Available online at www.sciencedirect.com

SCIENCE DIRECT®

Journal of Computational and Applied Mathematics 189 (2006) 606–621

JOURNAL OF COMPUTATIONAL AND APPLIED MATHEMATICS

www.elsevier.com/locate/cam

Image denoising using principal component analysis in the wavelet domain

Silvia Bacchelli*, Serena Papi

Department of Mathematics, University of Bologna, via Sacchi 3, 47023 Cesena, Italy

Received 30 September 2004

Abstract

In this work we describe a method for removing Gaussian noise from digital images, based on the combination of the wavelet packet transform and the principal component analysis. In particular, since the aim of denoising is to retain the energy of the signal while discarding the energy of the noise, our basic idea is to construct powerful tailored filters by applying the Karhunen–Loéve transform in the wavelet packet domain, thus obtaining a compaction of the signal energy into a few principal components, while the noise is spread over all the transformed coefficients. This allows us to act with a suitable shrinkage function on these new coefficients, removing the noise without blurring the edges and the important characteristics of the images. The results of a large numerical experimentation encourage us to keep going in this direction with our studies.

MSC: 65D; 65Y20; 65F

Keywords: Wavelet packets; KL transform; Filter banks; Recursive matrices; Image denoising

1. Introduction

Signals and images are often corrupted by noise in their acquisition or transmission. Let the noise model be

$$\bar{f} = f + \xi,$$

* Corresponding author. Tel.: +39 0547338875; fax: +39 0547338890.
E-mail addresses: silvia@csr.unibo.it (S. Bacchelli), spapi@csr.unibo.it (S. Papi).

0377-0427/$ - see front matter © 2005 Elsevier B.V. All rights reserved.
doi:10.1016/j.cam.2005.04.030

where $\bar{f}$ is the noisy image, f is the original image and ξ is i.i.d. Gaussian noise with mean zero and standard deviation σ. The goal of a denoising method is to remove the noise, while retaining the important signal features as much as possible. Traditionally, in the existing literature there are two kinds of denoising methods; namely, linear and nonlinear techniques. Linear denoising methods, such as Wiener filtering [14], are simple and cheap to implement. However, they sometimes tend to blur the proper edge structure of an image, which determines a good visual quality. In order to preserve the real characteristics of a signal, a vast literature has recently emerged on signal and image denoising using nonlinear denoising techniques, such as, e.g., the well-known *Wavelet Shrinkage* introduced in [11]. This method is based on filtering the wavelet coefficients in the frequency detail subbands by means of the *Soft Thresholding* operator, which takes the coefficients of absolute value larger than the threshold and shrinks them by the threshold value towards zero.

More precisely, the wavelet expansion of $\bar{f} \in L_2(\mathbb{R}^2)$ is considered, namely

$$\bar{f} = \sum_{j\in\mathbb{Z}^2,\ k\in\mathbb{Z},\ \psi\in\Psi} \bar{c}_{j,k,\psi}\psi_{j,k} \quad \text{with } \bar{c}_{j,k,\psi} = \int_{\mathbb{R}^2} \bar{f}(x)\psi_{j,k}(x)\,\mathrm{d}x,$$

where the functions $\psi \in \Psi$ are the two-dimensional wavelets, constructed via the tensor product of one-dimensional wavelet and scaling functions [8,10]. Hence, the Soft Thresholding filter acts on the noisy wavelet coefficients $\bar{c}_{j,k,\psi}$ in the following way:

$$S_\lambda(\bar{c}_{j,k,\psi}) = \begin{cases} \operatorname{sign}(\bar{c}_{j,k,\psi})(|\bar{c}_{j,k,\psi}| - \lambda), & |\bar{c}_{j,k,\psi}| > \lambda, \\ 0, & |\bar{c}_{j,k,\psi}| \leqslant \lambda. \end{cases}$$

The wavelet expansion of the denoised image is therefore given by

$$f^\star = \sum_{j\in\mathbb{Z}^2,\ k\in\mathbb{Z},\ \psi\in\Psi} S_\lambda(\bar{c}_{j,k,\psi})\psi_{j,k}.$$

In recent years, many papers have been devoted to the wavelet denoising problem, by proposing many alternative choices both for the thresholding rule and the shrinkage parameter λ (see, for example, [2,15,13,16]). In particular, in [16], a first attempt has been proposed to combine linear Wiener filtering and wavelet thresholding in an unique nonlinear denoising method. The idea discussed in [16] is to estimate the covariance matrix of each block of wavelet coefficients and to use its eigenvalues to obtain a more efficient thresholding rule.

On the other hand, an interesting aspect of wavelet thresholding algorithms is that they can lead to lossy compression of the starting data. Actually, in the existing literature, many works addressed a strong connection between lossy compression and denoising, especially with nonlinear algorithms (see [6,7]). Therefore, a good compression method can provide a suitable model for distinguishing between signal and noise.

In this paper, we exploit the analogy between denoising and lossy compression problems in order to introduce a new denoising method, which combines the good properties of the wavelet packet analysis with those of principal component analysis, realized using the Karhunen–Loéve (KL) transform in the wavelet domain. In fact, the KL transform is a well-established data-dependent tool for image and signal compression, but its great computational complexity highly reduces its field of application. Our idea is to evaluate the covariance matrix of the wavelet packet coefficients and to exploit the magnitude of its eigenvalues to make a decision about the amount of information contained in each coefficient. In fact, the

application of the KL transform in the wavelet packet domain leads to completely decorrelated frequency subbands, where the energy of the signal is clustered in few large coefficients, while the noise pollutes all the transformed coefficients by a small amount. An eigenvalue-based shrinkage function in the wavelet-KL domain can therefore exploit the possibilities provided by this new data-dependent filtering operator and hence leads to a more efficient denoising algorithm which removes the noise without sacrificing the important signal structures.

The mathematical derivation of this filtering operator is realized using the algebraic framework of recursive matrices, since it represents a flexible tool which easily handles the basic operation of signal processing. This allows us to interpret our operator as a data-dependent filter bank, hence exploiting all the mathematical results concerning this subject.

The paper is organized as follows: in Section 2 we give some algebraic preliminaries and the basic notions on recursive matrices, and in Section 3 the general theory of orthogonal wavelet packet filter banks is presented. Sections 4 and 5 are devoted to the algebraic interpretation of the KL transform and to the description of the proposed orthogonal wavelet packet-KL filter bank. Finally, in Section 6 we describe the use of the newly constructed filtering operator for image denoising, and we describe our numerical results.

2. Algebraic preliminaries and notations: basics on recursive matrices

We here briefly introduce some basic concepts on Laurent polynomials and recursive matrices, that will be useful in the rest of the paper. A *Laurent polynomial* in a fixed field K is a formal polynomial

$$\alpha = \sum_{i=p}^{q} \langle i|\alpha\rangle t^i = \sum_{i=p}^{q} a_i t^i,$$

where t is a formal variable. As usual, given two Laurent polynomials $\alpha = \sum_{i=p}^{q} a_i t^i$ and $\beta = \sum_{j=r}^{s} b_j t^j$, the composition $\beta \circ \alpha$ is defined as the following Laurent polynomial:

$$\beta \circ \alpha = \sum_{j=r}^{s} b_j \left(\sum_{i=p}^{q} a_i t^i \right)^j.$$

Now, given a fixed integer $k \geqslant 2$, the *k-decimated polynomials* relative to α are the k polynomials $D_0^k(\alpha), D_1^k(\alpha), \ldots, D_{k-1}^k(\alpha)$ defined, for $r = 0, 1, \ldots, k-1$, as follows:

$$\alpha_r^k = D_r^k(\alpha) = \sum_i a_{ki+r} t^i,$$

where the symbol D_r^k represents the decimation operator. Easy computations show that any Laurent polynomial can be recovered from its k-decimated polynomials in the following way [3]:

$$\alpha = \alpha_0^k \circ t^k + \cdots + t^r(\alpha_r^k \circ t^k) + \cdots + t^{(k-1)}(\alpha_{k-1}^k \circ t^k). \tag{1}$$

We recall that if $\alpha = \sum_{i=p}^{q} a_i t^i$ is a Laurent polynomial, the stared polynomial is defined as

$$\alpha^* = \sum_{i=p}^{q} a_i t^{-i}.$$

Laurent polynomials are often associated with vectors. In the following we will denote with $\boldsymbol{\alpha}$ the vector whose entries are given by the sequence $\{a_i\}_{i=p}^{q}$, defining the Laurent polynomial α. Laurent polynomials are useful to define *recursive matrices*. Briefly, recursive matrices are bi-infinite matrices such that each row can be recursively computed from the previous one. This kind of matrix has recently been revealed to be a powerful tool to represent the basic operations of filter theory. In particular, recursive Toeplitz and Hurwitz matrices are used in [5] to give an algebraic interpretation of signal processing. More precisely, given two nonzero Laurent polynomials α and β, the (α, β)-recursive matrix is the unique matrix B such that its ith row $B(i)$ is obtained as

$$B(i) = \alpha^i \beta \quad \text{for every integer } i.$$

The matrix B will be denoted by the symbol $R(\alpha, \beta)$, and the polynomials α and β are called the recurrence rule and the generating function of B, respectively. For a more complete description of this topic, the reader is referred to [3]. In practical applications, we frequently deal with recursive matrices whose recurrence rule is a monomial. More precisely, given a Laurent polynomial $\pi = \sum_{i=p}^{q} p_i t^i$, we define a Toeplitz matrix with generating function π the recursive matrix $R(t, \pi) = [p_{k,i}]$, where $p_{k,i} = p_{i-k}$. Moreover, we define a Hurwitz matrix of step $h \geqslant 2$ and generating function π the recursive matrix $R(t^h, \pi) = [p_{k,i}]$, where $p_{k,i} = p_{i-hk}$.

The main property of recursive matrices is the fact that the product of two recursive matrices, under suitable conditions, can always be performed and it is again a recursive matrix, which can be explicitly described. In fact, we have [3]:

Theorem 2.1. *Let K be a field, and $\alpha, \beta, \gamma, \delta$ be nonzero Laurent polynomials. Then*

$$R(\alpha, \beta) \times R(\gamma, \delta) = R(\alpha \circ \gamma, (\beta \circ \gamma)\delta).$$

We now recall some results on Toeplitz and Hurwitz matrices and their transposes, that will be useful in developing the theory of the next sections.

Proposition 2.1. *The products of Toeplitz and Hurwitz matrices are Hurwitz matrices*; *more precisely*:

$$R(t, \alpha) \times R(t^k, \beta) = R(t^k, (\alpha \circ t^k)\beta),$$

$$R(t^k, \beta) \times R(t, \alpha) = R(t^k, \beta\alpha).$$

Proposition 2.2. *The product of two Hurwitz matrices is again a Hurwitz matrix*, *with a different step*:

$$R(t^h, \alpha) \times R(t^k, \beta) = R(t^{hk}, (\alpha \circ t^k)\beta).$$

Proposition 2.3. *Let* $\alpha = \sum_{i=p}^{q} a_i t^i$ *be a Laurent polynomial*; *the k-upsampling of* α, *defined as*

$$(\uparrow k)\alpha = \alpha \circ t^k = \sum_{i=p}^{q} a_i t^{ki},$$

is equivalent to the following product:

$$(\uparrow k)\alpha = R(t^k, 1)^{\mathrm{T}} \boldsymbol{\alpha}.$$

Proposition 2.4. *Let* $\alpha = \sum_{i=p}^{q} a_i t^i$ *be a Laurent polynomial*; *the k-positions shift of* α, *defined as*

$$t^k \alpha = \sum_{i=p}^{q} a_i t^{i+k},$$

is equivalent to the following product:

$$t^k \alpha = R(t, t^{-k})\boldsymbol{\alpha}.$$

Theorem 2.2. *Let* α *and* β *be nonzero Laurent polynomials and k be a fixed integer*, $k \geqslant 2$. *Set* $M = R(t^k, \alpha)$, $N = R(t^k, \beta)$. *Then the product* $M^{\mathrm{T}} \times N$ *is a block Toeplitz matrix*, *whose generating function is the block Laurent polynomial* $\sum_i \mathbf{C}_i t^i$, *where* $\mathbf{C}_i$ *is a* $(k \times k)$ *matrix*, *namely*

$$\mathbf{C}_i = \begin{bmatrix} \langle i | \alpha_0^{k*} \beta_0^k \rangle & \dots & \langle i | \alpha_0^{k*} \beta_{k-1}^k \rangle \\ \vdots & & \vdots \\ \langle i | \alpha_{k-1}^{k*} \beta_0^k \rangle & \dots & \langle i | \alpha_{k-1}^{k*} \beta_{k-1}^k \rangle \end{bmatrix}.$$

For more details, the reader is referred to [3,5,1].

3. Orthogonal wavelet packet filter banks

Let us consider an orthonormal multiresolution analysis of $L_2(\mathbb{R})$, whose generating scaling function $\phi(x)$ and the corresponding wavelet $\psi(x)$ [10,9] satisfy the well-known two-scale equations

$$\Phi(x) = H\Phi(2x),$$

$$\Psi(x) = W\Phi(2x),$$

where

$$\Phi(x) = \begin{bmatrix} \dots \\ \phi(x+1) \\ \phi(x) \\ \phi(x-1) \\ \dots \end{bmatrix}, \quad \Psi(x) = \begin{bmatrix} \dots \\ \psi(x+1) \\ \psi(x) \\ \psi(x-1) \\ \dots \end{bmatrix}$$

and the matrices H and W are Hurwitz matrices, namely,

$$H = R(t^2, \alpha_0) = (h_{i,j})_{i,j\in\mathbb{Z}} = ((\alpha_0)_{j-2i})_{i,j\in\mathbb{Z}} \quad \text{and} \quad W = R(t^2, \alpha_1) = (w_{i,j})_{i,j\in\mathbb{Z}} = ((\alpha_1)_{j-2i})_{i,j\in\mathbb{Z}}.$$

Their generating functions $\alpha_0 = \sum_i (\alpha_0)_i t^i$ and $\alpha_1 = \sum_i (\alpha_1)_i t^i$ are the low-pass and high-pass filters, respectively, characterizing the multiresolution analysis.

An orthonormal wavelet-packet basis [8] is obtained by setting

$$\begin{aligned} &\psi_0(x) = \phi(x), \\ &\psi_1(x) = \psi(x) \end{aligned} \tag{2}$$

and by defining the other functions of the basis by means of the two-scale equations as follows:

$$\begin{cases} \Psi_{2l}(x) = H\Psi_l(2x), \\ \Psi_{2l+1}(x) = W\Psi_l(2x). \end{cases} \qquad l = 0, 1, \ldots \tag{3}$$

The family wavelet packet $\psi_l(x)$, $l \in \mathbb{Z}_+$, so obtained, preserves the orthogonality property of the scaling function $\psi_0(x) = \phi(x)$, and represent an orthonormal basis of $L_2(\mathbb{R})$. It follows that each finite energy signal σ can be decomposed into its wavelet-packet components.

By using the recursive matrix machinery it is easy to show how L wavelet-packet decomposition/reconstruction steps can be seen as a 2^L-channel analysis/synthesis filter bank.

In fact, in [3] it is shown that a filtering operation followed by decimation is equivalent to multiply the input signal σ by a Hurwitz matrix. Therefore, L wavelet packet decomposition steps of σ can be represented as follows:

$$\begin{aligned} &\boldsymbol{\sigma}_0 = F_0\boldsymbol{\sigma} = R(t^{2^L}, h_0)\boldsymbol{\sigma} \\ &\boldsymbol{\sigma}_1 = F_1\boldsymbol{\sigma} = R(t^{2^L}, h_1)\boldsymbol{\sigma} \\ &\quad\vdots \qquad\quad \vdots \qquad\qquad \vdots \\ &\boldsymbol{\sigma}_{2^L-1} = F_{2^L-1}\boldsymbol{\sigma} = R(t^{2^L}, h_{2^L-1})\boldsymbol{\sigma}, \end{aligned} \tag{4}$$

where the matrix F_i, $0 \leqslant i \leqslant 2^L - 1$, that acts on the signal σ, yielding its ith subband 'contribution' σ_i, is given by the following product of Hurwitz matrices:

$$F_i = R(t^2, \alpha_{\varepsilon_1^i}) \times R(t^2, \alpha_{\varepsilon_2^i}) \times \cdots \times R(t^2, \alpha_{\varepsilon_L^i}),$$

$\varepsilon_j^i \in \{0, 1\}$ being the coefficients of the dyadic expansion of index i, namely

$$i = \sum_{j=1}^{L} \varepsilon_j^i 2^{j-1}.$$

Using Proposition 2.2 we thus have that F_i is a Hurwitz matrix of step 2^L and generating function

$$h_i = \prod_{j=1}^{L} \alpha_{\varepsilon_j^i} \circ t^{2^{L-j}}. \tag{5}$$

Fig. 1 gives a schematic representation of the 2^L-channel analysis filter bank so obtained.

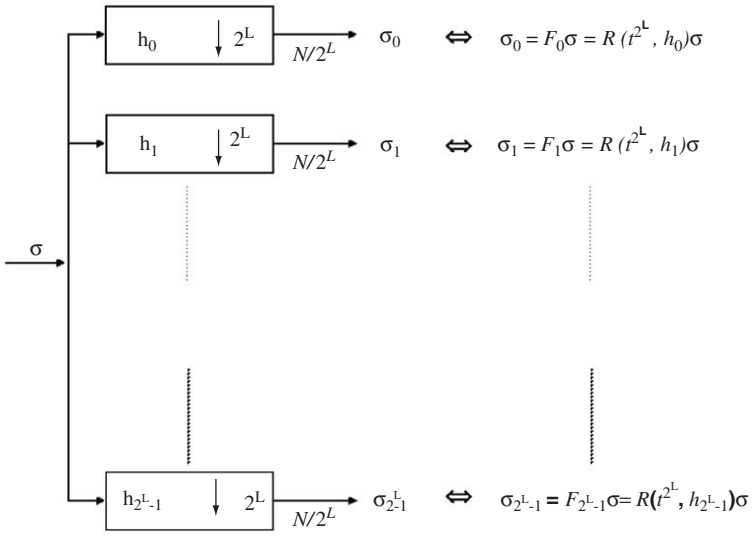

Fig. 1. Scheme of the filter bank representing L wavelet-packet decomposition steps.

4. An algebraic approach to principal component analysis

Given a random population X of vectors:

$$X = [\mathbf{x}_0, \mathbf{x}_1, \ldots, \mathbf{x}_{K-1}] = \begin{bmatrix} x_{0,0} & x_{0,1} & \cdots & x_{0,K-1} \\ x_{1,0} & x_{1,1} & \cdots & x_{1,K-1} \\ \vdots & \vdots & \vdots & \vdots \\ x_{M-1,0} & x_{M-1,1} & \cdots & x_{M-1,K-1} \end{bmatrix}.$$

The *mean vector* of the above random population is defined as

$$\mathbf{m}_x = \frac{1}{K} \begin{bmatrix} \sum_{j=0}^{K-1} x_{0j} \\ \sum_{j=0}^{K-1} x_{1j} \\ \vdots \\ \sum_{j=0}^{K-1} x_{M-1\,j} \end{bmatrix}$$

and its *covariance matrix* is obtained as

$$C = \frac{1}{K-1} \bar{X} \bar{X}^{\mathrm{T}}, \tag{6}$$

where

$$\bar{X} = [\mathbf{x}_0 - \mathbf{m}_x, \mathbf{x}_1 - \mathbf{m}_x, \ldots, \mathbf{x}_{K-1} - \mathbf{m}_x].$$

Let $\mathbf{v}_i$ and λ_i, $i = 0, \ldots, M-1$ be the eigenvectors and the corresponding eigenvalues of C, arranged in descending order so that $\lambda_i \geqslant \lambda_{i+1}$ for $i = 0, \ldots, M-2$. The matrix V, whose columns are the vectors

$\mathbf{v}_i$, is the well-known matrix used in the Karhunen–Loéve transform that satisfies the relation $V^{\mathrm{T}}CV = D = \mathrm{diag}(\lambda_0, \lambda_1, \ldots, \lambda_{M-1})$ and that maps the population of vectors $\mathbf{x}_k$ into a transformed population $\tilde{\mathbf{x}}_k$ as follows:

$$\tilde{X} = \left[\tilde{\mathbf{x}}_0, \tilde{\mathbf{x}}_1, \ldots, \tilde{\mathbf{x}}_{K-1}\right] = V^{\mathrm{T}}\bar{X}. \tag{7}$$

These new vectors turn out to be completely decorrelated, since their covariance matrix is diagonal, and their energy is compacted according to the principal component rule [12]. In fact, by denoting with $\tilde{\mathbf{X}}_i$, $i = 0, \ldots, M-1$, the row vector corresponding to the ith row of the transformed matrix $\tilde{X}$, we have that the mean energy of this row equals the ith eigenvalue of C, namely:

$$\frac{1}{K}\tilde{X}_i\tilde{X}_i^{\mathrm{T}} = \frac{1}{K}\sum_{j=0}^{K-1}\tilde{X}_{ij}^2 = \lambda_i.$$

This implies that the KL-transform compacts most of the energy of the entire population in the first rows of $\tilde{X}$.

If we now come back to the recursive matrix context, it is easy to see how, in this new algebraic framework, the KL transform can be interpreted as a filtering operation that uses the eigenvectors $\mathbf{v}_i$ as filters, yielding the vectors $\tilde{\mathbf{X}}_i$, $i = 0, \ldots, M-1$. More precisely, indicating with $\bar{\mathbf{X}}_i$, $i = 0, \ldots, M-1$, the row vector corresponding to the ith row of the mean-adjusted population matrix $\bar{X}$, the latter can be seen as a single vector $\hat{\mathbf{x}}$ of length $M \times K$, defined by the following Laurent polynomial:

$$\hat{x} = \sum_{i=0}^{M-1} t^i(\bar{X}_i \circ t^M),$$

i.e., by Propositions 2.3 and 2.4,

$$\hat{\mathbf{x}} = \sum_{i=0}^{M-1} R(t, t^{-i})R(t^M, 1)^{\mathrm{T}}\bar{\mathbf{X}}_i^{\mathrm{T}}. \tag{8}$$

It is now easy to verify that the KL-transformed matrix $\tilde{X}$ can be algebraically described in terms of recursive matrices as:

$$\begin{aligned}
\tilde{\mathbf{X}}_0^{\mathrm{T}} &= R(t^M, v_0)\hat{\mathbf{x}}\\
\tilde{\mathbf{X}}_1^{\mathrm{T}} &= R(t^M, v_1)\hat{\mathbf{x}}\\
\vdots &\qquad \vdots\\
\tilde{\mathbf{X}}_{M-1}^{\mathrm{T}} &= R(t^M, v_{M-1})\hat{\mathbf{x}}.
\end{aligned} \tag{9}$$

Fig. 2 gives a schematic representation of the filter bank associated with this interpretation of the KL transform.

5. An orthogonal wavelet packet-KL filter bank

In this section we show how the wavelet-packet transform and the principal component analysis can be suitably combined to obtain a new orthogonal filter bank with more effective filters. This result is stated in the following theorem, where, for the sake of simplicity, we consider the case of mean zero:

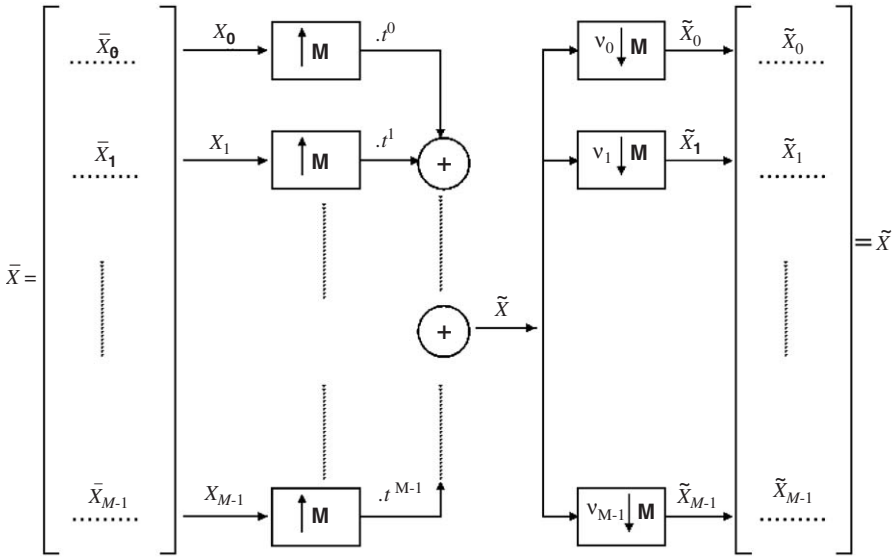

Fig. 2. Scheme of the filter bank representing the KL transform in an algebraic setting.

Theorem 5.1. *Given an orthonormal wavelet-packet basis and a signal* $\sigma \in U_n^0$, *L wavelet packet decomposition steps followed by the KL transform on the matrix S whose rows are the frequency components* $\boldsymbol{\sigma}_j$, *can be seen as an orthogonal filter bank described by*

$$\begin{aligned}
&\tilde{\boldsymbol{\sigma}}_0 = R(t^{2^L}, \mu_0)\boldsymbol{\sigma}\\
&\quad\vdots \qquad\qquad \vdots\\
&\tilde{\boldsymbol{\sigma}}_{2^L-1} = R(t^{2^L}, \mu_{2^L-1})\boldsymbol{\sigma},
\end{aligned}$$

yielding the rows of the transformed matrix $\tilde{S}$. *In particular, the generating functions of the Hurwitz matrices* $R(t^{2^L}, \mu_i)$ *are given by*

$$\boldsymbol{\mu}_i = \sum_{j=0}^{2^L-1} v_{ij}\mathbf{h}_j,$$

where the vectors $\mathbf{v}_i = [v_{i0}, \ldots, v_{i2^L-1}]$, $i = 0, \ldots, 2^L - 1$ *are the eigenvectors of the covariance matrix* (6) *of S, and the filters* h_j, $j = 0, \ldots, 2^L - 1$ *are defined in* (5).

Proof. The proof is based on the algebraic description to concatenate of relations (9), (8) and (4). In fact, L wavelet packet decomposition steps on the input σ, as described by (4), produce the 2^L vectors

$$\boldsymbol{\sigma}_i = R(t^{2^L}, h_i)\boldsymbol{\sigma}, \quad i = 0, \ldots, 2^L - 1.$$

These vectors must be suitably combined, according to (8), to yield the input for the KL transform, namely,

$$\hat{\boldsymbol{\sigma}} = \sum_{i=0}^{2^L-1} R(t, t^{-i}) \times R(t^{2^L}, 1)^{\mathrm{T}} \boldsymbol{\sigma}_i = \left[\sum_{i=0}^{2^L-1} R(t, t^{-i}) \times R(t^{2^L}, 1)^{\mathrm{T}} \times R(t^{2^L}, h_i) \right] \boldsymbol{\sigma}.$$

From Theorem 2.2, the product $R(t^{2^L}, 1)^{\mathrm{T}} \times R(t^{2^L}, h_i)$ is a block Toeplitz matrix whose generating function is $\Lambda_i = \sum_j C_j^i t^j$, with

$$C_j^i = \begin{bmatrix} \langle j|h_{i,0}^{2^L}\rangle & \cdots & \langle j|h_{i,2^L-1}^{2^L}\rangle \\ 0 & \cdots & 0 \\ \vdots & & \vdots \\ 0 & \cdots & 0 \end{bmatrix},$$

where $\langle j|h_{i,l}^{2^L}\rangle$ denotes the jth coefficient of the lth 2^L-decimated polynomial of the Laurent polynomial associated with the filter h_i. Since the above product is pre-multiplied by a shift matrix, each entry of the block coefficient C_j^i turns out to be shifted down by i positions. We denote by $\tilde{C}_j^i$ the shifted blocks and by $\tilde{\Lambda}_i = \sum_j \tilde{C}_j^i t^j$ the new block generating function. Therefore, we obtain

$$\hat{\boldsymbol{\sigma}} = \left[\sum_{i=0}^{2^L-1} R(t, \tilde{\Lambda}_i) \right] \boldsymbol{\sigma} = R(t, P)\boldsymbol{\sigma},$$

where the generating function of the block Toeplitz matrix $R(t, P)$ is the block Laurent polynomial

$$P = \sum_{i=0}^{2^L-1} \tilde{\Lambda}_i = \sum_{i=0}^{2^L-1} \sum_j \tilde{C}_j^i t^j = \sum_j \left(\sum_{i=0}^{2^L-1} \tilde{C}_j^i \right) t^j = \sum_j P_j t^j,$$

with

$$P_j = \begin{bmatrix} \langle j|h_{0,0}^{2^L}\rangle & \cdots & \langle j|h_{0,2^L-1}^{2^L}\rangle \\ \langle j|h_{1,0}^{2^L}\rangle & \cdots & \langle j|h_{1,2^L-1}^{2^L}\rangle \\ \vdots & & \vdots \\ \langle j|h_{2^L-1,0}^{2^L}\rangle & \cdots & \langle j|h_{2^L-1,2^L-1}^{2^L}\rangle \end{bmatrix}.$$

By now applying relation (9), which algebraically describes the KL transform, we have that

$$\tilde{\boldsymbol{\sigma}}_i = R(t^{2^L}, v_i)\hat{\boldsymbol{\sigma}} = R(t^{2^L}, v_i) \times R(t, P)\boldsymbol{\sigma}.$$

We remark that $R(t^{2^L}, v_i)$ is a scalar Hurwitz matrix of step 2^L, which can also be seen as a block Toeplitz matrix with blocks 1×2^L. We can thus apply the product rule for block Toeplitz matrices (see [1]) and we get

$$\tilde{\boldsymbol{\sigma}}_i = R(t, M_i)\boldsymbol{\sigma},$$

where M_i is a 1×2^L block Laurent polynomial, namely, $M_i = \sum_j \mathbf{v}_i^{\mathrm{T}} P_j t^j = \sum_j M_{ij} t^j$. In particular,

$$M_{ij} = \left[\sum_{k=0}^{2^L-1} v_{ik} \langle j | h_{k,0}^{2^L} \rangle \sum_{k=0}^{2^L-1} v_{ik} \langle j | h_{k,1}^{2^L} \rangle \dots \sum_{k=0}^{2^L-1} v_{ik} \langle j | h_{k,2^L-1}^{2^L} \rangle \right].$$

It is easy to see that the block generating function M_i can be seen as the following 1×2^L matrix whose entries are scalar Laurent polynomials:

$$\begin{aligned} M_i &= \left[\sum_j \sum_{k=0}^{2^L-1} v_{ik} \langle j | h_{k,0}^{2^L} \rangle t^j \dots \sum_j \sum_{k=0}^{2^L-1} v_{ik} \langle j | h_{k,2^L-1}^{2^L} \rangle t^j \right] \\ &= \left[\sum_{k=0}^{2^L-1} v_{ik} h_{k,0}^{2^L} \dots \sum_{k=0}^{2^L-1} v_{ik} h_{k,2^L-1}^{2^L} \right] = [m_0, \dots, m_{2^L-1}]. \end{aligned}$$

By using Theorem IV.1, given in [4], we obtain that $R(t, M_i)$ can be seen as a scalar Hurwitz matrix $R(t^{2^L}, \mu_i)$ of step 2^L with the following generating function:

$$\mu_i = m_0 \circ t^{2^L} + t(m_1 \circ t^{2^L}) + \dots + t^{2^L-1}(m_{2^L-1} \circ t^{2^L}).$$

By using the expression of the coefficients m_i, $i = 0, \dots, 2^L - 1$, and remembering relation (1), we obtain

$$\mu_i = \sum_{k=0}^{2^L-1} v_{ik} h_k,$$

which proves our assertion. □

The previous result shows us how to simply combine two powerful tools of signal processing in order to obtain a new data-dependent operator, which optimally splits a signal into its most significant frequency components. Fig. 3 shows an example of how the wavelet packet decomposition is improved by the use of the new filters μ_i.

6. Application of the wavelet packet-KL filter bank for image denoising

The results of the previous sections show how to construct a data-dependent orthogonal analysis filter bank which well separates the signal frequency contents. In this section, we make a practical application of the above procedure to show how the use of this method allows us to realize an efficient image denoising algorithm. Since we are working with images, we consider an extension of the wavelet-packet decomposition to the $2D$ case; this is obtained by using the well-known tensor product technique. After L steps of wavelet packet decomposition, the starting $N \times N$ image turns out to be decomposed into $M = 2^L \times 2^L$ subband images, $\boldsymbol{\sigma}_i$, each having dimension $K = N/2^L \times N/2^L$. These M subimages

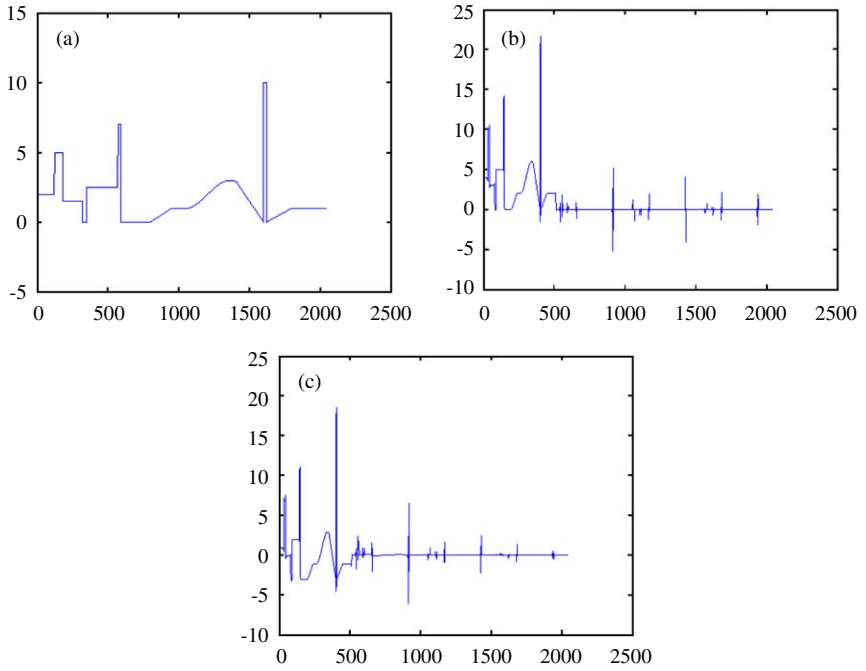

Fig. 3. (a) Original signal; (b) two wavelet-packet decomposition steps; (c) two wavelet-packet decomposition steps followed by KL.

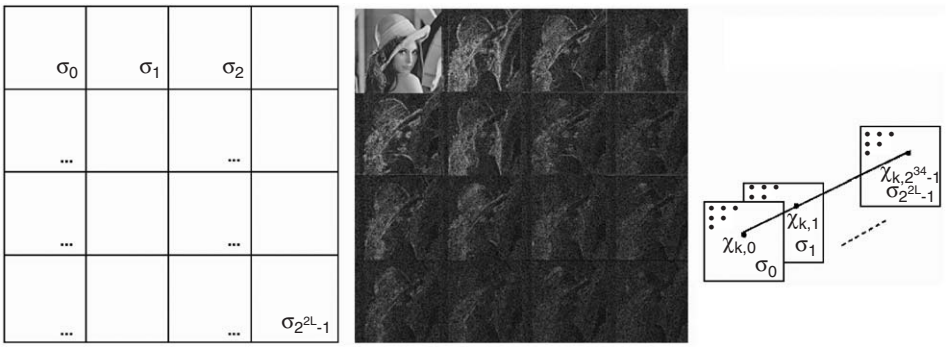

Fig. 4. Subbands of the two-dimensional orthogonal wavelet-packet transform for two decomposition levels and input vectors of the KL transform.

constitute the rows of the following population matrix

$$S = \begin{bmatrix} \sigma_{0,0} & \sigma_{0,1} & \cdots & \sigma_{0,K-1} \\ \vdots & \vdots & \vdots & \vdots \\ \sigma_{M-1,0} & \sigma_{M-1,1} & \cdots & \sigma_{M-1,K-1} \end{bmatrix},$$

which is now the input of the KL transform, as visually shown in Fig. 4.

The transformed population

$$\tilde{S} = V^{\mathrm{T}} S$$

represents the image in a more suitable form for dealing with the problem of image denoising. In fact, if the original image is corrupted by Gaussian noise with mean 0 and variance σ^2, this further KL decomposition step yields a better separation between signal and noise energy, allowing us to apply a new thresholding rule, that takes advantage of this separation, instead of the classical soft shrinkage function, to the wavelet packet-KL coefficients. More precisely, we propose an algorithm consisting of two phases:

1. *Hard Thresholding phase*: we discard all the frequency subbands whose mean energy is less than the variance σ^2 of the Gaussian noise,
2. *Filtering phase*: we apply to the retained coefficients the exponential shrinkage function

$$F_\tau(x) = \begin{cases} a^{-h\frac{\tau^2}{x^2}} & \text{if } x \neq 0 \text{ with } a \geqslant 1 \text{ and } h \geqslant 0 \\ 0 & \text{otherwise} \end{cases}$$

proposed in [2]. In this context, the threshold τ can be chosen to be "band-dependent"; namely, for each subband σ_j, τ depends on the relation existing between the mean energy of the subband (i.e., λ_j) and the variance of the noise, as follows:

$$\tau_j = \sqrt{\frac{\lambda_j \sigma^2}{\lambda_j - \sigma^2}}.$$

In summary, the thresholded coefficients are

$$\hat{\tilde{\sigma}}_{k,j} = \begin{cases} 0 & \text{if } \dfrac{1}{K}\tilde{\boldsymbol{\sigma}}_j \tilde{\boldsymbol{\sigma}}_j^{\mathrm{T}} \leqslant \sigma^2, \\ F_{\tau_j}(\tilde{\sigma}_{k,j})\tilde{\sigma}_{k,j} & \text{otherwise.} \end{cases}$$

In order to test the performance of the proposed denoising algorithm, we have considered four classical gray-scale images with dimension 512×512, shown in Fig. 5. These test images have been corrupted with i.i.d. Gaussian noise and with different values of the standard deviation ($\sigma = 5, 10, 15, 25, 35$). For the denoising algorithm we have considered $L = 3$ wavelet-packet decomposition steps, obtained with interval adapted Daubechies orthonormal wavelet bases with 4 vanishing moments [9], followed by the KL. In Table 1 we give an objective estimation of the quality of the results, evaluated by means of the root mean square error (RMSE) and the peak signal-to-noise ratio ($\mathrm{PSNR} = 20\log_{10}(255/\mathrm{RMSE})$). In Table 2, for a comparison purpose, we show both the results obtained in [16] and those obtained by means of the classical Soft thresholding method in the wavelet packet domain with the Generalized Cross Validation choice of the threshold [13,15] on the Barbara image, corrupted by Gaussian noise with $\sigma = 25$. It is easily seen that our method outperforms the two above techniques, highlighting its particular effectiveness on 'difficult' images that present sharp edges and many little details. For comparison, in Fig. 6, we also give the results obtained by means of the classical Soft thresholding with GCV threshold. It is evident that, for highly perturbed images, our method presents a sensible improvement on the denoised image quality, especially in the edge preserving capability.

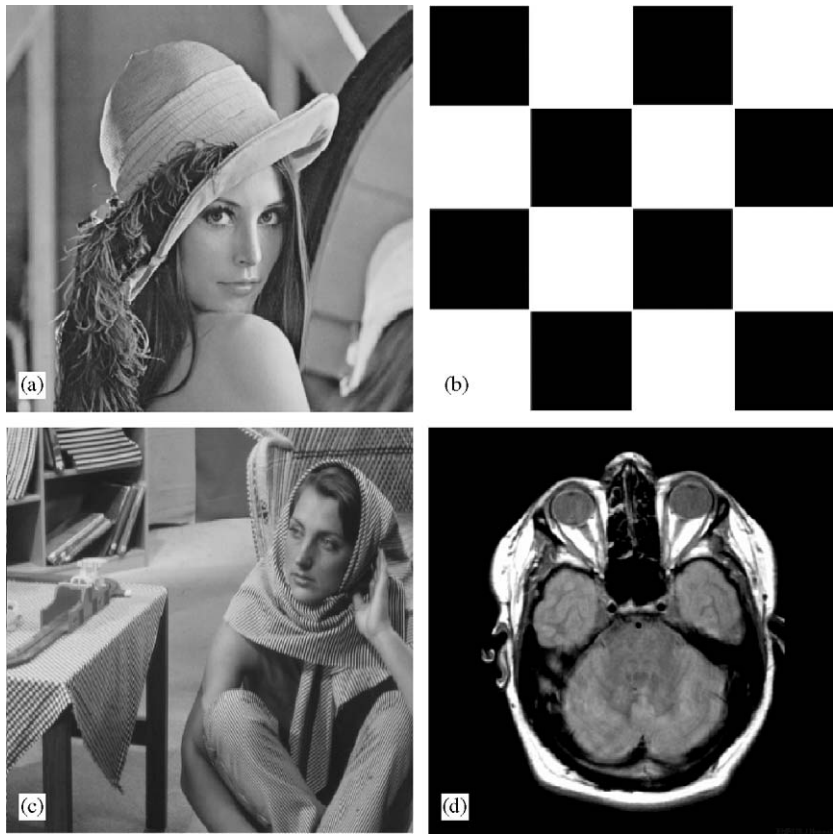

Fig. 5. Test images: (a) 512 × 512 Lena; (b) 512 × 512 Chessboard; (c) 512 × 512 Barbara; (d) 512 × 512 MRI.

Table 1
Numerical results on the test images with increasing values of the standard deviation of the noise

Images	Lena		Chessboard		Barbara		MRI images	
	RMSE	PSNR	RMSE	PSNR	RMSE	PSNR	RMSE	PSNR
$\sigma = 5$	3.67	36.83	2.18	41.32	3.93	36.27	2.69	39.52
$\sigma = 10$	5.41	33.46	3.59	37.01	6.56	31.79	4.29	35.46
$\sigma = 15$	6.70	31.61	4.68	34.71	8.77	29.26	5.56	33.22
$\sigma = 25$	8.78	29.26	6.68	31.62	12.22	26.38	7.71	30.85
$\sigma = 35$	10.44	27.75	8.42	29.62	15.18	24.50	9.50	28.57

A possible choice for the parameters h and a is $h = 4$, $a = 2$; generally, the values of these parameters are chosen depending on the amount of noise.

Table 2
Comparison between our proposed denoising method, the Soft thresholding with GCV threshold and the method proposed in [16]

Barbara	Proposed method		GCV method		Strela method	
	RMSE	PSNR	RMSE	PSNR	RMSE	PSNR
$\sigma = 25$	12.22	26.38	12.45	26.22	14.02	25.19

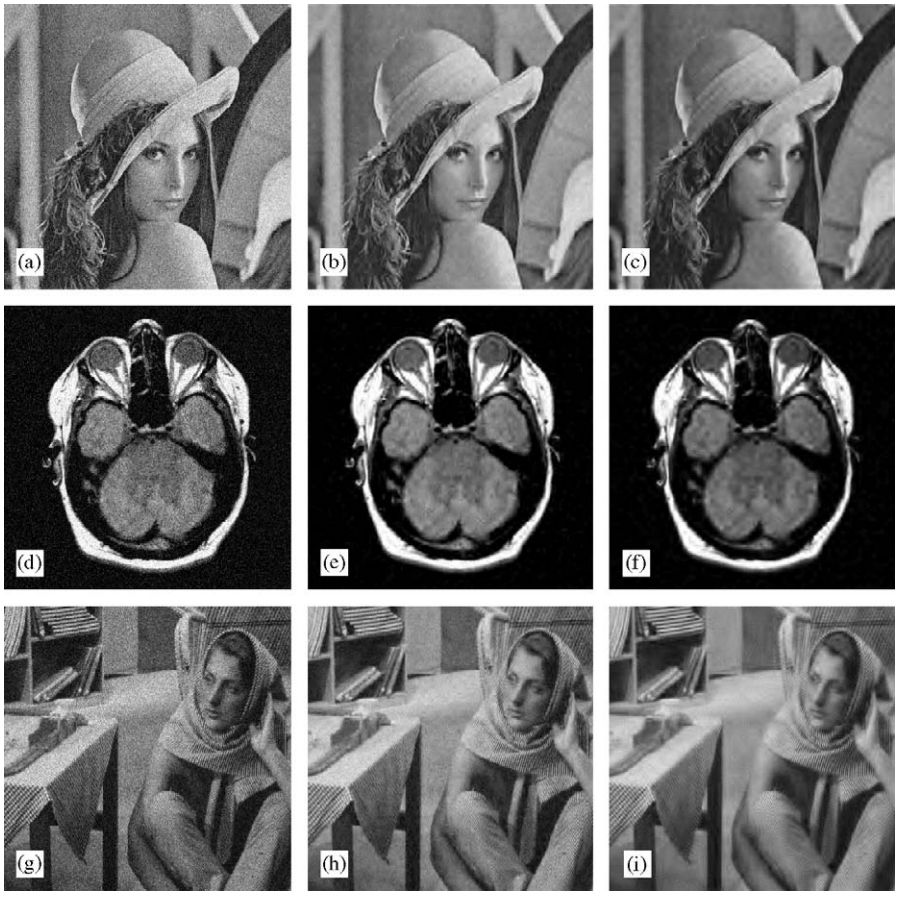

Fig. 6. (a) Lena noisy image with $\sigma = 25$; (b) denoised image with the proposed algorithm $PSNR = 29.23$; (c) denoised image with the wavelet packet Soft thresholding with GCV threshold $PSNR = 28.44$; (d) MRI noisy image with $\sigma = 35$; (e) denoised image with the proposed algorithm $PSNR = 28.57$; (f) denoised image with wavelet packet Soft thresholding with GCV threshold; $PSNR = 27.32$; (g) Barbara noisy image with $\sigma = 25$; (h) denoised image with the proposed algorithm $PSNR = 26.38$; (i) denoised image with wavelet packet Soft thresholding with GCV threshold; $PSNR = 26.22$.

Acknowledgements

This research was supported by MIUR, Cofin 2002 and R.F.O. projects.

References

[1] S. Bacchelli, Matrci ricorsive a blocchi: teoria ed applicazione nella elaborazione multiwavelet di immagini, Ph.D. Thesis, Padova, 2001.
[2] S. Bacchelli, S. Papi, Filtered wavelet thresholding methods, J. Comput. Appl. Math. 165–165C (2004) 39–52.
[3] M. Barnabei, L. Montefusco, Recursive properties of Toeplitz and Hurwitz matrices, Linear Algebra Appl. 274 (1998) 367–388.
[4] M. Barnabei, L. Montefusco, An algebraic approach to subband signal processing, Adv. Imaging Electron Phys. 125 (2002) 1–62.
[5] M. Barnabei, L. Montefusco, C. Guerrini, Some algebraic aspects of signal processing, Linear Algebra Appl. 284 (1998) 3–17.
[6] A. Chambolle, R.A. Devore, N. Lee, B.J. Lucier, Nonlinear wavelet image processing: variational problems, compression and noise removal through wavelet shrinkage, IEEE Trans. Image Process. 7 (1998) 319–335.
[7] S.G. Chang, B. Yu, M. Vetterli, Adaptive wavelet thresholding for image denoising and compression, IEEE Trans. Image Process. 9 (9) (2000) 1532–1546.
[8] C.K. Chui, An Introduction to Wavelets, Academic Press, Boston, 1992.
[9] A. Cohen, I. Daubechies, P. Vial, Wavelets on the interval and fast wavelet transforms, Appl. Comput. Harmonic Analysis 1 (1993) 54–81.
[10] I. Daubechies, Ten lectures on wavelets, CBMS-NSF Regional Conference on Series in Applied Mathematics, vol. 61, Society for Industrial and Applied Mathematics, Philadelphia, PA, 1992.
[11] D. Donoho, I. Johnstone, Ideal spatial adaptation by wavelet shrinkage, Biometrika 81 (1994) 425–455.
[12] R.C. Gonzalez, R.E. Woods, Digital Image Processing, Addison-Wesley, Reading, MA
[13] M. Jansen, M. Malfait, A. Bultheel, Generalized cross validation for wavelet thresholding, Signal Process. 56 (1997) 33–44.
[14] S. Mallat, Wavelet Tour of Signal Processing, Academic Press, New York, 1998.
[15] G.P. Nason, Wavelet shrinkage using cross-validation, J. Statist. Soc. Ser. B 58 (1996) 463–479.
[16] V. Strela, Denoising via block Wiener filtering in wavelet domain, Proceedings of the Third European Congress of Mathematics, Progress in Mathematics Series, Birkhauser, Barcelona, 2000.