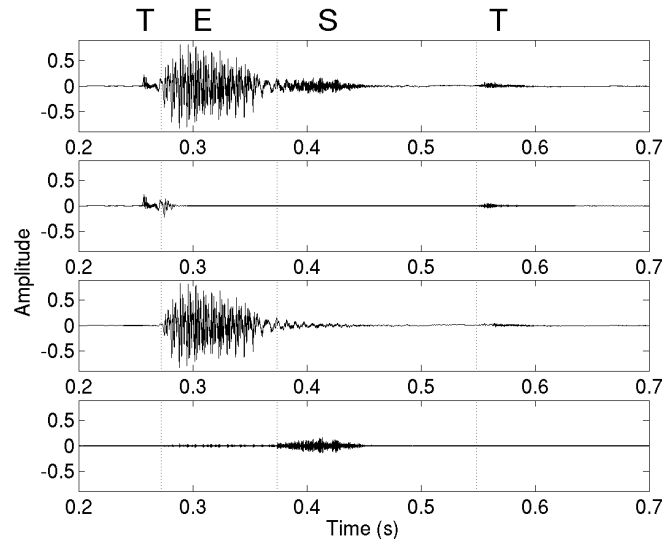
Fig. 11. Multilayered decomposition of a short piece of speech signal: /test/, obtained using TFJP1b. From top to bottom: waveforms of the original signal, the transient layer, the tonal layer and the residual signal.

preferred. To illustrate this situation, we display in Figure 11 the results of the decomposition obtained using TFJP1b on a speech signal: the word /test/. Remarkably enough, the algorithm was able to separate the different letters of the signal: the t are captured by the transient layer, the e by the tonal layer, and the s remain in the residual. However, such results turn out to be quite sensitive to shifts of the super tiles. Therefore, a systematic exploration of such approaches for speech signal processing will require extra tuning effort, which we plan to study in the future.

B. Perspective: towards a new approach for transient detection

Transient detection is a classical problem in signal analysis; however, a comparison of existing approaches is difficult to perform because, strictly speaking, there is no general agreement on the definition of a transient signal. We suggest that transientness may be defined in terms of sparsity of some narrow window Gabor representation, and show how to adapt the TFJP method in such a context. Given a signal x, a time-varying measure of *transientness*[14] may be associated with it as follows. One may for example compute, for each value of the time index associated to a super tile, the number of times the narrow window has been preferred to the wide one, or related quantities. This results in a time series exhibiting sharp peaks at the location of transients, from which transient detection may then be carried out by simple thresholding strategies.
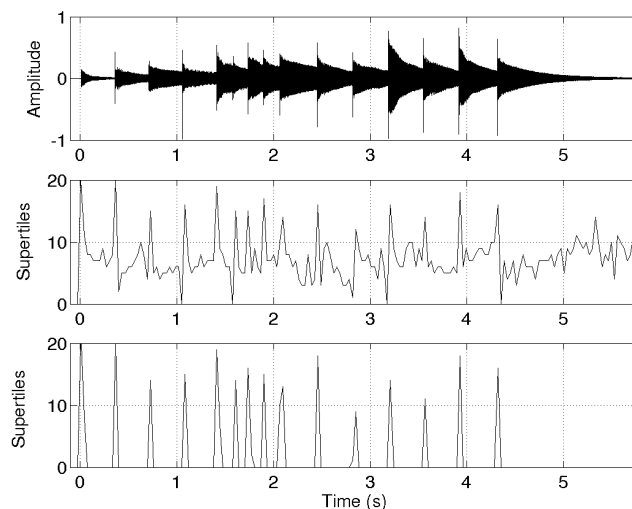
Fig. 12. Application to transient detection. From top to bottom: glockenspiel signal; time evolution of the number of supertiles where the narrow window has been selected at the first iteration of TFJP1; time evolution of the number of supertiles where the narrow window has been selected at the first iteration of TFJP1, and the entropy was below some threshold.

An illustration on the very simple case of the Glockenspiel signal may be found in Fig. 12, where two different countings of the selected windows are displayed. The middle plot represents the time evolution of the number of rectangles for which the narrow window has been selected in the first iteration of TFJP1. As may be seen, transients may be detected by thresholding. However, as often, the choice of the threshold is an important issue, as may be seen on the right hand side of the middle plot. In this approach, the peaks appearing there are not significant, because they originate from narrow windows that have been selected even though their entropy was large (but still smaller than that of the wide windows). To overcome such a problem, a threshold on the entropy values may be introduced. The bottom plot shows the time-evolution of the number of supertiles for which the narrow has been selected, and the entropy was below some threshold. In this case, the choice of the threshold on the curve is not crucial any more.

Of course, the transient detection task will be much more difficult for more complex signals, and such a simple approach will have to be refined[10]. Again, the purpose of this discussion was not to propose a new transient detection algorithm, but rather to show how the TFJP algorithms may help in this context.

V. Conclusion

We have presented in this paper a new approach for selection of adapted Gabor signal representations, starting from several "standard" Gabor expansions with different window functions. The main idea behind this approach is to exploit potential sparsity of the representation of some components of the analyzed signal in appropriate representations. Our approach provides ways of finding such appropriate representations. This approach is quite general, and may be adapted in various ways. We have also presented some of these variations, including a supervised version (adapted representation) and an unsupervised one (adaptive representation, the TFJP algorithms).

As a by-product, this approach also yields "multilayered" representations for the signal under study, a layer being defined as the "component" of the signal that is well represented by a given type of Gabor functions.

Even though we have focused here on a few illustrations on general audio and speech signals and transient/tonal separation, we believe that such approaches possess a much wider application range. To quote only a few of these, applications to blind source separation or automatic speech segmentations are examples of applications which we plan to address in the near future.

Additional material, including additional figures, and sound files, may be found on a companion web site:

$$\text{http://www.cmi.univ-mrs.fr/~torresan/papers/TFJP}$$

APPENDIX

A. TFJP1

For a signal x:

- Choose a value for $\alpha \in (0, 1]$. Choose two windows g^1, g^2 and corresponding sampling lattices; choose supertiles.

- Initialization: Set $\mathcal{R}^0 = x$

- Main loop: while $\|\mathcal{R}^k\| \geq \epsilon$, do

  - Compute coefficients $\langle \mathcal{R}^k, g_{mn}^r \rangle$ for the two windows $r = 1, 2$.

  - Compute entropies R_α for both windows within each supertile.

  - For each supertile, retain the window with smallest entropy. Reconstruct corresponding contributions $x^{(k,1)}$ and $x^{(k,2)}$ to the layers.

  - Set $\mathcal{R}^{k+1} = \mathcal{R}^k - x^{(k,1)} - x^{(k,2)}$.

- Reconstruct layers ℓ^1 and ℓ^2 by summing up contributions $x^{(k;1)}$ and $x^{(k;2)}$ respectively.

B. TFJP2

For a signal x:

- Choose a value for $\alpha \in (0, 1]$. Choose two windows g^1, g^2 and corresponding sampling lattices; choose supertiles.

- Initialization: Set $\mathcal{R}^0 = x$

- Main loop: while $\|\mathcal{R}^k\| \geq \epsilon$, do

  - Compute coefficients $\langle \mathcal{R}^k, g_{mn}^r \rangle$ for the two windows $r = 1, 2$.

  - Compute entropies R_α for both windows within each supertile.

  - Select supertiles for which window 1 yields the smallest entropy. Reconstruct corresponding contribution $x^{(k,1)}$ to layer 1.

  - Set $\mathcal{R}^{k+1/2} = \mathcal{R}^k - x^{(k,1)}$.

  - Compute coefficients $\langle \mathcal{R}^{k+1/2} g_{mn}^r \rangle$ for the two windows $r = 1, 2$.

  - Compute entropies R_α for both windows within each supertile.

  - Select supertiles for which window 2 yields the smallest entropy. Reconstruct corresponding contribution $x^{(k,2)}$ to layer 2.

  - Set $\mathcal{R}^{k+1} = \mathcal{R}^{k+1/2} - x^{(k,2)}$.

- Reconstruct layers ℓ^1 and ℓ^2 by summing up contributions $x^{(k;1)}$ and $x^{(k;2)}$ respectively.

C. TFJP1b

For a signal x:

- Choose a value for $\alpha \in (0,1]$. Choose two windows g^1, g^2 and corresponding sampling lattices; choose supertiles. Define a maximal number of iterations K. Choose a threshold τ for entropies.

- Initialization: Set $\mathcal{R}_\tau^0 = x$

- Main loop: for $k = 0 : K$, do

  - Compute coefficients $\langle \mathcal{R}_\tau^k, g_{mn}^r \rangle$ for the two windows $r = 1, 2$.

  - Compute entropies R_α for both windows within each supertile.

  - Select supertiles for which the smallest of the two entropies is below the threshold τ, and reconstruct the corresponding contributions $x_\tau^{(k,r)}$ to the two layers.

  - Set $\mathcal{R}_\tau^{k+1} = \mathcal{R}_\tau^k - x_\tau^{(k,1)} - x_\tau^{(k,2)}$.

- Reconstruct layers ℓ^1 and ℓ^2 by summing up contributions $x_\tau^{(k;1)}$ and $x_\tau^{(k;2)}$ respectively. Store the residual.

REFERENCES

[1] R. Carmona, W.L. Hwang, and B. Torrésani. *Practical Time-Frequency Analysis: continuous wavelet and Gabor transforms, with an implementation in S*, volume 9 of *Wavelet Analysis and its Applications*. Academic Press, San Diego, 1998.

[2] L. Cohen, *Time-Frequency Analysis*, Prentice Hall (1994), New York.

[3] I. Daubechies. *Ten Lectures on Wavelets*, volume 61 of *CBMS-NFS Regional Series in Applied Mathematics*, 1992.

[4] L. Daudet and B. Torrésani. Hybrid representations for audiophonic signal encoding. *Signal Processing*, 82(11):1595–1617, 2002. Special issue on Image and Video Coding Beyond Standards.

[5] M. Dörfler, *Gabor Analysis for a Class of Signals called Music*, PhD thesis, NuHAG, Vienna, 2003.

[6] M. Dörfler, private communication (August 2005).

[7] H.G. Feichtinger and T. Strohmer, *Gabor analysis and algorithms, theory and applications*, Birkhäuser Boston Inc., 1998.

[8] P. Flandrin, *Time-Frequency/Time-Scale Analysis*, volume 10 of *Wavelet Analysis and its Applications*. Academic Press, San Diego, 1999.

[9] R. Gribonval. *Approximations non-linéaires pour l'analyse des signaux sonores*. PhD thesis, Université de Paris IX Dauphine, 1999.

[10] F. Jaillet. *Représentation et traitement temps-fréquence des signaux audionumériques pour des applications de design sonore*. PhD thesis, LATP Marseille, June 2005.

[11] F. Jaillet and B. Torrésani. *Remarques sur l'adaptativité des représentations temps-fréquence*. Proceedings of the symposium *GRETSI'03*, Paris, France, 2003.

[12] F. Jaillet and B. Torrésani. *Adaptive time-frequency representation for sound analysis and processing*. Proceedings of the symposium *Congrès Français d'Acoustique*, Strasbourg, France, 2004.

[13] S. Mallat and Z. Zhang. Matching pursuits with time-frequency dictionaries. *IEEE Transactions on Signal Processing*, 41:3397–3415, 1993.

[14] S. Molla and B. Torrésani. Determining local transientness of audio signals. *IEEE Signal Process. Lett.* **11** (7) (July 2004) pp. 625-628.

[15] S. Molla and B. Torrésani. An Hybrid Audio Scheme using Hidden Markov Models of Waveforms *Applied and Computational Harmonic Analysis* **18** (2005), pp. 137-166.

[16] A. Rényi, On Measures of Entropy and Information. Proc. Fourth Berkeley Symp. Math. Stat. and Probability, Vol. 1. Berkeley, CA: University of California Press, pp. 547-561, 1961.

[17] M. V. Wickerhauser. *Adapted Wavelet Analysis from Theory to Software*. AK Peters, Boston, MA, USA, 1994.

[18] M. Zibulski and Y. Zeevi, Analysis of multiwindow Gabor-type schemes by frame methods, *Applied and Computational Harmonic Analysis*, **4**, Vol. 2 (1997), pp. 188-212.