Arbitrary Tilings of the Time–Frequency Plane Using Local Bases

Riccardo Bernardini and Jelena Kovačević, *Senior Member, IEEE*

***Abstract*— We show how to design filters given a prescribed tiling of the time–frequency plane. Moreover, we impose on these filters the structure of local orthogonal bases. These bases were recently constructed as a generalization of the cosine-modulated filter banks in discrete time and local trigonometric bases in continuous time. They have been found to be of considerable practical importance due to their simplicity (all filters are obtained from a single prototype) and low computational complexity. We show examples of design, in particular, that of a critical-band system for use in audio coding.**

I. Introduction

ONE OF THE main goals of signal analysis in recent years has been to develop mixed signal representations in terms of elementary blocks called time–frequency atoms well localized in time and frequency [1]. Each of these blocks resides mostly[1] in a well-defined area (usually a rectangle) in the time–frequency plane. An appropriate analogy is that of a musical score; knowing that there exists a C within a first few bars of a musical piece does not help us unless we know its position in time and its duration. Thus, we can think of notes as time–frequency atoms [1].

Currently, two classes of time–frequency atoms are in use, corresponding to two types of analyzes that Ville proposed: The first, wavelet packets [2], splits the signal first in frequency and then in time, whereas the second, local cosine bases [3]–[5], does the opposite, that is, it slices first in time and then in frequency. Our approach in this work produces a mixture of these two classes leading to filters corresponding to more general tilings. Refer also to [6] and [7] for some interesting works dealing with time–frequency tilings.

In discrete time, local cosine bases (which are also called cosine-modulated filter banks, modulated lapped transforms, or MDCT), have been in use for some time [8]–[11]. Due to a few of their properties, they have become quite popular, for example, all filters (basis functions) of a filter bank are obtained by appropriate modulation of a single prototype filter. Then, fast algorithms exist, making them very attractive for implementation. Finally, they have been used to achieve time-varying splittings of the time–frequency plane [12]. Their continuous counterpart is termed "Malvar's wavelets" [1].

Manuscript received December 6, 1995; revised December 17, 1998. The associate editor coordinating the review of this paper and approving it for publication was Dr. Nurgun Erdol.

R. Bernardini is with the University of Padova, Padova, Italy (e-mail: bernardi@dei.unipd.it).

J. Kovačević is with Bell Laboratories, Murray Hill, NJ 07974 USA (e-mail: jelena@bell-labs.com).

Publisher Item Identifier S 1053-587X(99)05412-4.

[1] By mostly, we mean that most of its energy lies within a prescribed region.

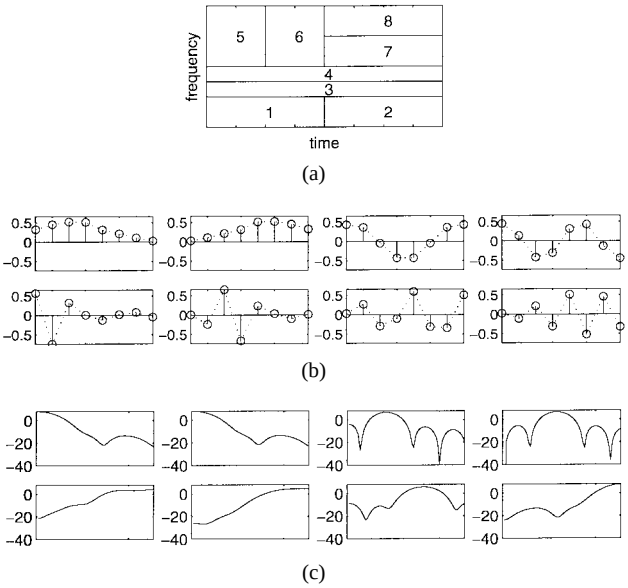


Fig. 1. (a) Desired tiling of the time–frequency plane. There are eight basis functions, denoted 1 through 8. The highest frequency value is π. (b) Impulse responses of the eight basis functions, numbered 1 through 8, starting from the upper left. (c) Magnitude frequency responses of the eight basis functions, numbered 1 through 8, starting from the upper left. Horizontal axis is the frequency axis, and the highest frequency is π. Values on the vertical axis are plotted on the log scale.

Local cosine bases have been used extensively in audio coding [13], [14]. They have also found use in image coding, due to the reduction of blocking effects [15] when compared with the DCT. Some video works contain local cosine bases as well [16]. Recently, local cosine bases were generalized to multiple dimensions as well as into local bases with noncosine modulating functions. The resulting bases are called local orthogonal bases [17], [18].

Here, we show how to construct time–frequency atoms leading to filters corresponding to fairly arbitrary tilings of the time–frequency plane using these local bases. Moreover, we also obtain time-varying tilings since the transitions between various tilings depend only on the windows used.[2] As an example of what we are trying to achieve, consider Fig. 1. We want to build a basis of size 8, where the basis functions $\mathbf{h}_1, \ldots, \mathbf{h}_8$ are localized in time and frequency as in Fig. 1(a). These basis functions can be thought of as time–frequency atoms. As a result of techniques developed in this paper, eight basis functions are obtained whose impulse and magnitude

[2] The only requirement for switching the tilings is that the tails of the windows across the boundary be power complementary [17].

frequency responses are given in Fig. 1(b) and (c), respectively (note, however, that the basis in this figure is not orthogonal).

II. Summary of Local Orthogonal Bases

In this section, we give a brief summary of the work in [17] that is necessary for the developments in the present paper. In [17], we extend the theory of local cosine bases (cosine-modulated filter banks) to a more general case. In a cosine-modulated filter bank, all the filters are obtained by modulating a prototype filter $w[n]$ (also known as the *window*) with suitable cosine functions. When the window satisfies certain constraints (symmetry and power complementarity), the resulting filter bank is orthogonal. In [17], we show that orthogonality depends only on the symmetries of the cosines; therefore, any other set of functions with the same symmetries would produce a perfect reconstruction filter bank. Consider the one-dimensional (1-D) case as an example. From [17], we can deduce a recipe to obtain a critically sampled orthogonal filter bank with M filters of length $2M$.

- Take any orthonormal basis $\mathbf{g}_i$, $i = 1, 2, \ldots, M$, and consider vectors $\mathbf{g}_i$ as signals $g_i[n]$ having support $\{0, 1, \ldots, M-1\}$. We refer to this basis as the *starting basis* and represent it by a matrix $\mathbf{G}$.
- Extend every $g_i[n]$ by symmetry and antisymmetry according to (we assume that M is even)

$$\hat{g}_i[n] \triangleq \begin{cases} g_i[-1-n], & \text{if } n = -M/2, \ldots, -1 \\ g_i[n], & \text{if } n = 0, \ldots, M-1 \\ -g_i[2M-1-n], & \text{if } n = M, \ldots, M+M/2-1. \end{cases} \tag{1}$$

  We refer to this process as *symmetry extension* and represent it by a matrix $\mathbf{K}$.
- Multiply every $\hat{g}_i[n]$ by the window $w[n]$ in order to obtain the final filters $h_i[n] = w[n]\hat{g}_i[n]$. We refer to this process as *windowing* and represent it by a matrix $\mathbf{W}$.

Signals $h_i[n]$ are the impulse responses of the desired filter bank. These steps are depicted in Fig. 4. Note that the above procedure can also be expressed as

$$\mathbf{H} = \mathbf{WKG} \tag{2}$$

where

- $\mathbf{G}$ $M \times M$ orthogonal matrix;
- $\mathbf{K}$ $2M \times M$ matrix representing symmetry extension;
- $\mathbf{W}$ $2M \times 2M$ diagonal matrix having the samples of the window $w[n]$ on the main diagonal.

The columns of matrix $\mathbf{H}$ are the impulse responses of the desired filters in a filter bank. A similar recipe can also be used in more than one dimension as well as in continuous time [17].

The aim of this paper is to exploit the new degrees of freedom obtained by relaxing the constraints that $\mathbf{G}$ be cosine to construct a filter bank, given a tiling of the time–frequency plane. Therefore, our goal is *to search for* $\mathbf{G}$ *such that the impulse responses contained in* $\mathbf{H}$ *given in (2) achieve a given tiling of the time–frequency plane*. In what follows, we assume that the window $w[n]$ is given (see [18] for more details on window design).

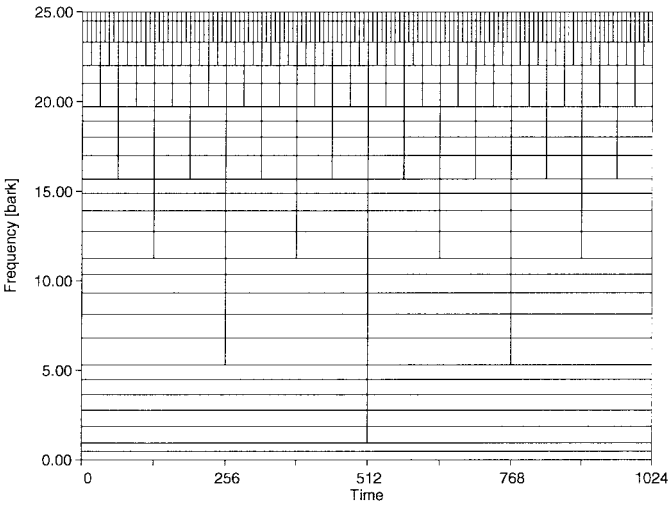


Fig. 2. Tiling of the time–frequency plane for critical-band audio coding. Frequency axis is given in barks where $B = 13.0 \arctan(0.76f/1000) + 3.5 \arctan(f/7500)^2$, with B being bark frequency and f frequency in Hertz.

III. Construction of Arbitrary Tilings

Our goal is to find a matrix $\mathbf{G}$ such that every $\mathbf{h}_i = \mathbf{WKg}_i$ is well concentrated in its corresponding tile. Although the multidimensional case is interesting in its own right (for use in image processing applications, for example), its study does not introduce anything new, except for a more complex notation. Therefore, for the sake of simplicity, here, we consider the 1-D discrete time-case.

Suppose $\mathbf{h}_i$ has to be well concentrated in the time interval $A_{T,i}$ and frequency interval $A_{F,i}$. In other words, let $A_{T,i} \times A_{F,i}$ be the tile corresponding to $\mathbf{h}_i$. As the measure of good concentration we use the sum $P_{T,i} + P_{F,i}$ of the two stopband errors[3]

$$P_{T,i} = \sum_{n \in A_{T,i}^C} h_i[n]^2, \quad P_{F,i} = \int_{A_{F,i}^C} |H_i(\omega)|^2 \, d\omega \tag{3}$$

where $H_i(\omega)$ is the Fourier transform of $h_i[n]$ [in (3), we used X^C to denote the complement of set X]. Cost functions (3) can be weighted with weight functions $\rho_T[n]$ and $\rho_F(\omega)$. By using, in (3), the definition of $H(\omega) = \sum_{n \in \mathbb{Z}} \exp(-j2\pi\omega n) h[n]$, we can rewrite (3) as

$$P_{T,i} = \mathbf{h}_i^T \mathbf{C}_{T,i} \mathbf{h}_i, \quad P_{F,i} = \mathbf{h}_i^T \mathbf{C}_{F,i} \mathbf{h}_i \tag{4}$$

with $\mathbf{C}_{T,i}$ and $\mathbf{C}_{F,i}$ two symmetric, positive semi-definite matrices. Clearly, the sum of two errors (4) is still a quadratic form with the matrix $\mathbf{C}_i = \mathbf{C}_{T,i} + \mathbf{C}_{F,i}$.

The same reasoning can be repeated for every vector $\mathbf{h}_i$. To obtain a cost function for the whole basis, use the sum of the cost functions of the single vectors. By exploiting the

[3]Note that we cannot use $P_{T,i}P_{F,i}$ as a measure of good concentration because it is easy to obtain a signal $h_i[n]$ having $P_{T,i} = 0$ implying $P_{T,i}P_{F,i} = 0$.

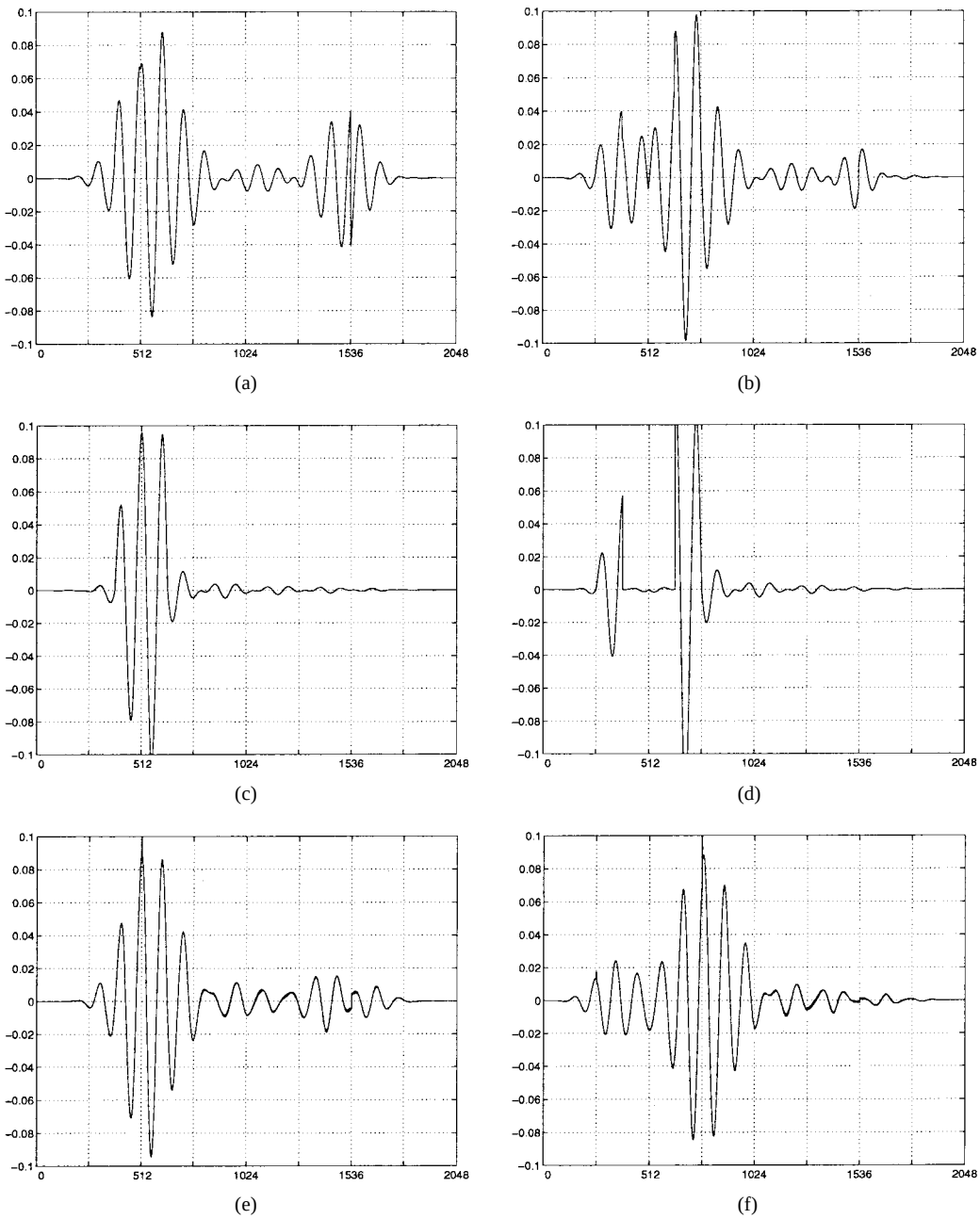


Fig. 3. Problems induced by the circular translation requirement. The filter in the figure corresponds to the eighth band. (a) Filter optimally designed *with* the circular translation constraint (note the discontinuity at the point 1536) and (b) its translation of 256. (c) Filter optimally designed *without* the circular translation requirement (that is, by optimizing only the first filter in each band) and (d) its translation of 256. Note that although the filter in (c) is smoother than the one in (a), its translation (d) is far worse than (b). (e) Filter obtained by using the pseudo-translation and (f) its pseudo-translation of 256.

dependence between $\mathbf{h}_i$ and $\mathbf{g}_i$ given in the previous section, the global cost function can be written as

$$\sum_i \mathbf{g}_i^T \mathbf{D}_i \mathbf{g}_i \tag{5}$$

where $\mathbf{D}_i = \mathbf{K}^T \mathbf{W}^T \mathbf{C}_i \mathbf{W} \mathbf{K}$.

IV. Solving for the Basis

We now describe ways to minimize (5) with the constraint that the vectors be orthogonal. A key observation is that if vectors $\mathbf{g}_i$ are orthogonal to one another and of unit norm, they can be organized as columns of a unitary matrix

$$\mathbf{U} \triangleq [\mathbf{g}_1 \quad \mathbf{g}_2 \quad \cdots \quad \mathbf{g}_N]. \tag{6}$$

This is important because the matrix $\mathbf{U}$, even if not square, can be written as a function of some free parameters. Thus, we can parameterize this matrix and then extract the vectors $\mathbf{g}_i$, which are now given as functions of free parameters. A unitary matrix $\mathbf{U}$ can be written either using Givens' rotations

$$\mathbf{U} = \mathbf{R}_1 \mathbf{R}_2 \cdots \mathbf{R}_M \mathcal{D}_G \tag{7}$$

where $\mathbf{R}_k$ are suitable rotation matrices, and $\mathcal{D}_G$ is a diagonal matrix having $d_{ii} = \pm 1$, or using Householder building blocks

(see, for example, [19])

$$\mathbf{U} = \mathbf{P}_1\mathbf{P}_2\cdots\mathbf{P}_{N-1}\mathcal{D}_H \tag{8}$$

where $\mathcal{D}_H$ is a diagonal matrix having $d_{ii} = \pm 1$, and each matrix $\mathbf{P}_k$ is defined as

$$\mathbf{P}_k \triangleq \mathbf{I} - 2\cdot\mathbf{u}_k\mathbf{u}_k^T \tag{9}$$

where $\mathbf{u}_k$ is a vector having the first $k-1$ components equal to zero such that $\|\mathbf{u}_k\| = 1$.

Note that matrices $\mathcal{D}_G$ and $\mathcal{D}_H$ have no effect on the value of (5) because they simply correspond to a change of sign for some of the vectors $\mathbf{g}_i$; therefore, we assume that $\mathcal{D}_G = \mathcal{D}_H = \mathbf{I}$.

A. Closed-Form Solution

Obtaining a closed-form solution is feasible only for small vector spaces with the help of a program capable of doing computation with polynomials.

The idea is to use the decomposition in (8). Write the entries of matrices $\mathbf{P}_k$ in (9) as polynomials of degree two in the nonzero components of $\mathbf{u}_k$. Therefore, $\mathbf{U}$, being a product of polynomial matrices, is a polynomial matrix as well, and the cost function (5) can be expressed as a polynomial of the free parameters. Our problem now becomes that of finding the minimum value assumed by a given polynomial on a multidimensional interval, and it can be solved by classical analytical methods. Such a polynomial has $N(N-1)/2$ free variables, and its degree is $4(N-1)$. For example, if $N = 10$, we have a polynomial of degree 36 with 45 variables. The gradient of such a polynomial is a vector having as components 45 polynomials of degree 35. Because of this, the closed-form solution is feasible only for small values of N. The fact that we can look at $\mathbf{U}$ as a polynomial matrix, however, can have theoretical relevance since it gives us strong information about the structure of (5).

If a closed-form solution is not feasible, use decompositions (7) and (8) to minimize (5) by using a program for unconstrained optimization.

B. A Simpler Solution

A simpler, even if suboptimal, approach that works also for bigger problems is to minimize each term of (5) separately, which in turn is obtained by finding the maximum eigenvalue of the matrix $\mathbf{C}_i$ and its corresponding eigenvector.

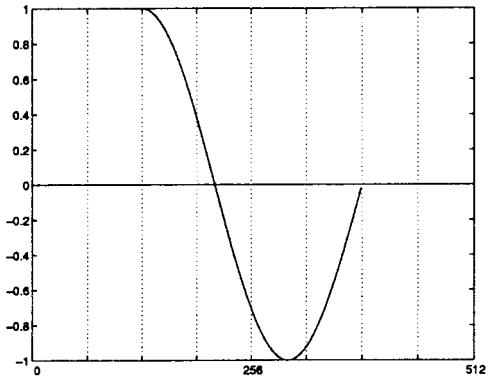
The resulting set of vectors is not guaranteed to be an orthonormal basis but can be orthogonalized. An interesting way to do so is to write the vectors obtained from the minimization of each term of (5) as the columns of a matrix $\mathbf{A}$ and decompose it, using the singular value decomposition, as

$$\mathbf{A} = \mathbf{OSQ} \tag{10}$$

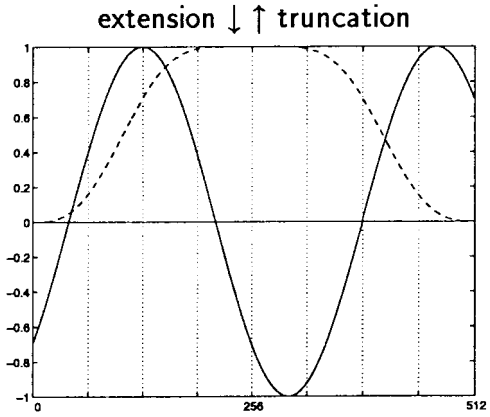
where $\mathbf{O}$ and $\mathbf{Q}$ are orthogonal matrices, and $\mathbf{S}$ is a diagonal matrix. The columns of

$$\mathbf{OQ} \tag{11}$$

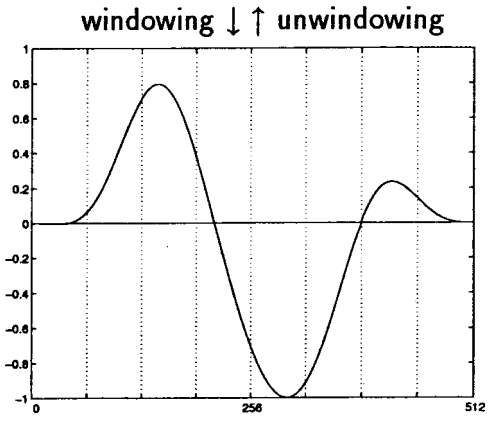
are orthogonal to one another and enjoy the property that they are the orthonormal basis having the minimum distance from the original set of vectors (the proof is given in Appendix A).



(a)



(b)



(c)

Fig. 4. Path between the starting basis vector and the final filter according to the procedure given in Section II. The basis vector in (a) is extended by symmetry and antisymmetry around the points $t = 128$ and $t = 384$. The resulting signal (b) (solid line) is multiplied by the window (dashed line) to obtain the filter in (c).

V. Design Examples

To start this section, we first use the techniques described in the previous section to obtain the tiling as given in Fig. 1(a). Although this is a simple example, it shows the results the

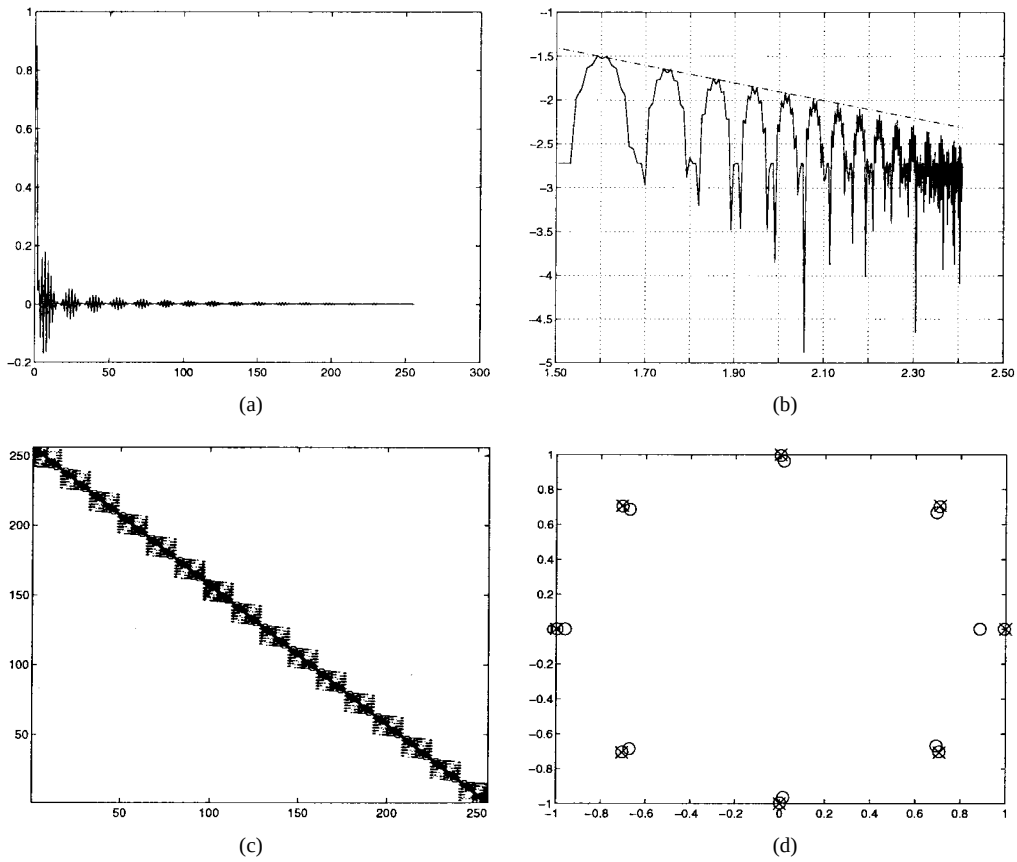


Fig. 5. Example of construction of an approximate translation of 32 steps for a signal of length 256. (a) Components, with respect to the cosine basis, of the signal obtained by translating the first vector. (b) Tail of the signal shown in (a) displayed as a log-log plot. The slope of the line is -1, showing that the decay is of order $1/x$. (c) Circular translation matrix with only the diagonal blocks. (d) Eigenvalues of a block of (c), shown with circles, and the corresponding eighth roots of unity, shown with crosses.

technique gives. This is followed by an example in which we design a critical-band filter bank for use in audio coding.

A. Toy Example

Consider a vector space of size 8 as well as the window of size 8.[4] We use the approach described in Section IV-B, obtaining a result that, although not orthogonal, is close to orthogonal. The corresponding basis functions are given in Fig. 1(b), whereas their magnitude frequency responses are depicted in Fig. 1(c). As can be seen from the figures, the technique indeed achieves the desired time–frequency localization. Consider, for example, the basis function $\mathbf{h}_1$. It is mostly localized in the first half of the interval and is of low frequency. On the other hand, basis function $\mathbf{h}_8$ is mostly localized in the second half of the interval and is of the highest frequency. Note how the frequency increases as we increase the number of the basis function.

B. Design of a Critical-Band Basis for Audio Coding

The problem we attack now is more complex and is governed by the needs of an audio coder. In [20], Johnston lists a few of the interactions between the human auditory system and filter banks that need to be addressed when designing a perceptual audio coder: Among them, we have time-domain effects such as pre-echo and stereo imaging effects; frequency-domain artifacts such as unmasked aliasing and large overcoding requirements due to the filter bank bandwidth; and coding gain problems, such as lack of sufficient coding gain as well as loss of coding gain due to pre-echo or nonstationarity issues. Many of these issues have been addressed in current filter banks designs. Many, however, are still left open, and we address a few of them, such as the need to "flexibly change the time–frequency tiling of the filter banks" such that "frequency portions of the signal that are highly varying may require a time resolution matching that of the critical band" [20]. Johnston further states that a filter bank that switches between the critical band and the uniform filter banks has to have a certain signal and frequency meaning. Thus, an allpass filter bank would not do the job. Moreover, an orthonormal filter bank is highly desirable. Finally, the resulting filter bank has to be of sufficiently low complexity so that it can be implemented.

Our technique lends itself well to the above requirements. First, since our scheme is window based, it is easy to switch from the uniform mode to a critical-band mode (since these transitions depend only on the window). This further implies that the transition filter bank keeps its frequency meaning. Due to the fact that our approach is based on (2) and if we design $\mathbf{G}$

[4]Note that time-varying tilings can be obtained since they depend only on the window. For example, we could switch from the tiling as in Fig. 1(a) to a uniform frequency tiling.

so that it has a fast algorithm for implementation, the resulting filter bank is not overly complex. In the rest of this section, we describe how we design a critical-band filter bank (the uniform filter bank is the local cosine basis filter bank or the MDCT).

1) Reducing Complexity by Using Translation: We wish to design a critical-band filter bank that follows (or precedes) a 1024 uniform bank. This means that we might have a situation where we have a 1024-channel cosine modulated filter bank, and at some point, we want to switch to a nonuniform critical-band 25-channel filter bank. Note, however, that this second filter bank is time varying in that a different filter (basis function) might be used in different time slots (which is not the case in the first filter bank). Therefore, the size of our basis is $N = 1024$. This means that we have 1024 basis functions tiling our given space in the time–frequency plane. Such a problem is inherently big, and even the approximate method would require too much computational time. Thus, we use the eigenvalue approach described in the previous section.

Fig. 2 shows the time–frequency tiling used in this case. We have 25 frequency bands.[5] Intuitively, it is clear that each filter in the same frequency band has to be some kind of a translation of the first one. Thus, in order to reduce the computational complexity, the filters in the same frequency band are constrained to be circular translations of one another.

Let us first impose that these filters be linear transformations of the first filter, that is, for band 8, for example

$$\mathbf{g}_{i_8} = \mathbf{L}_{i_8}\mathbf{g}_8 \tag{12}$$

where $\mathbf{L}_{i_8}$ is a linear transformation describing some kind of a translation. This can be used in (5) to obtain a simpler form of the cost function.

Indeed, by rewriting (5) for vectors (12), we get

$$\sum_i \mathbf{g}_{i_8}^T \mathbf{C}_{i_8}\mathbf{g}_{i_8} = \mathbf{g}_8^T \left(\sum_i \mathbf{L}_{i_8}^T \mathbf{C}_{i_8}\mathbf{L}_{i_8}\right)\mathbf{g}_8. \tag{13}$$

Repeating this process for each $\mathbf{g}_i$, cost function (5) can be rewritten as a function of the "primitive" vectors (25 in our case), leading to a simpler problem

$$\sum_{j=1}^{25} \mathbf{g}_j^T \left(\sum_i \mathbf{L}_{i_j}^T \mathbf{C}_{i_j}\mathbf{L}_{i_j}\right)\mathbf{g}_j. \tag{14}$$

Care has to be exercised when imposing linear dependencies (12). Consider, for example, what happens if we circularly translate the filters. Fig. 3(a) shows the impulse response obtained for the eighth filter $\mathbf{g}_8$ in Fig. 2, with the constraint of linear dependence. Note that at the point 1536, the impulse response is discontinuous.[6] Such discontinuities are induced by the circular translation constraint. To understand why, suppose that the impulse response is as depicted in Fig. 3(c). Although the filter in Fig. 3(c) is smoother, this property is lost when the same filter is translated, as shown in Fig. 3(d).

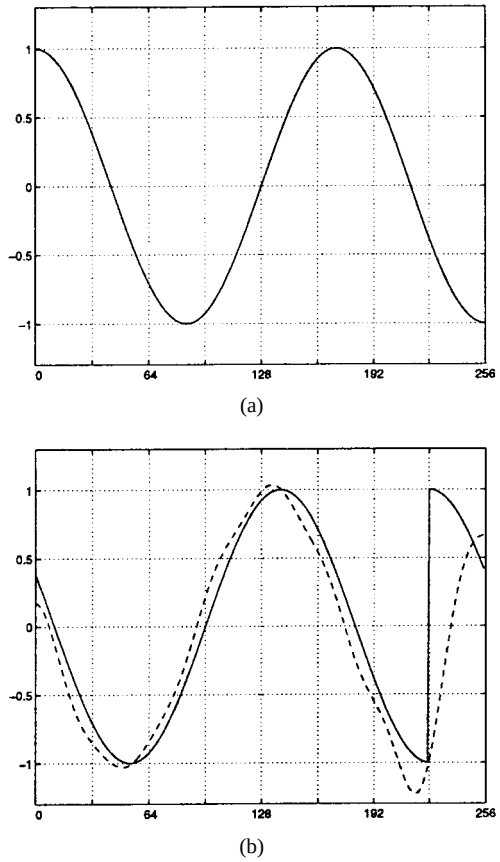


Fig. 6. (a) Test signal for the approximate translation. (b) Effect of the real circular translation (solid line) and of the approximate one (dashed line) on the test signal in (a).

Because of the requirement on frequency localization, the cost function is high when the impulse response is smooth. Since these filters belong to a local orthogonal bank given by (2), the vectors obtained by the minimization of (5) are extended by symmetry and modulated with a window as in Fig. 4. At the right symmetry point, the function is extended by anti-symmetry; the requirement that the function in Fig. 4(c) be smooth implies that the original vector is almost zero in the vicinity of such a point. However, if this is true, when the filter is translated, it becomes discontinuous. By the same token, if the circular translation is smooth, the first filter becomes discontinuous. Therefore, the optimization procedure has to accommodate two incompatible constraints, introducing the discontinuities seen in Fig. 3(a).

A possible solution to this problem is to search for some kind of "pseudo-translations" with the following requirements:

- They are approximate translations.
- They are periodic.
- They do not introduce discontinuities as we saw before.

[5] In the figure, the frequency axis is in barks, where one bark is the width of one critical band. The conversion between barks and the linear frequency scale is given by $B = 13.0\arctan(0.76f/1000) + 3.5\arctan(f/7500)^2$, where B is bark frequency, and f is frequency in Hz.

[6] Using the term "discontinuity" in the discrete-time case is abuse of language. It should be clear from the context that "discontinuity" refers to the fact that we are introducing large differences between samples.

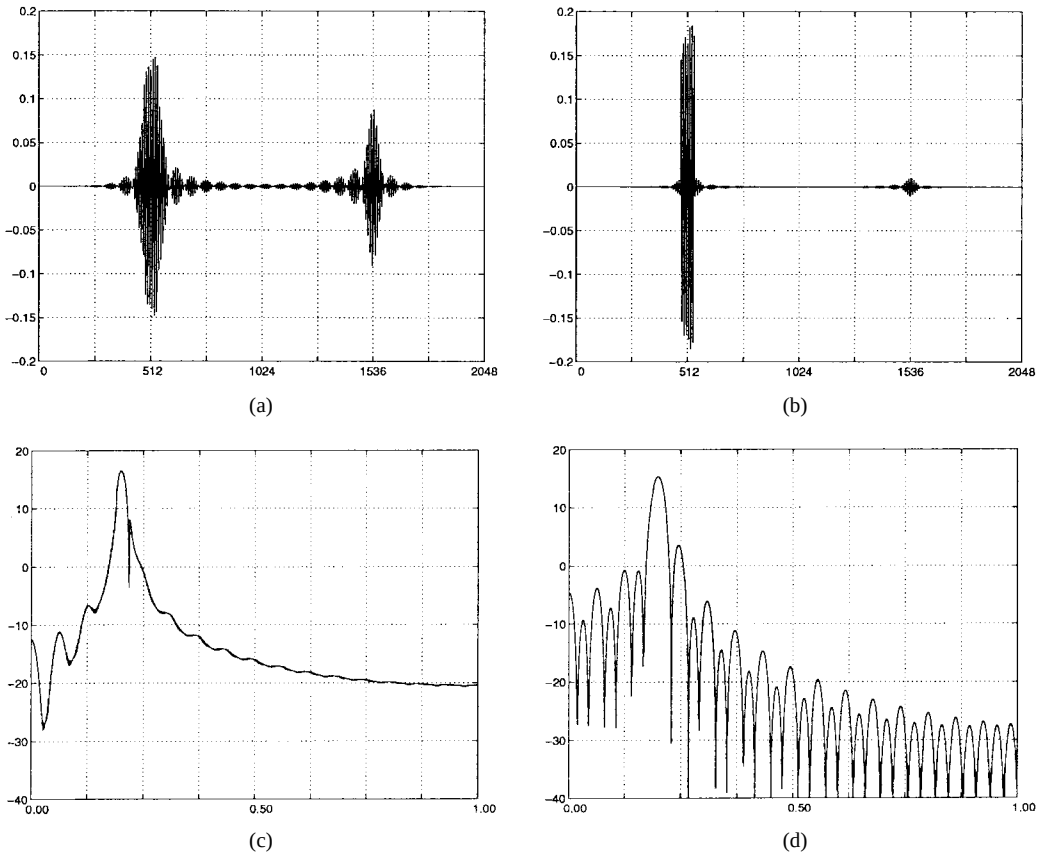


Fig. 7. Effect of different weights for time and frequency errors. (a) Filter corresponding to the 19th band optimally designed with equal weights for both time and frequency and (c) its magnitude frequency response. (b) Filter corresponding to the same band designed by imposing a weight 25 times greater on the time error and (d) its magnitude frequency response.

2) Constructing Pseudo-Translations Let us see now how to find such pseudo-translations. To this end, it is useful to understand how an orthogonal basis is mapped into a filter bank. We start in continuous time to get an idea of the spirit of the reasoning. At a convenient point, we switch to discrete time. Recall Fig. 4(a), and note that if it were a continuous-time example, the continuity of the signal in Fig. 4(c) would descend from the continuity of the signal in Fig. 4(a) as well as from it being zero at $t = 1$.[7]

Therefore, we choose the basis vectors from the set of functions that are continuous on $[0, 1]$ and are zero at 1. As an example, take the starting basis $\mathbf{G}$ from the usual local cosine basis. The set of vectors in $\mathbf{G}$ satisfies these requirements as does any of their finite linear combinations. Consider such cosine vectors $\mathbf{g}_n$ as a basis for $L^2([0,1])$. If we can express a vector $\mathbf{f}$ from $L^2([0,1])$ as

$$\mathbf{f} = \sum_{n=0}^{N_1} a_n \mathbf{g}_n \tag{15}$$

then $\mathbf{f}$ is continuous and zero at $t = 1$. Note that this is a sufficient condition but not necessary. The decay is of importance as well (if the values of coefficients a_n decay fast, we effectively have a finite number of nonzero values). On the other hand, a circular translation of $\mathbf{f}$ by $1/N$ produces

$$\tau_{1/N}\mathbf{f} = \sum_{n,m=0}^{\infty} t_{m,n} a_n \mathbf{g}_m \tag{16}$$

that is, even if the expansion of $\mathbf{f}$ is finite, the expansion of $\tau_{1/N}\mathbf{f}$ is not. Expressing the operator $\tau_{1/N}$ in matrix form, it is clear that for $\tau_{1/N}\mathbf{f}$ to have a finite expansion, the matrix $\tau_{1/N}$ has to have a finite number of nonzero columns.

Let us now go back to discrete time since we can start talking about finite matrices. An idea is to truncate the rows in the matrix, keeping only the columns with higher values, resulting in a diagonal block matrix (in continuous time this would ensure that each row would have a finite number of nonzero values). However, the drawback is that the obtained operator may not be of period N (that is, by applying it N times, we do not obtain an identity). To overcome this problem, note that our operator has period N if and only if all of its eigenvalues are Nth roots of unity. Moreover, a diagonal block operator has period N if and only if each block has period N. This suggests the following algorithm.

Algorithm 1:

1) Express the operator in the chosen basis as a matrix.
2) Truncate the matrix by keeping only the blocks $\mathbf{B}$ on the diagonal.
3) For each block $\mathbf{B}$, do the following.
   3.1) Diagonalize it as $\mathbf{B} = \mathbf{A}^{-1}\mathbf{D}\mathbf{A}$.

[7] Note that the time scale in Fig. 4 is discrete, and therefore, 0 and 1 would correspond to 128 and 384 on the horizontal axis.

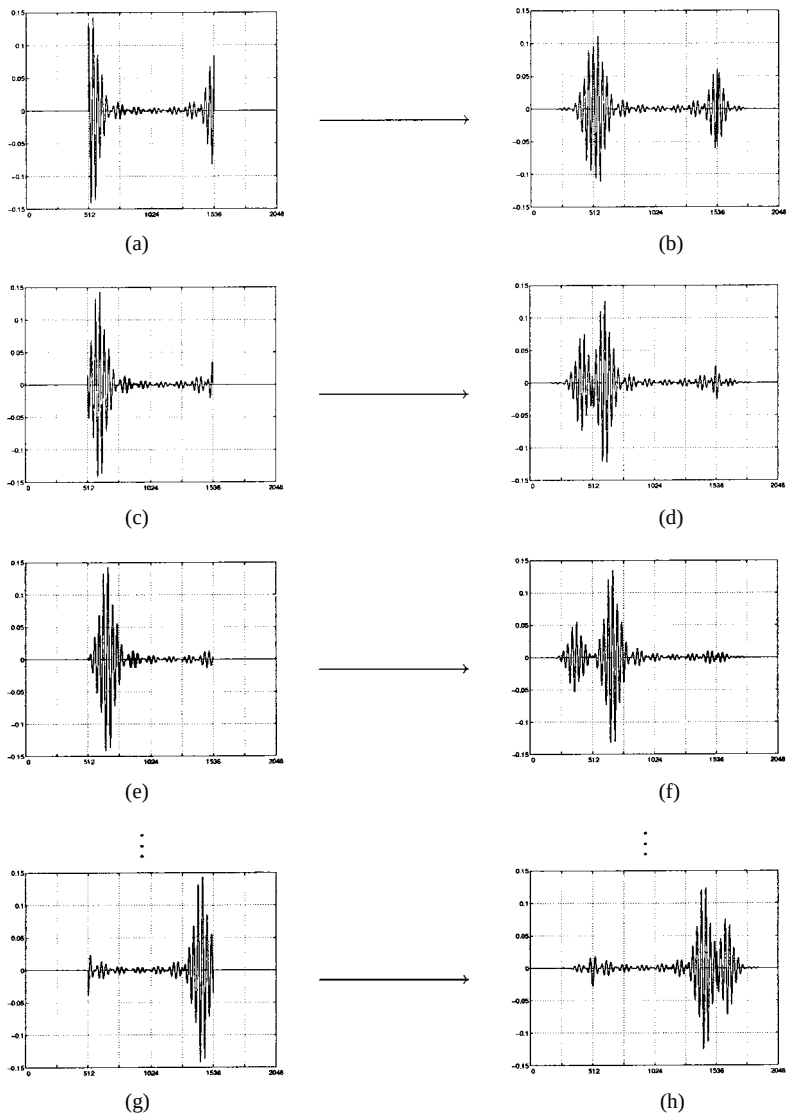


Fig. 8. (a) Vector of $\mathbf{G}$ corresponding to the 12th band and (c), (e), (g) some translations of (a). (b), (d), (f), (h) Filters of $\mathbf{H} = \mathbf{WKG}$ corresponding to the vectors in (a), (c), (e) and (g). Note how the "burst" in the vector generates pre- and post-echo in the resulting filter.

3.2) Approximate each diagonal element of $\mathbf{D}$ by a matrix $\hat{\mathbf{D}}$ whose eigenvalues are all Nth roots of unity.

3.3) Reconstruct $\hat{\mathbf{B}}$ as $\hat{\mathbf{B}} \triangleq \mathbf{A}^{-1}\hat{\mathbf{D}}\mathbf{A}$.

As an example, suppose we have a signal of length 256 and a circular translation of period 8. According to Algorithm 1, express the matrix corresponding to such a circular translation with respect to the cosine basis. The signal corresponding to the first column is depicted in Fig. 5(a). The slow decay of such a signal is more evident in Fig. 5(b), where it is displayed in a log-log plot. The slope of the envelope being -1 says that the signal decays as $1/x$.

Divide the matrix into blocks of size 16×16, and force the blocks outside the main diagonal to zero. Fig. 5(c) shows a contour plot of the new matrix. The eigenvalues of such a matrix are not exactly eighth roots of unity, that is, the matrix does not have period 8. Therefore, map each eigenvalue into its closest root. In Fig. 5(d), the original eigenvalues of one of the blocks are denoted by circles, while the eighth roots of unity are denoted by crosses.

Finally, reconstruct the matrix to obtain the new translation. Fig. 6(a) displays a test signal (a cosine), whereas Fig. 6(b) demonstrates the effect of both the true translation (solid line) and the approximate one (dashed line). As can be seen from the figure, the approximate translation results in a continuous signal while retaining the spirit of the true circular translation.

Going back to the example in Fig. 3, (e) shows a filter obtained using an approximate translation. Note that the discontinuities present in Fig. 3(a) are smoothed in Fig. 3(e). This is because the new operator maps continuous functions into continuous functions.

By changing the block partition, we can obtain several types of approximations. As a rule of thumb, small blocks give rise to smoother filters but lose, in some sense, the idea of a translation; on the other hand, big blocks give rise to operators closer to the true translation, but the resulting filter is less smooth.

3) Other Design Considerations For the highest frequency filters, time localization is far more important than frequency response. In order to achieve good time localization, we weight

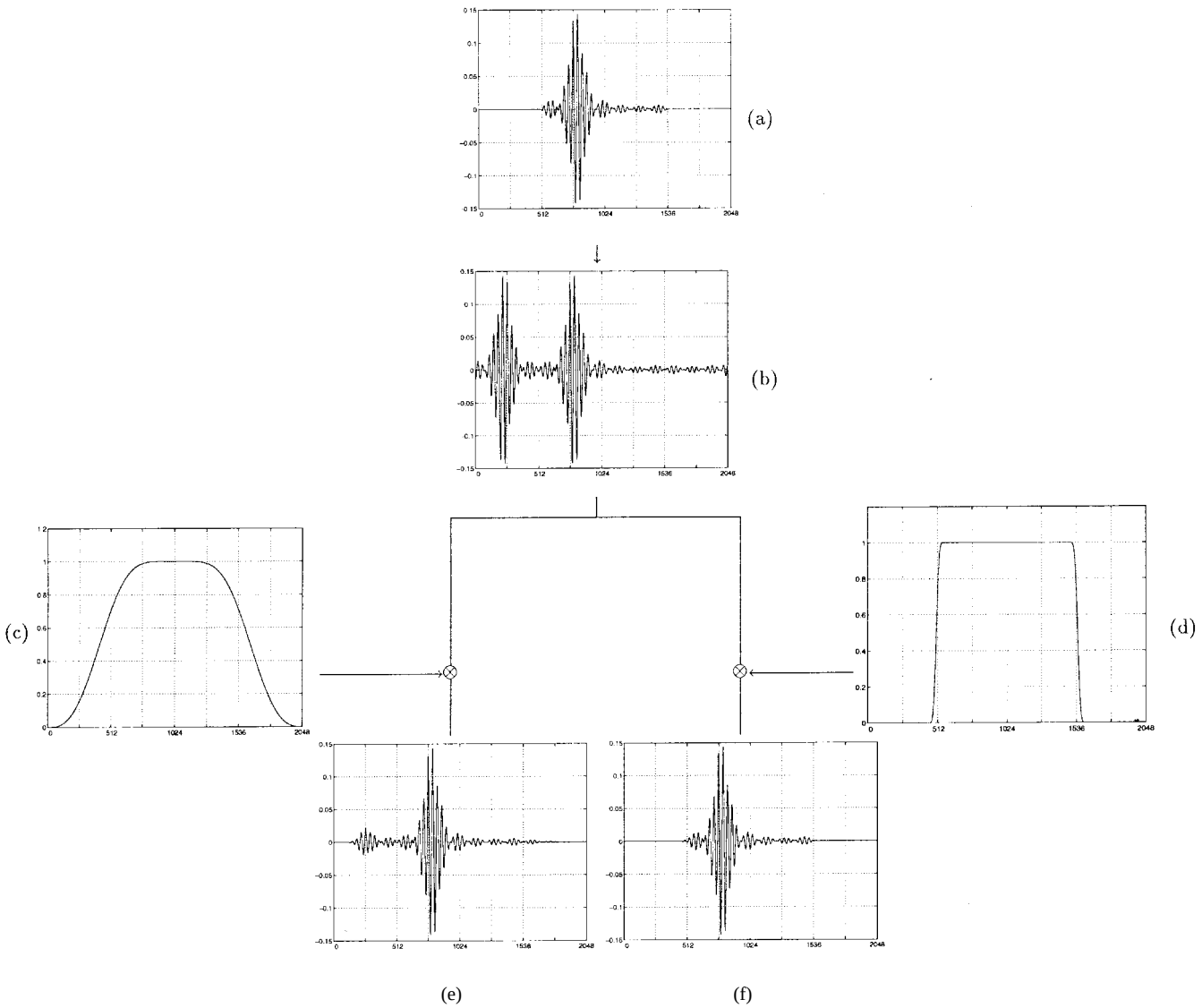


Fig. 9. Example of pre-echo. (a) Vector corresponding to the 12th band, translated by 256. (b) Vector (a) extended by symmetry around the two end points. (c) Window w_1. (d) Sharper window w_2. (e) Filter obtained by the product of (b) and (c). Note how the first (reflected) burst has been attenuated. However, this is not enough, leading to the use of the sharper window w_2 resulting in (f). Here, the pre-echo is almost nonexistent.

the frequency and the time errors differently. Fig. 7(a) shows the filter corresponding to the 19th band of Fig. 2 obtained by weighting equally time and frequency errors, whereas Fig. 7(b) shows the filter corresponding to the same band but obtained by weighting the time error 25 times more than the frequency one. Fig. 7(c) and (d) show respective magnitude frequency responses. However, even though the filter in Fig. 7(b) is better localized in time, it still suffers from the following.

The first problem is the appearance of the pre-echo effects due to the filter symmetry. Observe Fig. 8. In Fig. 8(a), the vector from $\mathbf{G}$ corresponding to the 12th band is shown, whereas in Fig. 8(b), the corresponding filter is shown. Such a vector looks like a burst because of its concentration in time. In Fig. 8(c), (e) and (g), we can see some of the circular[8] translations of the original vector, whereas the corresponding filters are displayed in Figs. 8(d), (f) and (h), respectively. Note that when translating the original vector, filters obtain a second burst that is a reflection of the original one. The second burst is particularly annoying in audio coding as it gives rise to pre-echo artifacts, that is, the listener perceives a copy of the signal before its due time.

Unfortunately, the pre-echo artifact cannot be avoided by modifying the basis because it depends on the symmetries imposed on the filter (theoretically necessary in order to achieve the orthogonality) and on the window shape. This is easily seen from Fig. 9. In Fig. 9(a), we can see a translation of the basis vector in Fig. 8(a). In Fig. 9(b), the same vector has been extended by symmetry and antisymmetry. Note how the burst that carries the vector energy reflects itself in the left tail. When the signal in Fig. 9(b) is multiplied by the window in Fig. 9(c), the reflected burst is attenuated but not enough [see Fig. 9(e)]. However, when the signal in Fig. 9(b) is multiplied by the sharper window in Fig. 9(d), the reflected

[8] Note that for the highest frequency filters, circular translations do not pose a problem since they are very short.

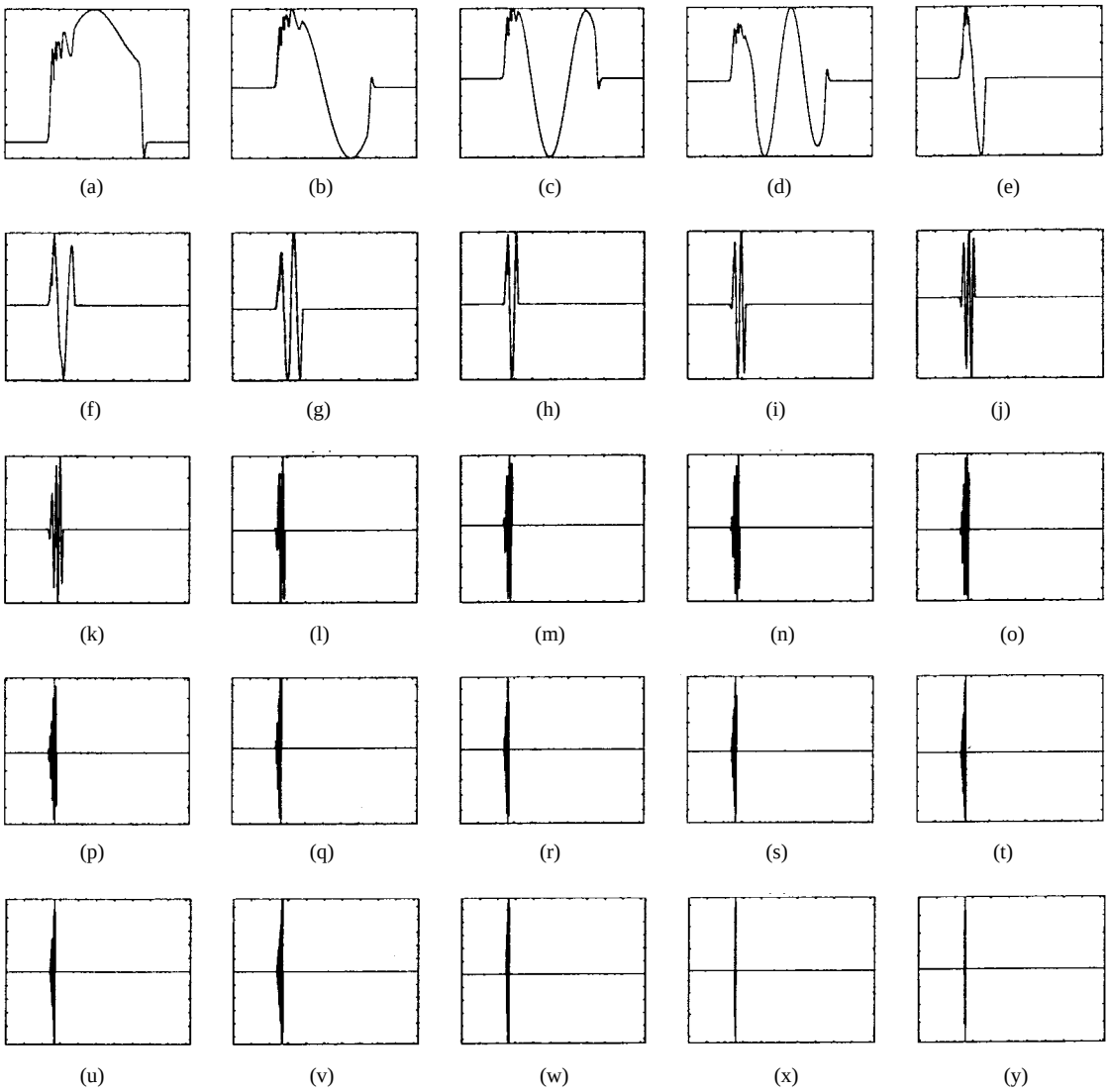


Fig. 10. Filters forming the critical-band filter banks (25 of them). The rest of the basis functions (1024–25) are obtained by pseudo-translations and circular translations (in the case of higher frequency filters) of the 25 critical-band filters. To solve the problem of pre-echo, a very sharp window was used. The filters were then orthogonalized and truncated to allow for the desired time localization. As a result, they lost their orthogonality.

burst is almost nonexistent [see Fig. 9(f)]. Therefore, to solve the pre-echo problem, choose a sharper window.

The second problem is that the vectors obtained are not necessarily orthogonal. To overcome this, the basis is orthogonalized via the singular value decomposition, resulting in the orthonormal basis having a minimum distance from the original one.

Finally, even if weighting the cost function allows us to obtain filters better localized in time, such localization is not enough for audio applications [21]. To improve time localization, the resulting vectors are forced to be zero outside the desired interval for the filter to have the desired support. The result of such an operation can be seen in Fig. 10, where all of the 25 filters are displayed in a 5×5 matrix. Observe that they are exactly zero outside the desired intervals.

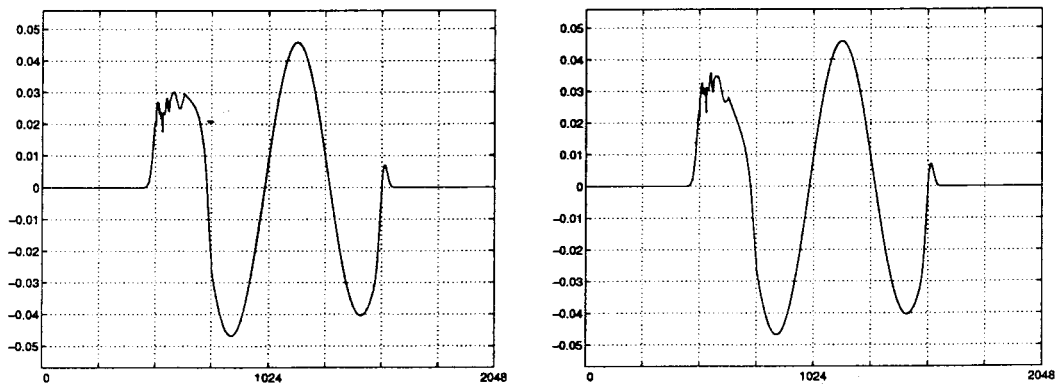
Unfortunately, because of the truncation procedure, the filters in Fig. 10 are not orthogonal. To regain some orthogonality, the filters in Fig. 10 have been modified "by hand." The result for the filter in Fig. 10(d) is shown in Fig. 11. Note that the filter has some artifacts at the beginning of its impulse response due to the linear combinations taken with the higher frequency filters in order to achieve the desired orthogonality.

Appendix
Orthonormal Basis Closest to the Starting Nonorthogonal One

Property 1: Let $\mathbf{A} = \mathbf{OSQ}$ be the singular value decomposition of a nonsingular square matrix $\mathbf{A}$, with $\mathbf{S} = \text{diag}(s_1, s_2, \ldots, s_N),\ s_i > 0$.

Matrix $\mathbf{OQ}$ is the orthogonal matrix having minimum Frobenius distance from matrix $\mathbf{A}$, where the Frobenius distance between two matrices $\mathbf{X}$, $\mathbf{Y}$ is defined as

$$||\mathbf{X}-\mathbf{Y}||_F \triangleq \sum_{i,j} |\mathbf{X}_{i,j}-\mathbf{Y}_{i,j}|^2 = \text{Tr}((\mathbf{X}-\mathbf{Y})^T(\mathbf{X}-\mathbf{Y})). \tag{17}$$



(a) (b)

Fig. 11. In order to regain some of the orthogonality lost in the previous figure, some of the filters were orthogonalized by hand. Since they are very close to those in the same figure, we show a pair where the difference is the most striking. (a) Filter (d) from the previous figure. (b) The same filter orthogonalized by hand.

Proof: Let $\mathbf{V}$ be any orthogonal matrix of the same size as $\mathbf{A}$. Write the distance between $\mathbf{V}$ and $\mathbf{A}$ as

$$\begin{aligned}\|\mathbf{A}-\mathbf{V}\|_F &= \mathrm{Tr}((\mathbf{OSQ}-\mathbf{V})^T(\mathbf{OSQ}-\mathbf{V})) \\ &= \mathrm{Tr}((\mathbf{Q}^T\mathbf{SO}^T-\mathbf{V}^T)(\mathbf{OSQ}-\mathbf{V})) \\ &= \mathrm{Tr}(\mathbf{Q}^T\mathbf{S}^2\mathbf{Q})+\mathrm{Tr}(\mathbf{I})-2\,\mathrm{Tr}(\mathbf{Q}^T\mathbf{SO}^T\mathbf{V}).\end{aligned} \tag{18}$$

The first two terms in (18) do not depend on $\mathbf{V}$; therefore, to minimize (18), we have to maximize the last term $\mathrm{Tr}(\mathbf{Q}^T\mathbf{SO}^T\mathbf{V})$.

Since $\mathbf{Q}$ is orthogonal, $\mathbf{Q}^T\mathbf{SO}^T\mathbf{V}$ is similar to $\mathbf{Q}(\mathbf{Q}^T\mathbf{SO}^T\mathbf{V})\mathbf{Q}^T=\mathbf{SO}^T\mathbf{VQ}^T$; therefore, the two matrices have the same trace, that is

$$\mathrm{Tr}(\mathbf{Q}^T\mathbf{SO}^T\mathbf{V})=\mathrm{Tr}(\mathbf{S}[\mathbf{O}^T\mathbf{VQ}^T]). \tag{19}$$

Moreover, the matrix inside square brackets in (19) is orthogonal as it is the product of orthogonal matrices. We want to prove that if $\mathbf{U}$ is an orthogonal matrix, then $\mathrm{Tr}(\mathbf{SU})$ is maximum when $\mathbf{U}=\mathbf{I}$. Since in (19) $\mathbf{U}=\mathbf{O}^T\mathbf{VQ}^T$, $\mathbf{U}=\mathbf{I}$ implies $\mathbf{V}=\mathbf{OQ}$, which is what we want to prove.

The trace of $\mathbf{SU}$ can be expressed as

$$\mathrm{Tr}(\mathbf{SU})=\sum_i \mathbf{S}_i\mathbf{U}_{i,i}. \tag{20}$$

Since matrix $\mathbf{U}$ is orthogonal, every column is of unit norm, implying that $|\mathbf{U}_{i,i}|\le 1$. Since every $\mathbf{S}_i$ in (20) is greater than zero, then (20) is maximum when $\mathbf{U}_{i,i}=1$ for each i, yielding $\mathbf{U}=\mathbf{I}$.

ACKNOWLEDGMENT

The authorswould like to thank D. Sinha of Bell Laboratories and J. Johnston of AT&T Laboratories for fruitful discussions that lead to the critical-band filter design. Our gratitude goes to J. Mazo for his useful comments.

REFERENCES

[1] Y. Meyer, *Wavelets: Algorithms and Applications*. Philadelphia, PA: SIAM, Translated and revised by R. D. Ryan, 1993.
[2] R. Coifman, Y. Meyer, S. Quake, and M. Wickerhauser, "Signal processing and compression with wavelet packets," Tech. Rep., Yale Univ., New Haven, CT, 1991.
[3] R. Coifman and Y. Meyer, "Remarques sur l'analyze de Fourier à fenêtre," *C. R. Acad. Sci. Paris*, vol. I, pp. 259–261, 1991.
[4] P. Auscher, G. Weiss, and M. Wickerhauser, "Local sine and cosine bases of Coifman and Meyer and the construction of smooth wavelets," in *Wavelets: A Tutorial in Theory and Applications*, C. K. Chui, Ed. San Diego, CA: Academic, 1992.
[5] B. Jawerth and W. Sweldens, "Biorthogonal smooth local trigonometric bases," *J. Fourier Anal. Appl.*, vol. 2, 1996.
[6] S. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries," *IEEE Trans. Signal Processing*, vol. 41, pp. 3397–3415, Dec. 1993.
[7] C. Thiele and L. F. Villemoes, "A fast algorithm for adapted time-frequency tilings," Yale Univ., New Haven, CT, preprint, 1994.
[8] H. Malvar, *Signal Processing with Lapped Transforms*, Norwood, MA: Artech House, 1992.
[9] J. Princen, A. Johnson, and A. Bradley, "Subband transform coding using filter bank designs based on time domain aliasing cancellation," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, Dallas, TX, Apr. 1987, pp. 2161–2164.
[10] T. A. Ramstad and J. Tanem, "Cosine-modulated analysis-synthesis filterbank with critical sampling and perfect reconstruction," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Toronto, Ont., Canada, May 1991, pp. 1789–1792.
[11] R. D. Koilpillai and P. P. Vaidyanathan, "New results on cosine-modulated FIR filter banks satisfying perfect reconstruction," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Toronto, Ont., Canada, May 1991, pp. 1793–1796.
[12] C. Herley, J. Kovačević, K. Ramchandran, and M. Vetterli, "Tilings of the time-frequency plane: Construction of arbitrary orthogonal bases and fast tiling algorithms," *IEEE Trans. Signal Processing*, Special Issue on Wavelets and Signal Processing, vol. 41, pp. 3341–3359, Dec. 1993.
[13] J. D. Johnston and A. J. Ferreira, "Sum-difference stereo transform coding," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, San Francisco, CA, Mar. 1992, pp. II:569–572.
[14] C. Todd, G. Davidson, M. Davis, L. Fielder, B. Link, and S. Vernon, "AC-3: Flexible preceptual coding for audio transmission and storage," in *Proc. Conv. AES*, Amsterdam, The Netherlands, Feb. 1994.
[15] J. Kovačević, D. LeGall, and M. Vetterli, "Image coding with windowed modulated filter banks," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Glasgow, U.K., May 1989, pp. 1949–1952.
[16] A. Johnson, J. Princen, and M. Chan, "Frequency scalable video coding using the MDCT," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Adelaide, Australia, 1994.
[17] R. Bernardini and J. Kovačević, "Local orthogonal bases I: Construction," *Multidim. Syst. Signal Process.*, Special Issue on *Wavelets and Multiresolution Signal Processing*, vol. 7, pp. 331–370, July 1996.

Invited paper. Reprinted in *Multidimensional Filter Banks and Wavelets*, S. Basu and B. Levy, Eds. Boston, MA: Kluwer, 1997.
[18] R. Bernardini and J. Kovačević, "Local orthogonal bases II: Window design," *Multidim. Syst. Signal Process.*, Special Issue on *Wavelets and Multiresolution Signal Processing*, vol. 7, pp. 371–400, July 1996. Invited paper. Reprinted in *Multidimensional Filter Banks and Wavelets*, S. Basu and B. Levy, Eds. Boston, MA: Kluwer, 1997.
[19] M. Vetterli and J. Kovačević, *Wavelets and Subband Coding*. Englewood Cliffs, NJ: Prentice-Hall, 1995.
[20] J. D. Johnston, "Subband and wavelet transforms: designs and applications," in *Filterbank Interactions with Audio Coding*. Boston, MA: Kluwer, 1995.
[21] D. Sinha and J. Johnston, private communication, 1995.

Riccardo Bernardini was born in Genova, Italy, in 1964. He received the "Laurea in Ingegneria Elettronica" degree from the University of Padova, Padova, Italy, in 1990.

Since then, he has been with the Dipartimento di Elettronica e Informatica, University of Padova, with a scholarship from the Consorzio Padova Ricerche, and from November 1992 to November 1995, he was a Ph.D. student. He spent the last year of his Ph.D. studies at the former AT&T Bell Laboratories, Murray Hill, NJ. From April 1996 to April 1997, he was with the Swiss Federal Technological Institute, Lausanne, as a Postdoctoral Fellow. Now, he is working as "assegnista" with the Dipartimento di Elettronica e Informatica, University of Padova. His main interests are in the area of multidimensional signal processing, wavelets, and filter banks.

Jelena Kovačević (SM'96) was born in Yugoslavia in 1962. She received the Dipl. Electr. Eng. degree from the Electrical Engineering Department, University of Belgrade, Belgrade, Yugoslavia, in 1986, and the M.S. and Ph.D. degrees from Columbia University, New York, NY, in 1988 and 1991, respectively.

In November 1991, she joined AT&T Bell Laboratories, Murray Hill, NJ, as a Member of Technical Staff. In the Fall of 1986, she was a Teaching Assistant at the University of Belgrade. From 1987 to 1991, she was a Graduate Research Assistant at Columbia University. In the summer of 1985, she worked for Gaz de France, Paris, France, during the summer of 1987 for INTELSAT, Washington, D.C., and in the summer of 1988 for Pacific Bell, San Ramon, CA. Her research interests include multirate signal processing, wavelets, image and video coding, and communications.

Dr. Kovačević received the E. I. Jury Award from Columbia University for outstanding achievement as a graduate student in the areas of systems, communication, or signal processing. She is the co-author of the book (with M. Vetterli) *Wavelets and Subband Coding* (Englewood Cliffs, NJ: Prentice Hall, 1995). She served as an Associate Editor of the IEEE TRANSACTIONS ON SIGNAL PROCESSING and as the Guest Co-Editor (with I. Daubechies) of the Special Issue on Wavelets of the PROCEEDINGS OF THE IEEE. She is on the Editorial Boards of the *Journal of Applied and Computational Harmonic Analysis* and the *Journal of Fourier Analysis and Applications*. She is on the IMDSP Technical Committee of the IEEE Signal Processing Society and was the General Co-Chair (with J. Allebach) of the Ninth Workshop on Image and Multidimensional Signal Processing.