AN ADAPTIVE APPROACH TO WAVELET FILTERS DESIGN

Nicola Neretti, Nathan Intrator
Institute for Brain and Neural Systems
Brown University, Providence, RI 02912, US
Phone: (401) 863-3920
Fax: (401) 863-3494
E-mail: Nicola_Neretti@brown.edu

Abstract. We present a general framework for the design of a mother wavelet best adapted to a specific signal or to a class of signals. The filter's coefficients are obtained via optimization of a smooth objective function. We develop an unconstrained gradient-based optimization algorithm for a discrete wavelet transform. The algorithm is extended to the joint optimization of the mother wavelet and of the wavelet packets basis.

INTRODUCTION

The general problem we are trying to address is to find an invertible linear transform $\mathcal{L}$ that minimizes an objective function ϕ for a specific signal. For a class of signals, the objective function is redefined to compute some statistics from the collection of data. We concentrate on the class of linear transforms known as discrete wavelet transforms.

In is paper we describe a novel optimization for the purpose of wavelet filter decomposition or of more general basis function decompositions based on wavelet packets. The optimization is based on the lattice decomposition of filter banks and leads to a fast unconstrained algorithm.

FORMULATION OF THE DISCRETE WAVELET TRANSFORM

In the discrete wavelet transform, a low-pass and a high-pass filter are applied to the signal, and the output is downsampled by two. The high-pass coefficients are retained, while the process is repeated on the low-pass coefficients, until the length of the residual signal's coefficients equals that of the filter. In order for this transform to be invertible, the filters have to satisfy some constrains. In particular, orthonormality is required to obtain an orthonormal basis. These constrains can be expressed in different forms.

0-7803-7616-1/02/$17.00 (c) 2002 IEEE

Exact details of the various forms can be found in chapter 5 of [4]. Here, in particular, we use two formulations: 1) the time-domain method and 2) the lattice method. The first approach expresses the constrains directly on the filters' coefficients. This leads to a constrained optimization algorithm. The second approach is based on the lattice structure method. With this method it is possible to reparametrize the coefficients so that the constrains are automatically satisfied. This leads to an unconstrained optimization algorithm.

Without loss of generality, we concentrate on wavelet transforms with periodic boundaries. What follows can be extended to wavelets with different boundary conditions.

The time-domain method

Consider an example which demonstrates how a discrete wavelet transform is computed and how some constrains on the filter's coefficients arise from the orthogonality condition. In what follows, the length of the signals will be a power of 2. The case of a filter of length 4 acting on a signal of length 8 is described in detail, as well as generalization to filters of even length. We use a notation from [4], where signals and filters are indexed from 0 to N, so that the total length is $N+1$.

Let $\vec{x} = [x_0, \ldots, x_7]^T$ be the original signal, $c_0, \ldots c_3$ and $d_0, \ldots d_3$ the low-pass and high-pass filters' coefficients respectively. After applying the two filters to $\vec{x}$, the result is decimated by two. These two operations, application of the filters and decimation, can be done more efficiently in a single step. In matrix form they are equivalent to the following:

$$[\sigma_0, \sigma_1, \sigma_2, \sigma_3, \delta_0, \delta_1, \delta_2, \delta_3]^T = C_1 \vec{x}, \tag{1}$$

where

$$C_1 = \begin{bmatrix} c_3 & c_2 & c_1 & c_0 & 0 & 0 & 0 & 0 \\ 0 & 0 & c_3 & c_2 & c_1 & c_0 & 0 & 0 \\ 0 & 0 & 0 & 0 & c_3 & c_2 & c_1 & c_0 \\ c_1 & c_0 & 0 & 0 & 0 & 0 & c_3 & c_2 \\ d_3 & d_2 & d_1 & d_0 & 0 & 0 & 0 & 0 \\ 0 & 0 & d_3 & d_2 & d_1 & d_0 & 0 & 0 \\ 0 & 0 & 0 & 0 & d_3 & d_2 & d_1 & d_0 \\ d_1 & d_0 & 0 & 0 & 0 & 0 & d_3 & d_2 \end{bmatrix}. \tag{2}$$

Then the same process is repeated just on the s's, the low-pass coefficients:

$$\begin{aligned} &[\Sigma_0, \Sigma_1, \Delta_0, \Delta_1, \delta_0, \delta_1, \delta_2, \delta_3]^T \\ &= C_2 [\sigma_0, \sigma_1, \sigma_2, \sigma_3, \delta_0, \delta_1, \delta_2, \delta_3]^T \end{aligned} \tag{3}$$

where

$$C_2 = \begin{bmatrix} c_3 & c_2 & c_1 & c_0 & 0 & 0 & 0 & 0 \\ c_2 & c_0 & c_3 & c_1 & 0 & 0 & 0 & 0 \\ d_3 & d_2 & d_1 & d_0 & 0 & 0 & 0 & 0 \\ d_1 & d_0 & d_3 & d_2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{4}$$

Combining the two steps together, the transformed signal $\vec{y}$ is given by:

$$\vec{y} = C\vec{x} = C_2 C_1 \vec{x}, \tag{5}$$

It is easy to see that when C_1 is orthonormal, C_2 and hence C are orthonormal. The condition for orthonormality are:

$$\begin{aligned} &c_0^2 + c_1^2 + c_2^2 + c_3^2 = 1, \\ &c_0 c_2 + c_1 c_3 = 0, \end{aligned} \tag{6}$$

$$\begin{aligned} &d_0^2 + d_1^2 + d_2^2 + d_3^2 = 1, \\ &d_0 d_2 + d_1 d_3 = 0, \end{aligned} \tag{7}$$

$$\begin{aligned} &c_0 d_0 + c_1 d_1 + c_2 d_2 + c_3 d_3 = 0, \\ &c_0 d_2 + c_1 d_3 = 0, \\ &c_0 d_0 + c_3 d_1 = 0, \end{aligned} \tag{8}$$

If the c's satisfy (6), it is possible to solve all the equations in (7) and (8) with the following choice:

$$d_k = (-1)^k c_{3-k} \, , k = 0, \ldots, 3. \tag{9}$$

In general, a filter of even length $N+1$, acts on a signal of length M+1, where M+1 is a power of 2. The transform is given by:

$$\vec{y} = C\vec{x}, \tag{10}$$

where the matrix C is the product of Q orthonormal matrices:

$$C = C_Q C_{Q-1} \cdots C_2 C_1 \, , \quad Q \leq Q_{\max} \, , \tag{11}$$

where $Q_{\max}$, the maximum number of decomposition levels allowed, depends on both the length of the signal and that of the filter (MATLAB notation):

$$Q_{\max} = \text{floor}\left(\log_2 \frac{M+1}{N+1}\right) \, . \tag{12}$$

The orthonormality conditions can then be expressed in a compact form:

- Conditions on the c's:

$$\sum_{n=2k}^{N} c_n c_{n-2k} = \delta(k)\ , k = 0, \dots, (N-1)/2 \tag{13}$$

- Conditions on the d's:

$$\sum_{n=2k}^{N} d_n d_{n-2k} = \delta(k)\ , k = 0, \dots, (N-1)/2 \tag{14}$$

- Relations between the c's and the d's:

$$\begin{array}{rcll} \sum_{n=2k}^{N} c_n d_{n-2k} & = & \delta(k) & , k = 0, \dots, (N-1)/2 \\ \sum_{n=2k}^{N} c_{n-2k} d_n & = & \delta(k) & , k = 0, \dots, (N-1)/2 \end{array} \tag{15}$$

The high-pass coefficients can be computed from the low-pass ones just like in (9):

$$d_k = (-1)^k c_{N-k}\ , k = 0, \dots, N. \tag{16}$$

With this choice, if (13) is satisfied, then (14) and (15) are.

The lattice method: reparametrization of the filter's coefficients

It is possible to reparametrize the coefficients $c_0, \dots, c_N$ so that the constrains in (13,14,15) are automatically satisfied. Starting from the simple case of 4 coefficients, the constrains in (6) imply that:

$$[c_0 + c_2]^2 + [c_1 + c_3]^2 = 1, \tag{17}$$

which is automatically satisfied setting

$$\left\{ \begin{array}{rcl} c_0 & = & \cos\theta_1 \cos\theta_2 \\ c_1 & = & \cos\theta_1 \sin\theta_2 \\ c_2 & = & -\sin\theta_1 \sin\theta_2 \\ c_3 & = & \sin\theta_1 \cos\theta_2 \end{array} \right. \tag{18}$$

It would be natural to extend this procedure to the general case. In fact, the orthonormality conditions imply that:

$$\left(\sum_{n \text{ even}} c_n \right)^2 + \left(\sum_{n \text{ odd}} c_n \right)^2 = 1, \tag{19}$$

so that we can set

$$\sum_{n=0}^{(N-1)/2} c_{2n} = \cos\left(\sum_{n=0}^{(N-1)/2} \theta_n\right), \sum_{n=0}^{(N-1)/2} c_{2n+1} = \sin\left(\sum_{n=0}^{(N-1)/2} \theta_n\right) \tag{20}$$

The right-hand side in both the above equations contains $(N+1)/2$ terms, but the expansions of the left-hand sides, obtained using the generalized trigonometric addition formula, contain $2^{(N-1)/2}$ terms. Distributing the trigonometric monomials to the various filter's coefficients is not straightforward. However, this problem can be solved in a systematic and very elegant way using the lattice factorization from the theory of filter banks. In fact, the polyphase matrix of any two channel orthogonal filter bank can be factorized as [4] [5] [6]:

$$H_p^{(K)} = \rho(\theta_1)\Lambda(z)\rho(\theta_2)\cdots\Lambda(z)\rho(\theta_K), \Lambda(z) = \begin{bmatrix} 1 & 0 \\ 0 & z^{-1} \end{bmatrix}, \tag{21}$$

and $\rho(\theta) \in O(2)$.

The factorization of the polyphase matrix leads naturally to a factorization of the wavelet transform in the time domain. This factorization process is illustrated below for a filter having six coefficients operating on a signal of length eight. Although illustrated for this example, the process can be extended to filters of even length and signals of arbitrary length.

Equation (21) can be rewritten as:

$$H_p^{(K)} = H_p^{(K-1)}\Lambda(z)\rho(\theta_K) \tag{22}$$

which links the coefficients of a filter of length $2K$ to those of a filter of length $2(K-1)$, and allows an iterative procedure to build a filter of arbitrary lenght. In particular, the polyphase matrices for filters of length six, four and two are given by:

$$\begin{aligned} H_p^{(3)} &= \begin{bmatrix} c_0^{(3)} + z^{-1}c_2^{(3)} + z^{-2}c_4^{(3)} & c_1^{(3)} + z^{-1}c_3^{(3)} + z^{-2}c_5^{(3)} \\ d_0^{(3)} + z^{-1}d_2^{(3)} + z^{-2}d_4^{(3)} & d_1^{(3)} + z^{-1}d_3^{(3)} + z^{-2}d_5^{(3)} \end{bmatrix} \\ H_p^{(2)} &= \begin{bmatrix} c_0^{(2)} + z^{-1}c_2^{(2)} & c_1^{(2)} + z^{-1}c_3^{(2)} \\ d_0^{(2)} + z^{-1}d_2^{(2)} & d_1^{(2)} + z^{-1}d_3^{(2)} \end{bmatrix}, H_p^{(1)} = \begin{bmatrix} c_0^{(1)} & c_1^{(1)} \\ d_0^{(1)} & d_1^{(1)} \end{bmatrix} \end{aligned} \tag{23}$$

where the othonormality is assured for $H_p^{(1)}$ and all polyphase matrices derived from it when

$$H_p^{(1)} = \rho(\theta_1) = \begin{bmatrix} \cos\theta_1 & \sin\theta_1 \\ -\sin\theta_1 & \cos\theta_1 \end{bmatrix} \tag{24}$$

Thus from equation (22), $H_p^{(2)}$ is given by $H_p^{(2)} = H_p^{(1)}\Lambda(z)\rho(\theta_2)$, which can be written as:

$$\begin{bmatrix} c_0^{(2)} + z^{-1}c_2^{(2)} & c_1^{(2)} + z^{-1}c_3^{(2)} \\ d_0^{(2)} + z^{-1}d_2^{(2)} & d_1^{(2)} + z^{-1}d_3^{(2)} \end{bmatrix} =$$

$$\begin{bmatrix} \cos\theta_1 & \sin\theta_1 \\ -\sin\theta_1 & \cos(\theta_1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & z^{-1} \end{bmatrix} \begin{bmatrix} \cos\theta_2 & \sin\theta_2 \\ -\sin\theta_2 & \cos\theta_2 \end{bmatrix} \tag{25}$$

Multiplying the right-hand side and solving for the value of the various coefficients on the left-hand side can be determined using the reparameterized coefficients directly by equating powers of z. This leads to:

$$\begin{aligned} c_0^{(2)} &= \cos\theta_1 \cos\theta_2 = c_0^{(1)} \cos\theta_2 \\ c_1^{(2)} &= \cos\theta_1 \sin\theta_2 = c_0^{(1)} \sin\theta_2 \\ c_2^{(2)} &= -\sin\theta_1 \sin\theta_2 = -c_1^{(1)} \sin\theta_2 \\ c_3^{(2)} &= \sin\theta_1 \cos\theta_2 = c_1^{(1)} \cos\theta_2 \end{aligned} \tag{26}$$

and similarly for the d's. These relationships can be rewritten in matrix form as:

$$\begin{bmatrix} c_3^{(2)} \\ c_2^{(2)} \\ c_1^{(2)} \\ c_0^{(2)} \end{bmatrix} = \begin{bmatrix} \cos\theta_2 & 0 \\ -\sin\theta_2 & 0 \\ 0 & \sin\theta_2 \\ 0 & \cos\theta_2 \end{bmatrix} \begin{bmatrix} c_1^{(1)} \\ c_0^{(1)} \end{bmatrix} \tag{27}$$

As noted above in equation (22), the next polyphase matrix $H_p^{(3)}$ can be found as a function of $H_p^{(2)}$. Multiplying the matrices and equating like powers of z as above yields:

$$\begin{bmatrix} c_5^{(3)} \\ c_4^{(3)} \\ c_3^{(3)} \\ c_2^{(3)} \\ c_1^{(3)} \\ c_0^{(3)} \end{bmatrix} = \begin{bmatrix} k_3 & 0 & 0 & 0 \\ -s_3 & 0 & 0 & 0 \\ 0 & s_3 & k_3 & 0 \\ 0 & k_3 & -s_3 & 0 \\ 0 & 0 & 0 & s_3 \\ 0 & 0 & 0 & k_3 \end{bmatrix} \begin{bmatrix} c_3^{(2)} \\ c_2^{(3)} \\ c_1^{(2)} \\ c_0^{(2)} \end{bmatrix} \tag{28}$$

where $s_j = \sin\theta_j$, $k_j = \cos\theta_j$. Substituting equation (27) in (28) yields

$$\begin{bmatrix} c_5^{(3)} \\ c_4^{(3)} \\ c_3^{(3)} \\ c_2^{(3)} \\ c_1^{(3)} \\ c_0^{(3)} \end{bmatrix} = \begin{bmatrix} k_3 & 0 & 0 & 0 \\ -s_3 & 0 & 0 & 0 \\ 0 & s_3 & k_3 & 0 \\ 0 & k_3 & -k_3 & 0 \\ 0 & 0 & 0 & s_3 \\ 0 & 0 & 0 & k_3 \end{bmatrix} \begin{bmatrix} k_2 & 0 \\ -s_2 & 0 \\ 0 & s_2 \\ 0 & k_2 \end{bmatrix} \begin{bmatrix} s_1 \\ k_1 \end{bmatrix} \tag{29}$$

A similar expression hold for the ds, so that combining the two together we have

$$\begin{bmatrix} c_5^{(3)} & c_4^{(3)} & c_3^{(3)} & c_2^{(3)} & c_1^{(3)} & c_0^{(3)} \\ d_5^{(3)} & d_4^{(3)} & d_3^{(3)} & d_2^{(3)} & d_1^{(3)} & d_0^{(3)} \end{bmatrix} =$$

$$\begin{bmatrix} s_1 & k_1 \\ k_1 & -sta_1 \end{bmatrix} \begin{bmatrix} k_2 & -s_2 & 0 & 0 \\ 0 & 0 & s_2 & k_2 \end{bmatrix} \begin{bmatrix} k_3 & -s_3 & 0 & 0 & 0 & 0 \\ 0 & 0 & s_3 & k_3 & 0 & 0 \\ 0 & 0 & k_3 & -s_3 & 0 & 0 \\ 0 & 0 & 0 & 0 & s_3 & k_3 \end{bmatrix} \tag{30}$$

Thus, each coefficient can be found as a function of the reparameterized angle coefficients. Performing the matrix multiplication in equation (35) defines each coefficient as a linear combination of two or more trigonometric functions. Plugging (30) into the matrix C_1 that performs the first step in the wavelet cascade of filters we obtain the following factorization:

$$C_1 = ER(\theta_1)\,SR(\theta_2)\,SR(\theta_3) \tag{31}$$

where

$$R(\theta_j) = \begin{bmatrix} s_j & k_j & 0 & 0 & 0 & 0 & 0 & 0 \\ k_j & -s_j & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & s_j & k_j & 0 & 0 & 0 & 0 \\ 0 & 0 & k_j & -s_j & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & s_j & k_j & 0 & 0 \\ 0 & 0 & 0 & 0 & k_j & -s_j & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & s_j & k_j \\ 0 & 0 & 0 & 0 & 0 & 0 & k_j & -s_j \end{bmatrix} \tag{32}$$

$$S = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{33}$$

and E separates the high-pass coefficients from the low-pass ones. Notice that $R(\theta_j)$ are rotation matrices and S is an up-shift matrix. In general, for a filter of length $2K$ we have:

$$C_1 = ER(\theta_1)SR(\theta_2)S\cdots SR(\theta_K) \tag{34}$$

The wavelet transform formulation in terms of a product of matrices can be easily extended to compute any basis in a wavelet packets table [1] [7]. Expression (11) still holds: $C_l, 1 \le l \le Q$ is a block matrix; each block corresponds to either the identity, if that component of the signal is not decomposed any further, or a wavelet matrix. Equation (34) provides a factorization for a single step in the wavelet transform. Putting all the pieces together we obtain a factorization for a generic wavelet packets basis:

$$C = \underbrace{E_Q R_Q^{(1)} S_Q R_Q^{(2)} S_Q \cdots S_Q R_Q^{(K)}}_{C_Q} \cdots \underbrace{E_1 R_1^{(1)} S_1 R_1^{(2)} S_1 \cdots S_1 R_1^{(K)}}_{C_1} \tag{35}$$

The library of wavelet packet bases can be searched to find the best basis according to some criterion, for example to minimize an objective function. Coifman et al. [1] showed that, if the objective function is additive a divide and conquer formulation can be applied to the search, so that a library of 2^n bases can be searched in $n\log(n)$. It is then possible to perform a joint search of the best basis and best mother wavelet. This can be obtained for example through an iterative process that alternates the two oprimizations.

OPTIMIZATION

The formulation given in (35) is very natural for a gradient optimization. It thus serves our goal to find the optimal parameterization of the transform given a signal or a class of signals. This is obtained by minimizing an appropriate objective function. A general procedure for its minimization based on the computation of the derivatives with respect to the parameters is given below. Due to the linearity of the transform, these derivatives can be expressed in a simple form. The gradient of the objective function with respect to the parameters of the transform is $\nabla_p \phi = \mathbf{J}_p^T \nabla_y \phi$, where $\mathbf{J}_p$ is the Jacobian of $\mathcal{L}$. For $\mathcal{L}_{\vec{p}}[\vec{x}] = C\vec{x}$

$$\mathbf{J}_p = \left[(\partial_{p_1} C)\,\vec{x}, (\partial_{p_2} C)\,\vec{x}, \ldots, (\partial_{p_q} C)\,\vec{x} \right] \;. \tag{36}$$

An explicit form for the derivatives with respect to the lattice parameters can be obtained through equation (35):

$$\begin{aligned} &\partial_j C(\theta_1, \ldots, \theta_K) = \\ =\;& E R_Q(\theta_1) S_Q \cdots D_Q R_Q(\theta_j) \cdots S_1 R_1(\theta_K) \quad + \\ +\;& E R_Q(\theta_1) S_Q \cdots\cdots D_{Q-1} R_{Q-1}(\theta_j) \cdots S_1 R_1(\theta_K) \quad + \\ &\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\ +\;& E R_Q(\theta_1) S_Q \cdots\cdots\cdots\cdots D_1 R_1(\theta_j) \cdots S_1 R_1(\theta_K) \end{aligned} \tag{37}$$

where D_l is a block-diagonal matrix whose block are equal to $\begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$. With the appropriate choice of the matrices $C_1, \ldots, C_Q$, the above expression allows the computation of the gradient with respect to the lattice angles for any basis in the wavelet packets table. It is then possible to combine a best basis algorithm with the optimization of the filters' coefficients in various ways (e.g. iterative two steps optimization). Moreover, given a basis in the wavelet packet table, the computation of the transform and of it's gradient with respect to the lattice angles can procede in parallel.

It is possible to add additional contrains to the filter's coefficients – for example a certain number of vanishing moments can be required – but in general this leads to a constrained optimization. However, it is possible to impose a zero-mean conidition to the high-pass filter at very little cost, and recast the optimization into an unconstrained one.

DISCUSSION

We have shown that it is possible to find an optimal mother wavelet with respect to a given objective function through an unconstrained optimization over a compact manifold, which guaranties the existence of a minimum. We explicitely reparametrized the filters' coefficients using the lattice decomposition so that the orthogonality constrains are automatically satisfied, and derived an expression for the gradient of the objective function with respect to the lattice angles. Any gradient based optimization can then be used to find the optimal solution. We showed that the framework we have developed is not constrained to wavelet transforms, but can be extended directly to a generic wavelet packets' basis. It is then possible to optimize the specific basis and the filters jointly.

Wavelet design is usually based on the optimization of wavelets according to some property of the filter itself, such as stopband attenuation, coding gain, or degree of smoothness. It has been shown that for some specific objective function it is convenient to optimize the filters coefficients in the time-domain [3]. This was possible because the optimization problem could be cast into a quadratic-constrained least-squares minimization one; moreover, the optimization was performed on a single step of the transform.

The constrained optimization leads to a polynomial with a degree proportional to the length of the filter and of the number of decomposition steps, which is in general linked to the length of the filter. Thus, realistic filters cannot be practically optimized due to the roughness of the polynomial error surface.

In our approach, the design is driven by the data themselves and the whole cascade of filters is taken into account. Preliminary work shows that there exist classes of signals for which the optimal solution is far from the classical wavelets developed with the above methods.

Furthermore, starting from the lattice formulation, it is simple to decompose the transform into lifting steps. In fact, every Givens rotation in the polyphase matrix (21) can be decomposed into three lifting steps [2]. Thus, once an optimal set of parameters are obtained through optimization, the corresponding lifting implementation can be used, making it possible to exploit all the advantages of the latter formulation.

Acknowledgments

We would like to thank Raphi Coifman, Leon Cooper, Quyen Huynh and Truong Nguyen for many fruitful discussions. This work was partially supported by DARPA/ONR grants N00014-99-1-0009 and N00014-01-1-0306.

REFERENCES

[1] R. R. Coifman and M. V. Wickerhauser, "Entropy-Based Algorithms for Best Basis Selection," **IEEE Transactions on Information Theory**, vol. 38, no. 2, pp. 713–718, 1992.

[2] I. Daubechies and W. Sweldens, "Factoring wavelet transforms into lifting steps," **The Journal of Fourier Analysis and Applications**, vol. 4, no. 3, pp. 247–269, 1998.

[3] T. Q. Nguyen, "Digital Filter Bank Design – Quadratic-Constrained Formulation," **IEEE Transactions on Signal Processing**, vol. 43, no. 9, pp. 2103–2108, 1995.

[4] G. Strang and T. Nguyen, **Wavelets and Filter Banks**, Wellesley, MA: Wellesley-Cambridge Press, 1996.

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

[6] P. P. Vaidyanathan and P. Q. Hoang, "Lattice Structures for Optimal Design and Robust Implementation of Two-Channel Perfect-Reconstruction QMF Banks," **IEEE Transactions on Acoustic, Speech, and Signal Processing**, vol. 36, no. 1, pp. 81–94, 1988.

[7] M. V. Wickerhauser, **Adapted Wavelet Analysis from Theory to Software**, Wellesley, Massachusetts: A K Peters, 1994.