The GenLOT: Generalized Linear-Phase Lapped Orthogonal Transform

Ricardo L. de Queiroz, Troung Q. Nguyen, *Member, IEEE*, and K. R. Rao, *Senior Member, IEEE*

***Abstract*— The general factorization of a linear-phase paraunitary filter bank (LPPUFB) is revisited. From this new perspective, a class of lapped orthogonal transforms with extended overlap (generalized linear-phase lapped orthogonal transforms (GenLOT's)) is developed as a subclass of the general class of LPPUFB. In this formulation, the discrete cosine transform (DCT) is the order-1 GenLOT, the lapped orthogonal transform is the order-2 GenLOT, and so on, for any filter length that is an integer multiple of the block size. The GenLOT's are based on the DCT and have fast implementation algorithms. The implementation of GenLOT's is explained, including the method to process finite-length signals. The degrees of freedom in the design of GenLOT's are described, and design examples are presented along with image compression tests.**

I. Introduction

The discrete cosine transform (DCT) [1] is used in most of the international standards for image compression and for several signal processing tasks. The signal is generally segmented in blocks of M samples, and each block is transformed and processed in the DCT domain. This segmentation process sometimes leads to discontinuities across the block boundaries after the processed signal is inverse transformed [1]. The lapped orthogonal transform (LOT) [2]–[4] was developed as a competitive alternative because of its extended basis functions, which overlap across traditional block boundaries, thus eliminating the blocking effect. One of the reasons for the growing popularity of the LOT is the fact that it possesses a fast implementation algorithm and good performance. In addition, its algorithm is based on the DCT, which is highly popular in image coding and for which a large number of fast algorithms, VLSI chips, and computer programs have been developed [1].

It is well known that the DCT and the LOT are particular choices of finite impulse response (FIR) linear-phase paraunitary filter banks (LPPUFB's) [5], [10]. Linear-phase filter banks have been studied extensively, and several design approaches can be found in the literature [5]–[9]. However, fast implementation algorithms were usually ignored. Very recently, a minimal structure to implement all LPPUFB's (where the filters' lengths are the same) was developed in [11] and [12]. We will introduce a particular simplification leading to a class of LPPUFB that we call the generalized linear-phase lapped orthogonal transforms (GenLOT's). The GenLOT's have a fast implementation algorithm based on the DCT. Both DCT and LOT can be regarded as particular GenLOT's.

In terms of notation, the following conventions are adopted:

- $\mathbf{I}_n$: $n \times n$ identity matrix
- $\mathbf{J}_n$: $n \times n$ counter-identity or exchange matrix, defined as

$$\mathbf{J}_n = \begin{bmatrix} 00 & \cdots & 01 \\ 00 & \cdots & 10 \\ \vdots & & \vdots \\ 01 & \cdots & 00 \\ 10 & \cdots & 00 \end{bmatrix}$$

- $\mathbf{0}_n$: $n \times n$ null matrix
- $(\ )^T$: transposition of matrices and vectors
- $(\ )^R$: column and row reversal of a matrix. For example, if $\mathbf{A}$ is a $n \times n$ matrix, then $\mathbf{A}^R = \mathbf{J}_n \mathbf{A} \mathbf{J}_n$.

The presentation assumes 1-D signal processing. When applying GenLOT's in image coding, 2-D processing by separable implementation is assumed [1], [5].

The paper is organized as follows. In Section II, the LOT is briefly reviewed, the generic parameterization of a LPPUFB [11], [12] is revisited and simplified, and the GenLOT is introduced. Section III presents the GenLOT implementation algorithm for finite-length signals, whereas Section IV presents some design methods and examples. Finally, the conclusions of this paper are presented in Section V.

II. GenLOT and the General LPPUFB

A. LOT

The DCT is implemented by segmenting the input signal into blocks of M samples and transforming each one independently. The LOT allows overlap of the basis functions, as shown in Fig. 1(a). In this figure, two different segmentation strategies of the input signal into blocks of M samples are shown, where they differ only in the displacement of the blocks. The top one refers to the block segmentation used for the DCT; thus, the basis functions of the LOT would be imposed over the position of the basis functions but overlapping $M/2$ samples on each side over adjacent blocks. However, in terms of implementation, the block positions used for the DCT are of little importance. The appropriate block segmentation for the LOT is at the bottom of Fig. 1(a). As

Manuscript received January 6, 1994; revised September 13, 1995. The associate editor coordinating the review of this paper and approving it for publication was Prof. Roberto H. Bamberger.

R. L. de Queiroz is with Xerox Corp., Webster NY, USA (e-mail: queiroz@wrc.xerox.com).

T. Q. Nguyen is with the University of Wisconsin—Madison, Madison, WI 53706 USA (e-mail: nguyen@ece.wisc.edu).

K. R. Rao is with the the University of Texas at Arlington, Arlington, TX, USA (e-mail: rao@ee.uta.edu).

Publisher Item Identifier S 1053-587X(96)02391-2.

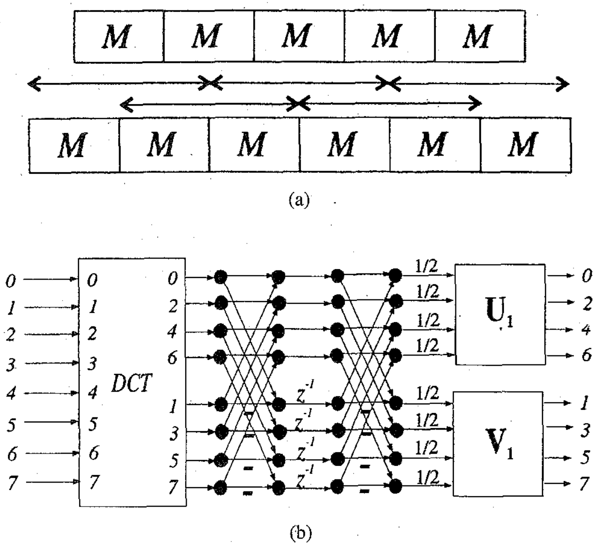


Fig. 1. LOT: (a) Basis functions of the LOT overlap across block boundaries so that each basis function has the length of two blocks of length M; (b) flowgraph for implementation of the LOT for blocks of $M = 8$ samples. The ordering of input-output coefficients for both DCT and LOT are indicated.

an example, for $M = 8$, the implementation algorithm for the LOT is shown in Fig. 1(b). In Fig. 1(b), we can see that the LOT is implemented by postprocessing the output of the DCT. Therefore, it is clear that the block segmentation for the DCT alone and for the DCT stage of the LOT are separated by a delay of $M/2$ samples.

The resulting transform matrix for the LOT, assuming blocks of M samples, is nonsquare and is given by

$$\mathbf{P}_{\mathrm{LOT}} = \begin{bmatrix} \mathbf{U}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{V}_1 \end{bmatrix} \begin{bmatrix} \mathbf{D}_e - \mathbf{D}_o & \mathbf{J}_{M/2}(\mathbf{D}_e - \mathbf{D}_o) \\ \mathbf{D}_e - \mathbf{D}_o & -\mathbf{J}_{M/2}(\mathbf{D}_e - \mathbf{D}_o) \end{bmatrix} \quad (1)$$

where $\mathbf{D}_e$ is the $M/2 \times M$ matrix with the even-symmetric basis functions of the DCT, and $\mathbf{D}_o$ is the matrix with the odd-symmetric ones. $\mathbf{U}_1$ and $\mathbf{V}_1$ are $M/2 \times M/2$ orthogonal matrices. The design suggested for the LOT [2], [10] uses $\mathbf{U}_1 = \mathbf{I}_{M/2}$ and approximates $\mathbf{V}_1$ by $M/2 - 1$ plane rotations [2], [10].

It is well-known [10] that the LOT is an M-channel uniform FIR filter bank, where the filters have length $L = 2M$, and their coefficients are formed by the coefficients of the basis functions. Hence, as the basis functions are symmetric, the LOT can be regarded as a linear-phase filter bank. It is also easy to show that the corresponding filter bank is also paraunitary so that the LOT is a particular LPPUFB [10].

B. Linear-Phase Paraunitary Filter Bank

Consider the uniform maximally decimated M-channel FIR filter bank described in Fig. 2, for which we impose some restrictions. First, we assume that M, which is the number of channels, is even and that the filters have linear phase. Second, we assume the filters have length L, which is an integer multiple of M as $L = NM$. Third, the filter bank is assumed to be paraunitary. Hence [5], [10], we have $g_i(n) = f_i(L-1-n)$, for $0 \le i \le M-1$ and $0 \le n \le L-1$. In addition, from [11], [12], we know that $M/2$ filters (in analysis or synthesis)

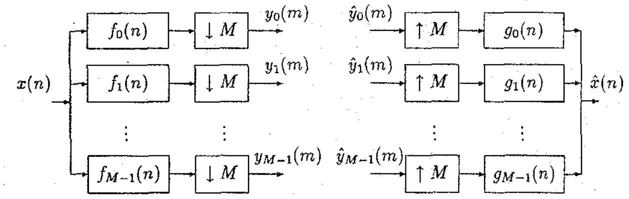


Fig. 2. Critically decimated uniform filter bank. Analysis (left) and synthesis (right) sections are shown.

have symmetric impulse responses, and the other $M/2$ filters have antisymmetric impulse responses.

$$g_i(n) = f_i(L-1-n) = \pm f_i(n). \quad (2)$$

Alternatively, we can develope the filter bank by segmenting the signal into blocks of M samples. For this, let the input signal $x(n)$ be expressed by its M polyphase components $x_i(m)$ [5], [13], as

$$x_i(m) = x(mM + i) \quad (3)$$

where $0 \le i \le M-1$. For a given instant m, the M polyphase samples form the mth block of M samples. The subband signals $y_k(m)$ in Fig. 2 are directly related to the polyphase components by a multi-input multi-output discrete transfer matrix with FIR filter entries [5] known as the polyphase transfer matrix (PTM), as shown in Fig. 3. In this figure, in the analysis section, the input is segmented into blocks of M samples and processed by a PTM $\mathbf{E}(z)$. In the synthesis section, for perfect reconstruction (PR) causal systems using uniform FIR paraunitary filter banks, the subbands are processed by the PTM $\tilde{\mathbf{E}}(z) = z^{-(N-1)}\mathbf{E}^T(z^{-1})$. The blocks are put back into serial form, reconstructing the signal sequence. The devices to segment the signal into blocks and its counterpart to reconstruct the signal, in Fig. 3, are called *blocking* and *unblocking* devices, respectively.

Under the assumptions on L, M, and on the filters symmetry, we know that [11], [12] $\mathbf{E}(z)$ for the LPPUFB of degree $N-1$ can be decomposed as a product of orthogonal factors and delays as

$$\mathbf{E}(z) = \mathbf{S}\mathbf{Q}\mathbf{T}_{N-1}\Lambda(z)\mathbf{T}_{N-2}\Lambda(z)\cdots\Lambda(z)\mathbf{T}_0\mathbf{Q} \quad (4)$$

where

$$\mathbf{Q} = \begin{bmatrix} \mathbf{I}_{M/2} & \mathbf{0}_{M/2} \\ \mathbf{0}_{M/2} & \mathbf{J}_{M/2} \end{bmatrix} \quad (5)$$

$$\Lambda(z) = \begin{bmatrix} \mathbf{I}_{M/2} & \mathbf{0}_{M/2} \\ \mathbf{0}_{M/2} & z^{-1}\mathbf{I}_{M/2} \end{bmatrix} \quad (6)$$

$$\mathbf{S} = \frac{1}{\sqrt{2}} \begin{bmatrix} \mathbf{S}_0 & \mathbf{0}_{M/2} \\ \mathbf{0}_{M/2} & \mathbf{S}_1 \end{bmatrix} \begin{bmatrix} \mathbf{I}_{M/2} & \mathbf{J}_{M/2} \\ \mathbf{I}_{M/2} & -\mathbf{J}_{M/2} \end{bmatrix} \quad (7)$$

where $\mathbf{S}_0$ and $\mathbf{S}_1$ can be any $M/2 \times M/2$ orthogonal matrices, and $\mathbf{T}_i$ are $M \times M$ orthogonal matrices described as

$$\mathbf{T}_i = \begin{bmatrix} \mathbf{A}_i & \mathbf{B}_i \\ \mathbf{B}_i & \mathbf{A}_i \end{bmatrix}. \quad (8)$$

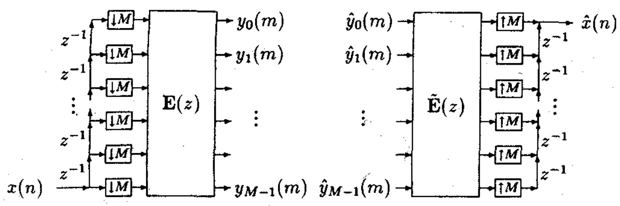


Fig. 3. Filter bank as a transfer matrix applied to the polyphase components of the signal. The matrix $\mathbf{E}(z)$ is called a polyphase transfer matrix and, for paraunitary filter banks, it is a paraunitary matrix, i.e., its inverse is $\mathbf{E}^T(z^{-1})$. For a PR causal system, we must choose $\tilde{\mathbf{E}}(z) = z^{-(N-1)}\mathbf{E}^T(z^{-1})$.

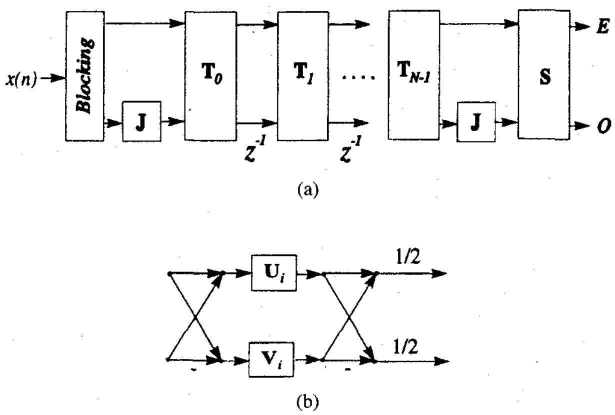


Fig. 4. (a) Flowgraph for the implementation of the PTM $\mathbf{E}(z)$ describing the analysis section of the LPPUFB. Each branch carries $M/2$ samples, and e and o stand for even and odd output subband coefficients. In this factorization, the stages $\mathbf{T}_i$ can be factorized as in part (b).

We will abbreviate the notation for (4) as

$$\mathbf{E}(z) = \mathbf{SQT}_{N-1}\left(\prod_{i=N-2}^{0} \Lambda(z)\mathbf{T}_i\right)\mathbf{Q}. \tag{9}$$

Let

$$\mathbf{W} = \frac{1}{\sqrt{2}}\begin{bmatrix} \mathbf{I}_{M/2} & \mathbf{I}_{M/2} \\ \mathbf{I}_{M/2} & -\mathbf{I}_{M/2} \end{bmatrix}$$

and

$$\Phi_i = \begin{bmatrix} \mathbf{U}_i & \mathbf{0}_{M/2} \\ \mathbf{0}_{M/2} & \mathbf{V}_i \end{bmatrix} \tag{10}$$

where $\mathbf{U}_i$ and $\mathbf{V}_i$ can be any $M/2 \times M/2$ orthogonal matrices. The implementation flowgraph of the LPPUFB is shown in Fig. 4. Note that $\mathbf{T}_i$ can be expressed as [11], [12]

$$\mathbf{T}_i = \mathbf{W}\Phi_i\mathbf{W} \tag{11}$$

for $\mathbf{A}_i = (\mathbf{U}_i + \mathbf{V}_i)/2$ and $\mathbf{B}_i = (\mathbf{U}_i - \mathbf{V}_i)/2$. Then, it is easy to see that $\mathbf{SQT}_{N-1}$ can be simplified to

$$\mathbf{SQT}_{N-1} = \begin{bmatrix} \mathbf{S}_0\mathbf{U}_{N-1} & \mathbf{0}_{M/2} \\ \mathbf{0}_{M/2} & \mathbf{S}_1\mathbf{V}_{N-1} \end{bmatrix}\mathbf{W}. \tag{12}$$

As $\mathbf{U}_{N-1}$ and $\mathbf{S}_0$ are generic orthogonal matrices, and the product $\mathbf{S}_0\mathbf{U}_{N-1}$ is also a generic orthogonal matrix, we can discard the term $\mathbf{S}_0$ without any loss of generality. The

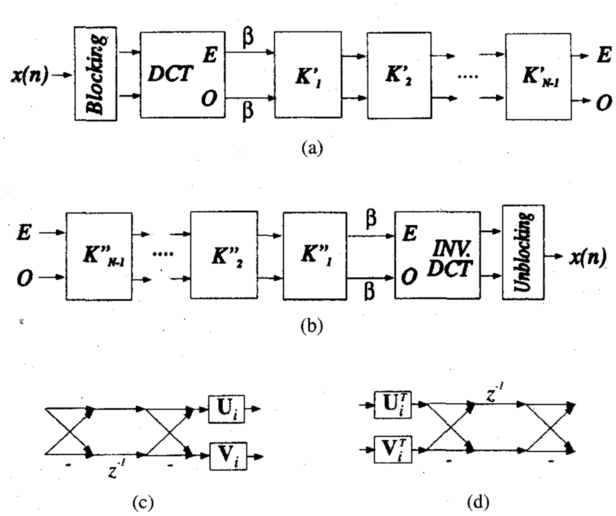


Fig. 5. Flowgraph for implementation of GenLOT's. Each branch carries $M/2$ samples, and e and o stand for the even and odd transform coefficients, respectively, of output (analysis) and input (synthesis) for both DCT and GenLOT. Even and odd coefficients also correspond to symmetric and antisymmetric basis functions (which are the filters' impulse responses), respectively. β is a scaling factor incorporating all scaling factors present in $\mathbf{W}$ so that $\beta = 2^{-(N-1)}$: (a) Analysis; (b) synthesis; (c) details of the analysis stages K'_i; (d) details of the synthesis stages K''_i.

same is valid for $\mathbf{S}_1$ with regard to $\mathbf{V}_{N-1}$. Therefore, we get $\mathbf{SQT}_{N-1} = \Phi_{N-1}\mathbf{W}$, and (9) reduces to

$$\mathbf{E}(z) = \Phi_{N-1}\mathbf{W}\left(\prod_{i=N-2}^{0} \Lambda(z)\mathbf{W}\Phi_i\mathbf{W}\right)\mathbf{Q} \tag{13}$$

or to

$$\mathbf{E}(z) = \left(\prod_{i=N-1}^{1} \Phi_i\mathbf{W}\Lambda(z)\mathbf{W}\right)\mathbf{E}_0 \tag{14}$$

where $\mathbf{E}_0 = \Phi_0\mathbf{WQ}$ is a general $M \times M$ orthogonal matrix with symmetric basis functions, i.e., the PTM of order 0 of a LPPUFB. Since an order-n PTM leads to filters of length $(n+1)M$, a LPPUFB with filter's length $nM + M$ can be obtained from one with filters' length nM by adding a stage to the PTM of the later. If $\mathbf{E}_n(z)$ denotes an order-n PTM, then we can state that

$$\mathbf{E}_n(z) = \mathbf{K}_n(z)\mathbf{E}_{n-1}(z) \tag{15}$$

where

$$\mathbf{K}_i(z) = \Phi_i\mathbf{W}\Lambda(z)\mathbf{W}. \tag{16}$$

Therefore, for any $N > 1$, any PTM of a LPPUFB can be expressed as

$$\mathbf{E}(z) = \mathbf{K}_{N-1}(z)\mathbf{K}_{N-2}(z)\cdots\mathbf{K}_1(z)\mathbf{E}_0. \tag{17}$$

C. Alternative Forms

As a remark, if we let

$$\tilde{\mathbf{I}} = \begin{bmatrix} \mathbf{I}_{M/2} & \mathbf{0}_{M/2} \\ \mathbf{0}_{M/2} & -\mathbf{I}_{M/2} \end{bmatrix}$$

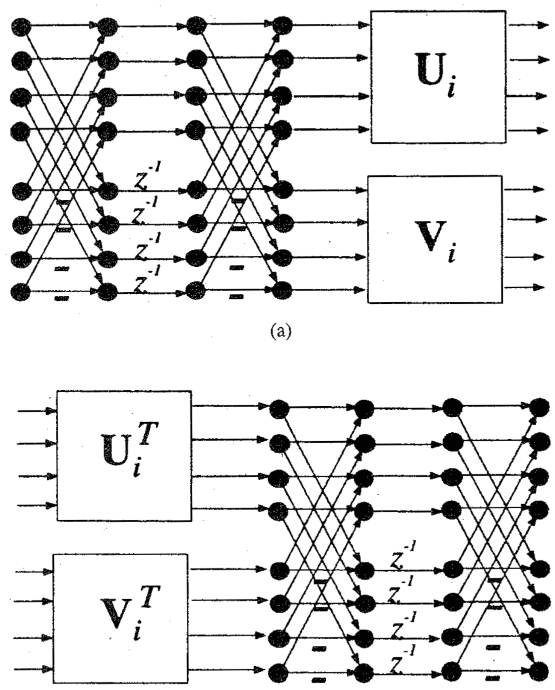


Fig. 6. Details of (a) analysis stage K_i' and (b) synthesis stage K_i'', for $M = 8$.

then $\mathbf{T}_i$ can also be expressed as

$$\mathbf{T}_i = \mathbf{W}^R\Phi\mathbf{W}^R;\ \mathbf{A}_i = \frac{\mathbf{U}_i + \mathbf{V}_i}{2};\ \mathbf{B}_i = \frac{\mathbf{V}_i - \mathbf{U}_i}{2}, \quad (18)$$

$$\tilde{\mathbf{I}}\mathbf{T}_i = \mathbf{W}^R\Phi\mathbf{W};\ \mathbf{A}_i = \frac{\mathbf{V}_i - \mathbf{U}_i}{2};\ \mathbf{B}_i = -\frac{\mathbf{U}_i + \mathbf{V}_i}{2}, \quad (19)$$

$$\tilde{\mathbf{I}}\mathbf{T}_i = \mathbf{W}\Phi\mathbf{W}^R\ ;\ \mathbf{A}_i = \frac{\mathbf{V}_i - \mathbf{U}_i}{2};\ \mathbf{B}_i = \frac{\mathbf{U}_i + \mathbf{V}_i}{2}. \quad (20)$$

Hence, we can say that

$$\mathbf{E}_n(z) = \left(\prod_{i=n}^{1} \Phi_{N-1}\mathbf{W}_i\Lambda(z)\mathbf{W}_{i-1}\right)\Phi_0\mathbf{W}_0\mathbf{Q} \quad (21)$$

where $\mathbf{W}_i$ can be either $\mathbf{W}$ or $\mathbf{W}^R$, such that $\mathbf{T}_i$ is as in (11) or (18). Suppose we violate this rule, for example, by reversing only one $\mathbf{W}$ matrix, as in (19) or (20). Then, we will obtain a PTM $\mathbf{E}_n'(z)$, which is related to the original one by $\mathbf{E}_n'(z) = \pm\tilde{\mathbf{I}}\mathbf{E}_n(z)$. Therefore, $\mathbf{E}_n'(z)$ also corresponds to a LPPUFB, although the sign of some of the filters is inverted. Odd-symmetric filters are not affected because the sign change is equivalent to time-reversion of the coefficients. For even-symmetric filters, the sign change can be compensated by inverting the signs of the elements of the last matrix Φ_i because the odd-symmetric flters are not significantly affected by the overall sign change. As a conclusion, the stage $\mathbf{K}_i(z)$ can be expressed as

$$\mathbf{K}_i(z) = \Phi_i\mathbf{W}_1\Lambda(z)\mathbf{W}_2 \quad (22)$$

where $\mathbf{W}_1$ and $\mathbf{W}_2$ can be either $\mathbf{W}$ or $\mathbf{W}^R$, independently.

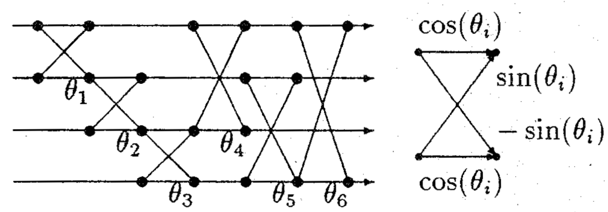


Fig. 7. Implementation of a 4×4 orthogonal matrix through plane rotations. The detail of each plane rotation is shown on the right.

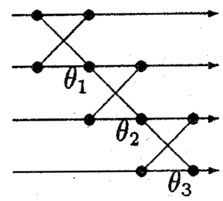


Fig. 8. Implementation of a constrained 4×4 orthogonal matrix using only three plane rotations.

D. Transform Matrix

It is useful to consider the lapped transform matrix $\mathbf{P}$ associated with the LPPUFB [10]. This matrix has size $M \times L$ and elements p_{ij} given by

$$p_{ij} = g_i(j) = f_i(L - 1 - j) \quad (23)$$

for $0 \leq i \leq M - 1$ and $0 \leq j \leq L - 1$. In this way, the filters can be found from $\mathbf{P}$ and vice versa. For LPPUFB's, $\mathbf{P}$ can be found from the following recursion:

$$\mathbf{P}_0 = \mathbf{E}_0 \quad (24)$$

$$\mathbf{P}_i = \Phi_i\mathbf{W}\begin{bmatrix}\mathbf{I}_{M/2} & \mathbf{0}_{M/2} & \mathbf{0}_{M/2} & \mathbf{0}_{M/2}\\ \mathbf{0}_{M/2} & \mathbf{0}_{M/2} & \mathbf{0}_{M/2} & \mathbf{I}_{M/2}\end{bmatrix} \times \begin{bmatrix}\mathbf{W}\mathbf{P}_{i-1} & \mathbf{0}_M\\ \mathbf{0}_M & \mathbf{W}\mathbf{P}_{i-1}\end{bmatrix} \quad (25)$$

$$\mathbf{P} = \mathbf{P}_{N-1}. \quad (26)$$

E. The GenLOT

The GenLOT is defined as a LPPUFB obeying (17), where $\mathbf{E}_0$ is chosen to be the DCT matrix [1], which we denote as $\mathbf{D}$. The output of the DCT is then separated into the groups of even and odd coefficients. The GenLOT with $N-1$ stages after the DCT has basis functions (filters) with length $L = NM$ and has its PTM defined as

$$\mathbf{E}(z) = \mathbf{K}_{N-1}(z)\mathbf{K}_{N-2}(z)\cdots\mathbf{K}_1(z)\mathbf{D}. \quad (27)$$

The implementation flowgraphs for the analysis and synthesis sections are shown in Fig. 5. In this figure, each branch carries $M/2$ samples, and one analysis stage is shown in detail in Fig. 6 for $M = 8$.

The class of GenLOT's, defined in this way, allow us to view the DCT and LOT as special cases, respectively, for $N = 1$ and $N = 2$. The degrees of freedom reside on the matrices $\mathbf{U}_i$ and $\mathbf{V}_i$, which are only restricted to be real $M/2 \times M/2$ orthogonal matrices. Thus, each one can be parameterized into a set of $M(M-2)/8$ plane rotations. Each plane rotation represents one degree of freedom in the design

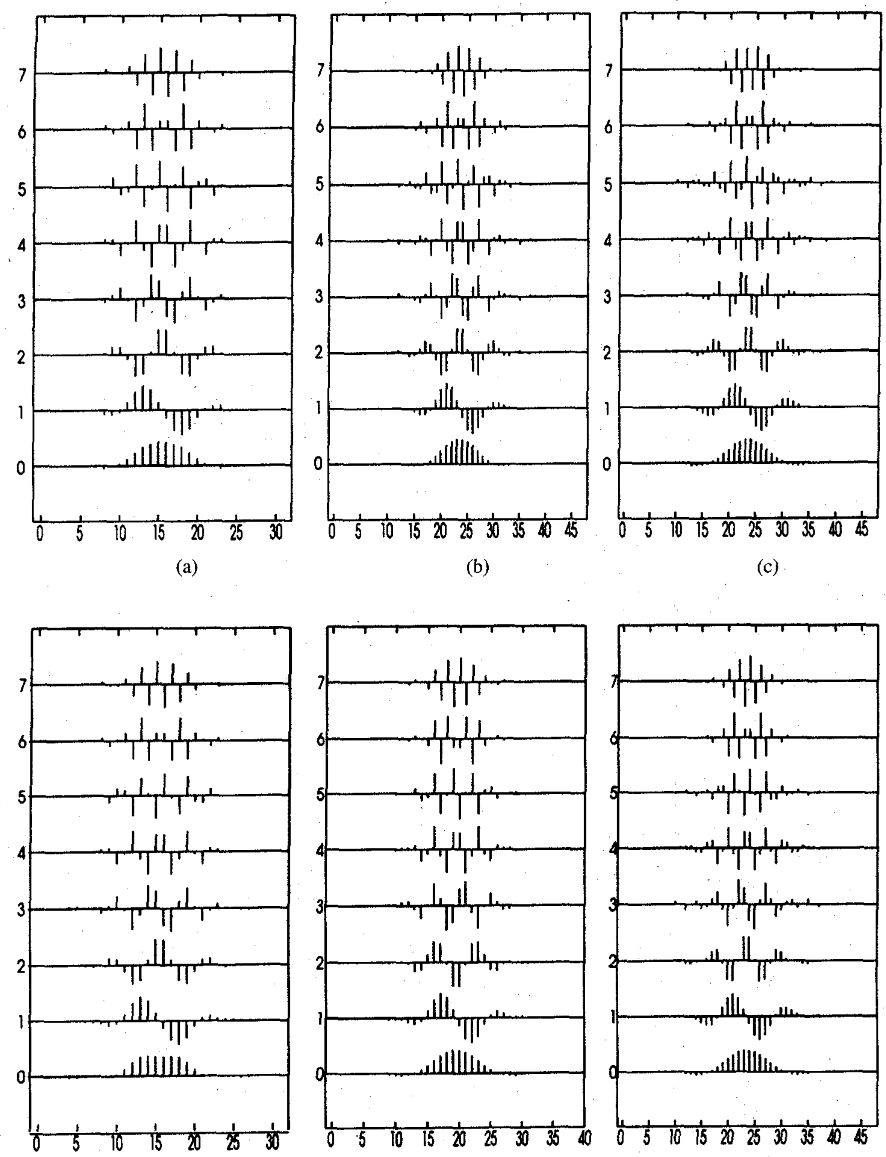


Fig. 9. Basis functions $f_k(n)$ (filters' impulse responses) of a GenLOT with $M = 8$ for different designs: (a) $N = 4$, maximum $G_{\rm TC}$; (b) $N = 5$, maximum $G_{\rm TC}$; (c) $N = 6$, maximum $G_{\rm TC}$; (d) $N = 4$, maximum stopband attenuation; (e) $N = 6$, maximum stopband attenuation; (f) $N = 6$, maximum $G_{\rm TC}$ but with polyphase normalization.

and can be implemented with either three additions and three multiplications or two additions and four multiplications. In either case, the total number of floating-point operations (flops) is 6. For $N-1$ stages after the DCT, this results in a total of $M(N-1)(M-2)/4$ degrees of freedom. For example, for $M = 8$, $\mathbf{U}_i$, and $\mathbf{V}_i$ are 4×4 orthogonal matrices. Hence, each one can be parameterized as a cascade of six plane rotations, as shown in Fig. 7. $\mathbf{U}_i$ and $\mathbf{V}_i$ can be implemented with $3M(M-2)/4$ flops, each, using plane rotations, or $(M-1)M/2$ flops using direct matrix multiplication. Note that for $M > 4$, it is advantageous to use direct matrix multiplication to implement each factor ($\mathbf{U}_i$ or $\mathbf{V}_i$) than to use plane rotations. For $M = 4$, the number of flops is the same, and there are no LPPUFB's for $M = 2$ [5]. Therefore, plane rotations are just useful for the design of Φ_i and not for their implementation. One can achieve a reduction in the implementation cost by forcing each matrix to be composed by a reduced set of plane rotations, let us say $(M/2)-1$. For $M = 8$, a matrix with only three plane rotations is shown in Fig. 8. Using only matrices parameterized in this form, the total number of degrees of freedom is reduced to $(N-1)(M-2)$, which is a reduction of a factor of $M/4$. Each matrix can be implemented with $3M-6$ flops compared with $(M-1)M/2$ flops in direct matrix multiplication.

III. Implementation Over Finite-Length Signals

The input signal is processed as described in Fig. 5. The mth output block has the kth coefficent as a sample of the

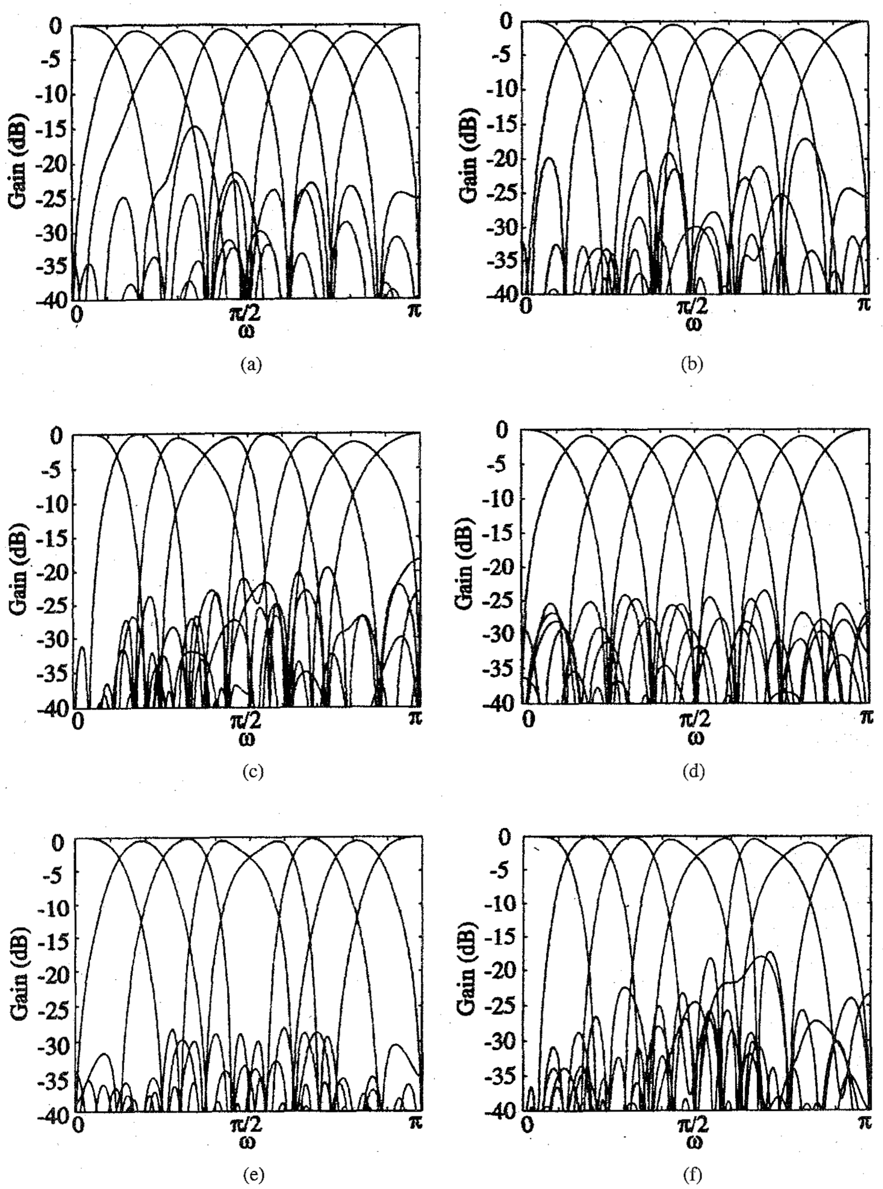


Fig. 10. Filters' frequency responses ($20 \log_{10} |F_k(e^{j\omega})|$), given in decibels, of the GenLOT with $M = 8$ for different designs: (a) $N = 4$, maximum G_{TC}; (b) $N = 5$, maximum G_{TC}; (c) $N = 6$, maximum G_{TC}; (d) $N = 4$, maximum stopband attenuation; (e) $N = 6$, maximum stopband attenuation; (f) $N = 6$, maximum G_{TC} but with polyphase normalization.

kth subband signal $y_k(m)$ as shown in Figs. 2 and 3. It is a *clocked* system, where at each instant (block index), a block of M samples in the time domain is input and tranformed into another block of same dimensions with subband samples. In addition, internal states (corresponding to the delays) left from the previous iteration are used in the process, and they are actually responsible for differentiating a lapped transform from a block transform. The time reference in this *clocked* system is the index of the block of M samples in the input signal. For a signal with very large (or infinite) number of samples, such as speech and audio, the delay to process a block is generally unimportant, and the signal after synthesis can be reconstructed with a delay of approximately N blocks, compared with the original signal before analysis.

Consider a finite-length signal $x(n)$ of N_x samples so that $N_x = N_B M$, i.e., N_B is the number of blocks in the signal and is an integer. As the transform overlaps across the block boundaries, we expect to use more than N_x samples to calculate the N_B transform-domain blocks. Hence, the extra samples are located outsided the signal support region and have to be *guessed*. The choice for these samples may ensure that no abrupt change occurs across the image boundaries. In addition, the initial internal states will affect the analysis or synthesis processes. One of the first solutions to this problem (assuming M-channel filter banks) was used by Malvar [4] when developing the algorithm for the LOT [2], [4], [10] and is tailored for the LOT only. However, several authors studied the problem of processing images with linear-phase filter banks, avoiding the use of periodic convolution [14]–[19].

We need an algorithm independent of the initial states and we will show how to perform analysis and synthesis, independent of the initial internal states, and assuming the signal is continuous across the signal borders using a symmetric extension of the boundary samples inside the support region of the signal. Furthermore, perfect reconstruction of the signal can be achieved (assuming no processing/quantization of the subbands) using only N_B samples in each subband. The approach used here is a consequence of the results presented in [14]. As the main difference between a general LPPUFB and a GenLOT is in the design and on the choice of the first stage as the DCT, the results of this section apply to any M-channel (M even) uniform FIR LPPUFB by replacing the DCT matrix by $\mathbf{E}_0$.

A. Analysis

Let $x(0), \ldots, x(N_x - 1)$ be the samples in the input signal. Extend the signal through a mirror-image reflection applied to the last $\lambda = (L - M)/2$ samples on each border, resulting in a signal $\tilde{x}(n)$ with $N_x + 2\lambda = N_x + L - M$ samples, as

$$x(\lambda - 1), \cdots, x(0), x(0), \cdots, x(N_x - 1),$$
$$x(N_x - 1), \cdots, x(N_x - \lambda).$$

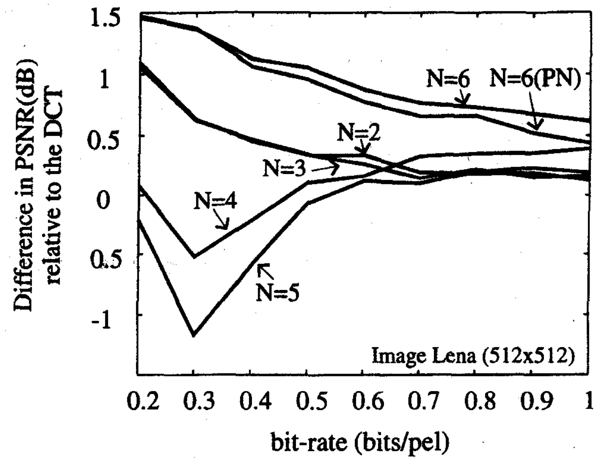
Process this signal, which corresponds to $N_B + N - 1$ blocks. Discard the first $N - 1$ output blocks, obtaining N_B transform-domain blocks corresponding to N_B samples of each subband. The internal states in Fig. 5 can be initialized in any way.

B. Synthesis

The general strategy to achieve PR without great increase in complexity or change in the implementation algorithm is to extend the samples in the subbands, generating more blocks to be inverse transformed, in such a way that after synthesis, assuming no processing of the subband signals, the signal recovered is identical to the original at the borders. The extension of the kth subband signal depends on the symmetry of the kth filter. Let $f_k(n) = v_k f_k(L - 1 - n)$ for $0 \leq k \leq M - 1$ and $0 \leq n \leq L - 1$, i.e., $v_k = 1$ if $f_k(n)$ is symmetric and $v_k = -1$ if $f_k(n)$ is antisymmetric. Before synthesis, for each subband signal $\hat{y}_k(m)$ of N_B samples, fold the borders of $\hat{y}_k(m)$ (as in the analysis section) in order to find a signal $\hat{\tilde{y}}_k(m)$ and invert the sign of the extended samples if $f_k(n)$ is antisymmetric. For s samples reflected around the borders, then the kth subband signal will have samples

$$v_k\hat{y}_k(s - 1), \cdots, v_k\hat{y}_k(0), \hat{y}_k(0), \cdots \hat{y}_k(N_B - 1),$$
$$v_k\hat{y}_k(N_B - 1), \cdots, v_k\hat{y}_k(N_B - s).$$

- *N odd*: Reflect $s = (N - 1)/2$ samples around each border, thus getting $N_B + N - 1$ blocks to be processed as in the synthesis flowgraph in Fig. 5. To obtain the inverse transformed samples $\hat{x}(n)$, initialize the internal states in any way, run the synthesis process over the $N_B + N - 1$ blocks, and discard the first $N - 1$ reconstructed blocks, retaining the $N_x = N_B M$ remaining samples.



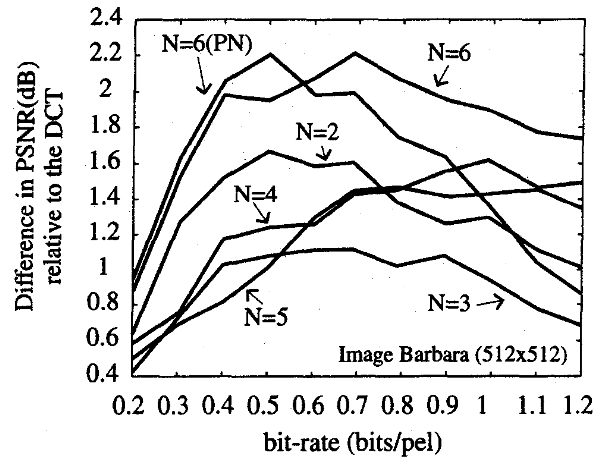


Fig. 11. PSNR (in decibels) difference among GenLOT's optimized for maximum G_{TC} and the DCT for several bit-rates using test images "Lena" and "Barbara." PN stands for the GenLOT with $N = 6$ and with polyphase normalization (design #6).

- *N even*: Reflect $s = N/2$ samples around each border, thus getting $N_B + N$ blocks to be processed as in the synthesis flowgraph of Fig. 5. To obtain the inverse transformed samples $\hat{x}(n)$, initialize the internal states in any way, and run the synthesis process over the $N_B + N$ blocks. Discard the first $N - 1$ reconstructed blocks and the first $M/2$ samples of the Nth block. Include in the reconstructed signal the last $M/2$ samples of the Nth block and the subsequent $(N_B - 1)M$ samples. In the last block, include the first $M/2$ samples in the reconstructed signal, and discard the rest.

This approach will assure the PR property and orthogonality of the analysis and synthesis processes [20]. The price paid is to run the algorithm over extra N or $N - 1$ blocks. As it is common to have $N_B \gg N$, the computational increase is only marginal.

IV. Design

The LOT can be obtained from the DCT by direct determination of Φ_1 [10]. In this case, $\mathbf{U}_1$ and $\mathbf{V}_1$ are determined in a general form, without obeying any particular structure.

Optimization, in this case, is carried solely to determine an approximation to the matrices $\mathbf{U}_1$ and $\mathbf{V}_1$ found through the techniques described in [10]. For the LOT, $\mathbf{U}_1$ is approximated to $\mathbf{I}_{M/2}$, and $\mathbf{V}_1$ is approximated by a cascade of $\frac{M}{2}-1$ plane rotations [2] through optimization routines. Therefore, for the LOT, the optimization is only necessary to find faster implementation algorithms. However, for $N > 2$, there are no techniques available to find directly all matrices Φ_i. The design of a GenLOT is the determination of the q free parameters (angles for the plane rotations). This number can be $q = M(N-1)(M-2)/4$ for the full set of rotations or $q = (N-1)(M-2)$ for the reduced one. The q-dimensional space of solutions is searched through optimization routines in such a way as to minimize a particular cost function. However, due to the highly nonlinear relationships among the angles and the cost functions, there is no guarantee of obtaining a global minimum. All GenLOT examples presented here were obtained using unconstrained nonlinear optimization and simplex search, using the routines provided by MATLAB[1] version 4.0.

Examples of features we can try to maximize in the design are the transform coding gain (G_{TC}) [21] or a measure of the atenuation in the stopband region of each filter, or a combination of both. Other features can be considered as well. Thus, the cost function can be selected as the inverse of any of these functions.

A. Coding Gain

Let the autocorrelation matrix for this process be $\mathbf{R}_{xx}$. Then, the transformed signal has an autocorrelation matrix given by [2]

$$\mathbf{R}_{yy} = \mathbf{P}\mathbf{R}_{xx}\mathbf{P}^T \tag{28}$$

with elements $r_y(i,j)$. The cost function J to be minimized is the inverse of the coding gain [21] as

$$J_{\mathrm{GAIN}} = -G_{\mathrm{TC(dB)}} = 10\log_{10}\left(\frac{\left(\prod_{i=0}^{M-1} r_y(i,i)\right)^{1/M}}{\frac{1}{M}\sum_{i=0}^{M-1} r_y(i,i)}\right). \tag{29}$$

If the full set of angles is used, we can speed up the optimization by not optimizing the last stage, i.e., Φ_{N-1}. This is possible by using the method applied by Malvar [2] for the LOT. For this, in the recursion to find $\mathbf{P}$, assume that

$$\begin{aligned}\tilde{\mathbf{P}} &= \mathbf{W}\begin{bmatrix}\mathbf{I}_{M/2} & \mathbf{0}_{M/2} & \mathbf{0}_{M/2} & \mathbf{0}_{M/2}\\ \mathbf{0}_{M/2} & \mathbf{0}_{M/2} & \mathbf{0}_{M/2} & \mathbf{I}_{M/2}\end{bmatrix}\\ &\times\begin{bmatrix}\mathbf{W}\mathbf{P}_{N-2} & \mathbf{0}_M\\ \mathbf{0}_M & \mathbf{W}\mathbf{P}_{N-2}\end{bmatrix}\end{aligned}$$

so that $\mathbf{P} = \Phi_{N-1}\tilde{\mathbf{P}}$. The matrix Φ_{N-1} for maximum decorrelation of the input signal (given matrices Φ_1 through Φ_{N-2} and a statistical model for the input) is given by the matrix whose rows are the M eigenvectors of $\tilde{\mathbf{P}}\mathbf{R}_{xx}\tilde{\mathbf{P}}^T$ [2].

[1] MATLAB is a trademark of The Math Works Inc.

Thus, for $N = 4$ (three times the overlap amount present in the LOT), it is only necessary to optimize two out of four stages (because the first stage is DCT, and the last stage is determined by the remaining ones). For a reduced set of angles, this method does not make sense because it would force us to run a second optimization to approximate $\mathbf{U}_{N-1}$ and $\mathbf{V}_{N-1}$ by a series of $M/2-1$ plane rotations each.

B. Stopband Atenuation

Another criteria for the design of the GenLOT can be the maximization of the stopband atenuation of the filters $f_k(n)$ $(0 \le kM-1$, and $0 \le n \le L-1)$. Let $F_k(e^{j\omega})$ be the Fourier transform of $f_k(n)$, which is a bandpass filter with low and high cutoff frequencies denoted by $\omega_{k,L}$ and $\omega_{k,H}$. Let the filters be sorted by their frequency slots so that

$$\omega_{k,L} = k\pi/M, \quad \omega_{k,H} = (k+1)\pi/M.$$

The stopband region Ω_k corresponding to $f_k(n)$ is defined by

$$\begin{aligned}\Omega_0 &\equiv \{\omega \mid |\omega| \in [\omega_{0,H}+\epsilon, \pi)\}\\ \Omega_k &\equiv \{\omega \mid |\omega| \in ([0, \omega_{k,L}-\epsilon] \cup [\omega_{k,H}+\epsilon, \pi))\}\\ \Omega_{M-1} &\equiv \{\omega \mid |\omega| \in [0, \omega_{M-1,L}-\epsilon]\}\end{aligned} \tag{30}$$

where ϵ is a small positive real number used to reduce the influence of the transition region into the stopband region.

A possible cost function to be minimized can be the energy of the filters frequency response in the stopband region, which is defined as

$$J_{\mathrm{STOP}} = \sum_{k=0}^{M-1}\int_{\omega\in\Omega_k} |F_k(e^{j\omega})|^2\, d\omega. \tag{31}$$

C. Polyphase Normalization and DC Leakage

For a constant input, it is sometimes desirable that only one transform coefficient is nonzero. Such a coefficient is, therefore, called the DC coefficient. This property is commonly referred as polyphase normalization and when it does not hold the filter bank is said to have DC leakage (leakage to other coefficients other than the dc one). As the filter bank is paraunitary, the power-complementary property implies

$$\sum_{k=0}^{M-1} |F_k(e^{j\omega})|^2 = \sqrt{M}.$$

In frequency domain, polyphase normalization means that

$$F_k(e^{j\omega})|_{\omega=0} = F_k(1) = 0 \quad k > 0, \tag{32}$$

$$F_0(1) = \sqrt{M}. \tag{33}$$

Note that one of the above equations implies the other. In this case, we can translate (33) to the time domain as

$$\sum_{n=0}^{L-1} f_0(n) = \sqrt{M} \tag{34}$$

and define a cost function as

$$J_{\mathrm{DC}} = \left|\sum_{n=0}^{L-1} f_0(n) - \sqrt{M}\right|. \tag{35}$$

The above condition may not be used as a cost function by itself. It may actually be used in conjuction with other cost

Fig. 12. Reconstructed versions of image "Barbara": (top left) using DCT at 0.25 b/pel; (top right) using a GenLOT with $N = 6$ (design #6) at 0.25 b/pel; (bottom left) using DCT at 0.4 b/pel; (bottom right) using a GenLOT with $N = 6$ (design #6) at 0.4 b/pel.

functions as

$$J = \alpha_1 J_{\mathrm{GAIN}} + \alpha_2 J_{\mathrm{DC}} \quad \text{or} \quad J = \alpha_1 J_{\mathrm{STOP}} + \alpha_2 J_{\mathrm{DC}} \tag{36}$$

where α_i are simple weights for the respective cost functions.

D. Design Examples

Several GenLOT's were designed and tested. We will present a small but illustrative set of GenLOT's. We have selected $M = 8$ for illustrative purposes, and we present six GenLOT examples:

1) optimized for maximum G_{TC} with $N = 4$ (32-tap filters)
2) optimized for maximum G_{TC} with $N = 5$ (40-tap filters)
3) optimized for maximum G_{TC} with $N = 6$ (48-tap filters)
4) optimized for maximum stopband attenuation with $N = 4$ (32-tap filters)
5) optimized for maximum stopband attenuation with $N = 6$ (48-tap filters)
6) optimized for maximum G_{TC} with $N = 6$ (48-tap filters), but including the cost function J_{DC} (i.e., polyphase normalization).

The coding gain was calculated assuming the input signal as a zero mean AR(1) signal with adjacent sample correlation coefficient 0.95 (i.e., its autocorrelation function is $r_x(n) = 0.95^{|n|}$). The impulse responses $f_k(n)$ of these filters are plotted in Fig. 9. Their respective frequency responses $|F_k(e^{j\omega})|$ are shown in Fig. 10.

We tested the performance of the GenLOT using the maximum G_{TC} design in image coding. The coder algorithm used is the JPEG baseline system [22], merely replacing the

TABLE I
$N = 6$, DESIGNED FOR MAXIMUM G_{TC},
CONSTRAINED TO HAVE POLYPHASE NORMALIZATION

$f_0(n)$	$f_1(n)$	$f_2(n)$	$f_3(n)$	$f_4(n)$	$f_5(n)$	$f_6(n)$	$f_7(n)$
-0.000530	-0.000596	-0.000454	-0.000463	-0.000673	0.000574	0.000273	0.000342
-0.000457	0.000902	-0.000476	0.000857	0.001225	0.000732	-0.000399	0.000114
0.003066	-0.002729	0.004887	-0.000943	-0.003088	-0.005685	0.000219	-0.001485
0.000949	0.000980	0.000816	0.000300	0.000942	-0.001196	0.000205	-0.000128
-0.004275	0.008698	-0.008881	0.003916	0.009709	0.009998	-0.001589	0.001725
0.002707	0.001569	0.002403	0.000003	0.001203	-0.003537	0.000794	-0.000455
0.006701	0.001928	0.007774	0.002266	0.002072	-0.009485	-0.001396	-0.003572
0.002226	0.010035	0.001748	0.006722	0.011839	-0.002111	-0.001483	-0.000138
0.020445	0.005747	0.025283	-0.000747	-0.003772	-0.024037	-0.001651	-0.009860
0.005916	-0.026386	0.003612	-0.019953	-0.039914	-0.021234	0.007586	0.000643
-0.004552	-0.027360	-0.025185	-0.025006	0.019764	0.050310	-0.008128	0.008131
-0.014804	-0.001890	-0.005886	0.010839	0.015341	0.007971	-0.012466	-0.009244
-0.028875	0.011958	0.006381	0.042207	-0.059065	-0.082959	0.033648	-0.000020
-0.050474	-0.009635	-0.026121	0.022157	0.038625	0.047133	-0.027038	-0.016220
-0.052974	-0.054295	-0.076701	-0.012207	0.051332	0.062341	0.011183	0.038324
-0.041136	-0.106574	-0.048255	-0.057312	-0.098980	-0.053651	-0.006360	-0.023082
-0.029504	-0.151656	0.081177	-0.079048	0.119873	-0.064979	0.069196	0.029360
0.028973	-0.137742	0.208095	0.043694	0.033750	0.186027	-0.106130	-0.012807
0.079061	-0.035396	0.168522	0.240059	-0.264146	-0.095672	0.039440	-0.031329
0.160365	0.155557	-0.089954	0.007153	0.040556	-0.162141	0.129591	0.138515
0.240193	0.334046	-0.354118	-0.371751	0.374019	0.357922	-0.309000	-0.257022
0.323746	0.409128	-0.325729	-0.157785	-0.177828	-0.262903	0.425901	0.354909
0.365394	0.339754	0.038501	0.388145	-0.367310	-0.117174	-0.401403	-0.391886
0.402053	0.148177	0.412559	0.336892	0.294525	0.441633	0.159009	0.361495

TABLE II
$N = 4$, DESIGNED FOR MAXIMUM G_{TC}.

$f_0(n)$	$f_1(n)$	$f_2(n)$	$f_3(n)$	$f_4(n)$	$f_5(n)$	$f_6(n)$	$f_7(n)$
0.004799	0.004829	0.002915	-0.002945	0.000813	-0.000109	0.000211	0.000483
0.009320	-0.000069	-0.005744	-0.010439	0.001454	0.003206	0.000390	-0.001691
0.006394	-0.005997	-0.011121	-0.010146	0.000951	0.004317	0.000232	-0.002826
-0.011794	-0.007422	-0.001800	0.009462	-0.001945	-0.001342	-0.000531	0.000028
-0.032408	-0.009604	0.008083	0.031409	-0.005262	-0.007504	-0.001326	0.003163
-0.035122	-0.016486	0.001423	0.030980	-0.005715	-0.006029	-0.001554	0.001661
-0.017066	-0.031155	-0.027246	0.003473	-0.003043	0.005418	-0.000789	-0.005605
0.000288	-0.035674	-0.043266	-0.018132	-0.000459	0.013004	-0.000165	-0.010084
-0.012735	-0.053050	0.007163	-0.083325	0.047646	0.011562	0.048534	0.043066
-0.018272	-0.090207	0.131531	0.046926	0.072761	-0.130875	-0.089467	-0.028641
0.021269	-0.054379	0.109817	0.224818	-0.224522	0.136666	0.022488	-0.025219
0.126784	0.112040	-0.123484	-0.032818	-0.035078	0.107446	0.147727	0.109817
0.261703	0.333730	-0.358887	-0.379088	0.384874	-0.378415	-0.339368	-0.216652
0.357269	0.450401	-0.292453	-0.126901	-0.129558	0.344379	0.439129	0.317070
0.383512	0.369819	0.097014	0.418643	-0.419231	0.045807	-0.371449	-0.392556
0.370002	0.140761	0.478277	0.318691	0.316307	-0.433937	0.146036	0.427668

8×8 DCT by a 8×8 GenLOT ($N > 1$) obtained through separable implementation of the 1-D transform (as is the case for the 2-D DCT). For the 8 b/pel 512×512-pels images "Lena" and "Barbara," we tested the JPEG coder comparing the GenLOT's with $N = 1$ (DCT), $N = 2$ (optimal LOT [2], [10]), and GenLOT's with $N = 3$ through $N = 6$. For $N = 6$, two versions were included in the tests, and both were optimized for maximum G_{TC}. However, one has polyphase normalization. The reason for the inclusion of the two types of GenLOT's is because without polyphase normalization, one can achieve higher G_{TC} and, perhaps, higher peak signal-to-noise ratio (PSNR) after decompressing the image. However, the design with polyphase normalization apparently yields decompressed images with higher visual quality. The difference in PSNR (in decibels) among the GenLOT's and the DCT is shown in Fig. 11. Reconstructed versions of image "Barbara" coded at 0.25 and 0.4 b/pel, using GenLOT's with $N = 1$ (regular JPEG) and $N = 6$ (replacing the DCT by a GenLOT with polyphase normalization) are shown in Fig. 12.

The coefficients of some eight-channel GenLOT's used as examples are shown in Tables I–V. Only half of the filter taps are shown because the bases are (anti) symmetric.

The maximum G_{TC} design is not necessarily the best one for image coding, even considering that the AR(1) process is, in general, a good model for images. For example, the "smoothness" of the basis functions is an important issue because in low bit-rate coding, only few coefficients are nonzero, and thus, the signal is reconstructed using only few basis functions. If these basis functions are very concentrated or present "bends" or "edges," then these will produce visible patterns in the reconstructed image. Such image could have a better aspect if the lowest frequency basis functions were

TABLE III
$N = 6$, DESIGNED FOR MAXIMUM G_{TC}.

$f_0(n)$	$f_1(n)$	$f_2(n)$	$f_3(n)$	$f_4(n)$	$f_5(n)$	$f_6(n)$	$f_7(n)$
-0.000137	-0.000225	0.000234	0.000058	-0.000196	-0.000253	0.000078	0.000017
-0.000222	-0.000228	0.000388	0.000471	0.000364	0.000163	-0.000220	-0.000283
0.001021	0.000187	0.002439	0.001211	-0.000853	-0.002360	0.000157	-0.000823
0.000536	0.000689	0.000029	0.000535	0.000572	0.000056	0.000633	0.000502
-0.001855	0.000515	-0.006584	-0.002809	0.003177	0.006838	-0.000886	0.001658
0.001429	0.001778	-0.000243	0.000834	0.000977	-0.000056	0.001687	0.001429
0.001440	0.001148	0.000698	0.000383	0.000109	-0.000561	-0.000751	-0.001165
0.001056	0.001893	0.002206	0.005386	0.005220	0.001676	0.001673	0.000792
0.009734	0.002899	0.018592	0.004888	-0.006600	-0.018889	-0.000261	-0.006713
-0.005196	-0.013699	-0.008359	-0.021094	-0.020406	-0.009059	-0.012368	-0.005263
-0.000137	-0.001344	-0.027993	-0.028046	0.026048	0.024169	-0.001643	-0.000402
-0.007109	-0.002130	0.002484	0.013289	0.013063	0.002655	-0.002180	-0.006836
-0.011238	-0.002219	0.033554	0.062616	-0.058899	-0.031538	-0.001404	0.004060
-0.020287	-0.006775	0.003214	0.019082	0.018132	0.004219	-0.006828	-0.019040
-0.028214	-0.018286	-0.059401	-0.023539	0.024407	0.056646	0.009849	0.021475
-0.034379	-0.055004	-0.048827	-0.052703	-0.051123	-0.048429	-0.049853	-0.031732
-0.029911	-0.106776	0.070612	-0.088796	0.086462	-0.066383	0.097006	0.031014
-0.004282	-0.107167	0.197524	0.049701	0.051188	0.193302	-0.104953	-0.006324
0.058553	-0.026759	0.144748	0.241758	-0.239193	-0.143627	0.020370	-0.048085
0.133701	0.147804	-0.123524	0.026563	0.025910	-0.125263	0.147501	0.130959
0.231898	0.330343	-0.376982	-0.365965	0.366426	0.377886	-0.332858	-0.228016
0.318102	0.430439	-0.312564	-0.174852	-0.174803	-0.314092	0.431705	0.317994
0.381693	0.368335	0.061832	0.393949	-0.395534	-0.060887	-0.369244	-0.384842
0.417648	0.144412	0.409688	0.318912	0.319987	0.411214	0.145256	0.419936

TABLE IV
$N = 6$, DESIGNED FOR MAXIMUM STOPBAND ATENUATION.

$f_0(n)$	$f_1(n)$	$f_2(n)$	$f_3(n)$	$f_4(n)$	$f_5(n)$	$f_6(n)$	$f_7(n)$
-0.000137	-0.000225	0.000234	0.000058	-0.000196	-0.000253	0.000078	0.000017
-0.000222	-0.000228	0.000388	0.000471	0.000364	0.000163	-0.000220	-0.000283
0.001021	0.000187	0.002439	0.001211	-0.000853	-0.002360	0.000157	-0.000823
0.000536	0.000689	0.000029	0.000535	0.000572	0.000056	0.000633	0.000502
-0.001855	0.000515	-0.006584	-0.002809	0.003177	0.006838	-0.000886	0.001658
0.001429	0.001778	-0.000243	0.000834	0.000977	-0.000056	0.001687	0.001429
0.001440	0.001148	0.000698	0.000383	0.000109	-0.000561	-0.000751	-0.001165
0.001056	0.001893	0.002206	0.005386	0.005220	0.001676	0.001673	0.000792
0.009734	0.002899	0.018592	0.004888	-0.006600	-0.018889	-0.000261	-0.006713
-0.005196	-0.013699	-0.008359	-0.021094	-0.020406	-0.009059	-0.012368	-0.005263
-0.000137	-0.001344	-0.027993	-0.028046	0.026048	0.024169	-0.001643	-0.000402
-0.007109	-0.002130	0.002484	0.013289	0.013063	0.002655	-0.002180	-0.006836
-0.011238	-0.002219	0.033554	0.062616	-0.058899	-0.031538	-0.001404	0.004060
-0.020287	-0.006775	0.003214	0.019082	0.018132	0.004219	-0.006828	-0.019040
-0.028214	-0.018286	-0.059401	-0.023539	0.024407	0.056646	0.009849	0.021475
-0.034379	-0.055004	-0.048827	-0.052703	-0.051123	-0.048429	-0.049853	-0.031732
-0.029911	-0.106776	0.070612	-0.088796	0.086462	-0.066383	0.097006	0.031014
-0.004282	-0.107167	0.197524	0.049701	0.051188	0.193302	-0.104953	-0.006324
0.058553	-0.026759	0.144748	0.241758	-0.239193	-0.143627	0.020370	-0.048085
0.133701	0.147804	-0.123524	0.026563	0.025910	-0.125263	0.147501	0.130959
0.231898	0.330343	-0.376982	-0.365965	0.366426	0.377886	-0.332858	-0.228016
0.318102	0.430439	-0.312564	-0.174852	-0.174803	-0.314092	0.431705	0.317994
0.381693	0.368335	0.061832	0.393949	-0.395534	-0.060887	-0.369244	-0.384842
0.417648	0.144412	0.409688	0.318912	0.319987	0.411214	0.145256	0.419936

TABLE V
$N = 4$, DESIGNED FOR MAXIMUM STOPBAND ATENUATION.

$f_0(n)$	$f_1(n)$	$f_2(n)$	$f_3(n)$	$f_4(n)$	$f_5(n)$	$f_6(n)$	$f_7(n)$
-0.001195	-0.001281	0.000648	-0.000474	0.000738	-0.000506	-0.000501	-0.000724
0.002427	-0.001444	0.002063	-0.002005	0.002636	0.002225	0.002004	0.003171
0.001285	0.004539	-0.003097	0.003373	-0.004595	-0.000607	0.000112	-0.000155
0.000963	0.002747	-0.001935	0.001426	-0.002202	-0.000265	0.000456	0.000275
0.001294	-0.003426	0.002757	-0.004287	0.005447	0.002474	0.000912	0.001883
0.000254	0.007906	-0.006468	0.005203	-0.007619	-0.002536	-0.000658	-0.002094
0.009307	0.013898	-0.008095	0.006922	-0.010181	0.002481	0.003553	0.004603
0.020214	0.005003	0.003092	-0.005164	0.005173	0.012652	0.014885	0.021325
0.061866	0.081866	-0.022559	0.063402	-0.063648	0.022184	-0.071325	-0.063950
-0.003180	0.079320	-0.141730	-0.074203	-0.067788	-0.159205	0.073406	-0.003704
-0.033868	0.031999	-0.134430	-0.199122	0.212518	0.126809	-0.024039	0.020479
-0.117796	-0.138045	0.084738	0.028858	0.023472	0.088267	-0.140896	-0.116246
-0.225508	-0.335045	0.375433	0.391870	-0.389178	-0.381980	0.339016	0.217549
-0.324735	-0.446667	0.345393	0.130278	0.133268	0.340326	-0.445176	-0.323426
-0.382887	-0.367919	-0.049095	-0.426039	0.423327	0.047013	0.368276	0.386332
-0.421573	-0.139535	-0.437439	-0.312747	-0.311387	-0.431468	-0.141824	-0.424600

smoother, even though they could lead to a GenLOT with lower G_{TC} and/or DC leakage.

V. Conclusion

The general factorization of LPPUFB's is revisited, leading to a new perspective from which the GenLOT's emerged as a trivial particularization. One of the most interesting properties is that the procedure to increase the overlap (filters' length) is identical for any order n by applying a post-processing stage $\mathbf{K}_n(z)$. The elegance of the factorization and the fact that it is a linear-phase filter bank with a fast algorithm based on the DCT are important attributes for GenLOT's.

The large number of degrees of freedom forced us to use nonlinear optimization procedures in the design of GenLOT's. This is not very desirable because we cannot guarantee a global minimum of the cost function but only a local one. However, for most of our tests, several different initializations led to the same resulting angles, even when very distant starting points were used. This leads us to believe that the optimized solutions are reasonably stable.

In dealing with optimization for signal compression, the major problem is the definition of the cost function. Further research may be concentrated on design issues aimed at specific applications.

Acknowledgment

The authors wish to thank S. Trautman for his help in updating the design of GenLOT's. He actually designed some GenLOT's used in this paper (designs 4, 5, and 6). The authors are also thankful to H. S. Malvar for his participation on discussions about the LOT and its factorization.

References

[1] K. R. Rao and P. Yip, *Discrete Cosine Transform: Algorithms, Advantages, Applications.* San Diego, CA: Academic, 1990.

[2] H. S. Malvar and D. H. Staelin, "The LOT: Transform coding without blocking effects," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 37, pp. 553–559, Apr. 1989.

[3] P. Cassereau, "A new class of optimal unitary transforms for image processing," Master's thesis, Mass. Inst. Technol., Cambridge, MA, May 1985.

[4] H. S. Malvar, "Optimal pre- and post-filters in noisy sampled data systems," Ph.D. dissertation, Mass. Inst. Technol., Cambridge, MA, Sept. 1986.

[5] P. P. Vaidyanathan, *Multirate Systems and Filter Banks*, Englewood Cliffs, NJ: Prentice-Hall, 1993.

[6] T. Q. Nguyen and P. P. Vaidyanathan, "Structures for M-channel perfect reconstruction FIR QMF banks which yield linear phase analysis filters," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 38, pp. 433–446, Mar. 1990.

[7] T. Q. Nguyen and P. P. Vaidyanathan, "Two-channel perfect reconstruction FIR QMF structures which yield linear phase analysis and synthesis filters," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 37, pp. 676–690, May 1989.

[8] M. Vetterli and C. Herley, "Wavelets and filter banks: Theory and design," *IEEE Trans. Signal Processing*, vol. 40, pp. 2207–2232, Sept. 1992.

[9] M. Vetterli and D. Le Gall, "Perfect reconstruction filter banks: Some properties and factorizations," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 37, pp. 1057–1071, July 1989.

[10] H. S. Malvar, *Signal Processing With Lapped Transforms.* Norwood, MA: Artech House, 1992.

[11] A. K. Soman, P. P. Vaidyanathan, and T. Q. Nguyen, "Linear-phase orthonormal filter banks," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, Minneapolis, MN, vol. III, Apr. 1993, pp. 209–212.

[12] ——, "Linear-phase paraunitary filter banks: Theory, factorizations and applications," *IEEE Trans. Signal Processing*, vol. 41, Dec. 1993.

[13] R. E. Crochiere and L. R. Rabiner, *Multirate Digital Signal Processing.* Englewood Cliffs, NJ: Prentice-Hall, 1983.

[14] R. L. de Queiroz, "Subband processing of finite length signals without border distortions," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, San Francisco, CA, vol. IV, 1992, pp. 613–616.

[15] M. J. Smith and S. L. Eddins, "Analysis/synthesis techniques for subband image coding," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 38, pp. 1446–1456, Aug. 1990.

[16] G. Karlsson and M. Vetterli, "Extension of finite length signals for subband coding," *Signal Processing*, vol. 17, pp. 161–168, 1989.

[17] K. Nishikawa, H. Kiya, and M. Sagawa, "Property of circular convolution for subband image coding," in *Proc. Int. Conf. Acoust., Speech, Signal Processing*, vol. IV, 1992, pp. 281–284.

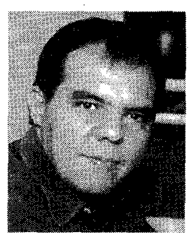
[18] S. Martucci, "Signal extension and noncausal filtering for subband coding of images," in *Proc. SPIE VCIP '91: Visual Commun. Image Processing*, vol. 1605, Nov. 1991, pp. 137–148.

[19] V. Nury and R. Bamberger, "A theory of size-limited filter banks," in *Proc. Int. Conf. Acoust., Speech, Signal Processing*, vol. III, 1993, pp. 161–164.

[20] R. L. de Queiroz and K. R. Rao, "On orthogonal transforms of images using paraunitary filter banks," *J. Visual Commun. Image Representation*, vol. 6, no. 2, pp. 142–153, June 1995.

[21] N. S. Jayant and P. Noll, *Digital Coding of Waveforms.* Englewood Cliffs, NJ: Prentice-Hall, 1984.

[22] W. B. Pennebaker and J. L. Mitchell, *JPEG: Still Image Compression Standard.* New York: Van Nostrand Reinhold, 1993.



Ricardo L. de Queiroz received the B.S. degree from Universidade de Brasilia, Brazil, in 1987, the M.S. degree from Universidade Estadual de Campinas, Brazil, in 1990, and the Ph.D. degree from University of Texas at Arlington, in 1994, all in electrical engineering.

From 1990 to 1991, he was with the DSP research group at Universidade de Brasilia as a research associate. In 1994, he was a teaching assistant with the Electrical Engineering Department of the University of Texas at Arlington. He joined Xerox Corp., Webster, NY, USA, in August 1994, where he is currently a member of the research staff at the Advanced Color Imaging Group. His research interests are multirate signal processing, filter banks, image and signal compression, color imaging, and processing of compressed images.

Dr. de Queiroz received the Academic Excellence Award from the Electrical Engineering Department of the University of Texas at Arlington in 1993.

Truong Q. Nguyen (M'90), for a photograph and biography, please see this issue, p. 483.

K. R. Rao (SM'73) received the Ph.D. degree in electrical engineering from the University of New Mexico, Albuquerque, in 1966.

Since 1966, he has been with the University of Texas at Arlington (UTA), where he is currently a professor of electrical engineering. He has published extensively in reviewed technical journals in the areas of discrete orthogonal transforms and digital image coding. He, along with two other researchers, introduced the discrete cosine transform in 1975, which has since become very popular in digital signal/image processing. He is the co-author of the books *Orthogonal Transforms for Digital Signal Processing* (Springer-Verlag, 1975), *Fast Transforms: Analyses and Applications* (Academic, 1982), and *Discrete Cosine Transform—Algorithm, Advantages, and Applications* (Academic, 1990), as well as coauthor or editor on other works. He is co-author of the book *Digital Image/Video/Audio Coding and International Standards*, which will be published by Prentice-Hall. Some of his books have been translated in to Chinese, Russian, and Japanese. He has conducted workshops on digital image coding worldwide.