SHRINKAGE FOR REDUNDANT REPRESENTATIONS

Michael Elad

Department of Computer Science
The Technion - Israel Institute of Technology, Haifa 32000 Israel
email: elad@cs.technion.ac.il.

ABSTRACT

Shrinkage is a well known and appealing denoising technique. The use of shrinkage is known to be optimal for Gaussian white noise, provided that the sparsity on the signal's representation is enforced using a unitary transform. Still, shrinkage is also practiced successfully with non-unitary, and even redundant representations. In this paper we shed some light on this behavior. We show that simple shrinkage could be interpreted as the first iteration of an algorithm that solves the basis pursuit denoising (BPDN) problem. Thus, this work leads to a sequential shrinkage algorithm that can be considered as a novel and effective pursuit method.

1. INTRODUCTION

One way to pose the maximum a-posteriori probability (MAP) estimator for the denoising problem is the minimization of the function

$$f(\mathbf{x}) = \frac{1}{2} \cdot \|\mathbf{x}-\mathbf{y}\|_2^2 + \lambda \cdot \mathbf{1}^T \cdot \rho\left\{\mathbf{Tx}\right\} \ . \qquad (1)$$

The first term is known as the log-likelihood, describing the relation between the desired (clean) signal, $\mathbf{x} \in \mathbb{R}^N$, and a noisy version of it, $\mathbf{y} \in \mathbb{R}^N$. We assume the model $\mathbf{y} = \mathbf{x} + \mathbf{v}$, with $\mathbf{v} \in \mathbb{R}^N$ a Gaussian zero mean white noise. The term $\mathbf{1}^T \cdot \rho\left\{\mathbf{Tx}\right\}$ stands for the prior posed on the unknown signal $\mathbf{x}$, based on sparsity of the unknown signal with respect to its transformed ($\mathbf{T}$) representation. The function ρ is a scalar robust measure (e.g., $\rho(z) = |z|$), and when operating on a vector, it does so entry-wise. The multiplication by $\mathbf{1}^T$ sums those robust measures.

Donoho and Johnstone pioneered a wavelet based signal denoising algorithm in line with the above structure. They advocated the use of sparsity of the wavelet coefficients $\mathbf{Wx}$ (i.e., here $\mathbf{T}$ is the unitary matrix $\mathbf{W}$) as a driving force in recovering the desired signal [1, 2]. Later work in [3, 4, 5] simplified these ideas and related them to the MAP formulation as presented above. Interestingly, using such a prior in Equation (1) leads to a *simple closed-form solution, known as shrinkage*. This solution amounts to a wavelet transform on the noisy signal, a look-up-table (LUT) function on the coefficients (that depends on the function ρ), $\mathcal{S}\{\mathbf{Wy}\}$, and an inverse wavelet transform to produce the outcome $\hat{\mathbf{x}}$. More on how the shrinkage algorithm becomes indeed the optimal solver of (1) can be found in [6]. This optimality depends strongly on the ℓ^2-norm used in evaluating the distance $\mathbf{x}-\mathbf{y}$, and this has direct roots in the white Gaussianity assumptions on the noise. Also, crucial to the optimality of this method is the orthogonality of $\mathbf{W}$.

A new trend of recent years is the use of overcomplete transforms, replacing the traditional unitary ones – see [7, 8, 9, 10, 11, 12] for representative works. This trend was partly motivated by the growing realization that orthogonal wavelets are weak in describing the singularities found in images. Another driving force in the introduction of redundant representations is the sparsity it can provide, which many applications find desirable. Finally, we should mention the desire to obtain shift-invariant transforms, again calling for redundancy in the representation. In these methods the transform is defined via a non-square full rank matrix $\mathbf{T} \in \mathbb{R}^{L\times N}$, with $L > N$. Such redundant methods, like the undecimated wavelet transform, curvelet, contourlet, and steerable-wavelet, were shown to be more effective in representing images, and other signal types.

Given a noisy signal $\mathbf{y}$, one can still follow the shrinkage procedure, by computing the forward transform $\mathbf{Ty}$, putting the coefficients through a shrinkage LUT operation $\mathcal{S}\{\mathbf{Ty}\}$, and finally applying the inverse transform to obtain the denoised outcome, $\mathbf{T}^+\mathcal{S}\{\mathbf{Ty}\}$. Will this be the solver of (1)? The answer is no! As we have said before, the orthogonality of the transform plays a crucial role in the construction of the shrinkage as an optimal procedure. Still, shrinkage is practiced quite often with non-unitary, and even redundant representations, typically leading to satisfactory results – see [7, 8, 9] for representative examples. Naturally, we should wonder why this is so.

In this paper we shed some light on this behavior. Our main argument is that such a shrinkage could be interpreted as the first iteration of a converging algorithm that solves the basis pursuit denoising (BPDN) problem. The BPDN forms a similar problem to the one posed in (1), replacing the analysis prior with a generative one. While the desired solution of BPDN is hard to obtain in general, a simple iterative procedure that amounts to step-wise shrinkage can be employed with quite successful performance. Thus, beyond showing that shrinkage has justified roots in solid denoising methodology, we also show how shrinkage can be iterated in a simple form, to further strengthen the denoising effect. As a byproduct, we get an effective pursuit algorithm that minimizes the BPDN functional via simple steps.

In the next section we bridge between an analysis based objective function and a synthesis one, leading to the BPDN. Section 3 then develops the iterated shrinkage algorithm that minimizes it. In Section 4 we present few simulations to illustrate the algorithm proposed.

2. FROM ANALYSIS TO SYNTHESIS-BASED PRIOR

Starting with the penalty function posed in (1), we define $\mathbf{x}_T = \mathbf{Tx}$. Multiplying both sides by $\mathbf{T}^T$, and using the fact that $\mathbf{T}$ is full-rank, we get[1] $\mathbf{x} = (\mathbf{T}^T\mathbf{T})^{-1}\mathbf{T}^T\mathbf{x}_T = \mathbf{T}^{+}\mathbf{x}_T$. Using these relations to rearrange Equation (1), we obtain a new function of the representation vector $\mathbf{x}_T$,

$$\tilde{f}(\mathbf{x}_T) = \frac{1}{2} \cdot \|\mathbf{D}\mathbf{x}_T - \mathbf{y}\|_2^2 + \lambda \cdot \mathbf{1}^T \cdot \rho\left\{\mathbf{x}_T\right\}, \quad (2)$$

where we have used the notation $\mathbf{D} = \mathbf{T}^{+}$.

Denoising can be done by minimizing f and obtaining a solution $\hat{\mathbf{x}}_1$. Alternatively, we can minimize $\tilde{f}$ with respect to $\mathbf{x}_T$ and deduce the denoised outcome by $\hat{\mathbf{x}}_2 = \mathbf{D}\hat{\mathbf{x}}_T$. Interestingly, *these two results are not expected to be the same in the general case*, since in the conversion from f to $\tilde{f}$ we have expanded the set of feasible solutions by allowing $\mathbf{x}_T$ to be an arbitrary vector in $\mathbb{R}^L$, whereas the original definition $\mathbf{x}_T = \mathbf{T}\mathbf{x}$ implies that it must be confined to the column space of $\mathbf{T}$. Notice that this difference between the two formulations disappears when $\mathbf{T}$ is full rank square matrix, which explains why this dichotomy of methods do not bother us for the regular wavelet transform.

Still, the formulation posed in (2) is a feasible alternative Bayesian method that uses a generative prior. Indeed, for the choice $\rho\{z\} = |z|$, this formulation is known as the basis pursuit denoising (BPDN). Referring to $\mathbf{D}$ as a dictionary of signal prototypes (atoms) being its columns, we assume that the desired signal $\mathbf{x}$ is a linear construction of these atoms, with coefficients drawn independently from a probability density function proportional to $\exp\{-\text{Const} \cdot \rho\{x_T(j)\}\}$. In the case of $\rho(z) = |z|$ this is the Laplace distribution, and we effectively promote sparsity in the representation.

3. PROPOSED ALGORITHM

3.1. Sequential Approach

We desire the minimization of (2). Assume that in an iterative process used to solve the above problem, we hold the k-th solution $\hat{\mathbf{z}}_k$. We are interested in updating its j-th entry, $z(j)$, assuming all the others as fixed. Thus, we obtain a one-dimensional optimization problem of the form

$$\min_w \frac{1}{2} \cdot \| \left[\mathbf{D}\mathbf{z}_k - \mathbf{d}_j z_k(j)\right] + \mathbf{d}_j w - \mathbf{y}\|_2^2 + \lambda \cdot \rho\left\{w\right\}. \quad (3)$$

In the above expression, $\mathbf{d}_j$ is the j-th column in $\mathbf{D}$. The term $\mathbf{D}\mathbf{z}_k - \mathbf{d}_j z_k(j)$ uses the current solution for all the coefficients, but discards of the j-th one, assumed to be replaced with a new value, w.

Since this is a 1D optimization task, it is relatively easy to solve. If $\rho(w) = |w|$, the derivative is

$$0 = \mathbf{d}_j^T \left(\left[\mathbf{D}\mathbf{z}_k - \mathbf{d}_j z_k(j)\right] + \mathbf{d}_j w - \mathbf{y}\right) + \lambda \cdot \text{sign}\left\{w\right\}, \quad (4)$$

leading to

$$\begin{aligned} w &= z_k(j) + \frac{\mathbf{d}_j^T\left(\mathbf{y} - \mathbf{D}\mathbf{z}_k\right)}{\|\mathbf{d}_j\|_2^2} - \frac{\lambda \cdot \text{sign}\left\{w\right\}}{\|\mathbf{d}_j\|_2^2} \quad (5)\\ &= v(\mathbf{D}, \mathbf{y}, \mathbf{z}_k, j) - \hat{\lambda}(j) \cdot \text{sign}\left\{w\right\}. \end{aligned}$$

Here we have defined

$$\begin{aligned} v(\mathbf{D}, \mathbf{y}, \mathbf{z}_k, j) &= \frac{\mathbf{d}_j^T\left(\mathbf{y} - \mathbf{D}\mathbf{z}_k\right)}{\|\mathbf{d}_j\|_2^2} + z_k(j) \text{ and} \quad (6)\\ \hat{\lambda}(j) &= \frac{\lambda}{\|\mathbf{d}_j\|_2^2}. \end{aligned}$$

[1] If $\mathbf{T}$ is a tight frame ($\alpha\mathbf{T}^T\mathbf{T} = \mathbf{I}$), then $\mathbf{x} = \alpha\mathbf{T}^T\mathbf{x}_T$.

Both $v(\mathbf{D}, \mathbf{y}, \mathbf{z}_k, j)$ and $\hat{\lambda}(j)$ are computable using the known ingredients and thus this leads to a closed form formula for the optimal solution for w, being a shrinkage operation on $v(\mathbf{D}, \mathbf{y}, \mathbf{z}_k, j)$,

$$\begin{aligned} w_{opt} &= \mathcal{S}\left\{v(\mathbf{D}, \mathbf{y}, \mathbf{z}_k, j)\right\} \quad (7)\\ &= \begin{cases} v(\mathbf{D}, \mathbf{y}, \mathbf{z}_k, j) - \hat{\lambda}(j) & \text{for } v(\mathbf{D}, \mathbf{y}, \mathbf{z}_k, j) > \hat{\lambda}(j) \\ 0 & \text{for } |v(\mathbf{D}, \mathbf{y}, \mathbf{z}_k, j)| \leq \hat{\lambda}(j) \\ v(\mathbf{D}, \mathbf{y}, \mathbf{z}_k, j) + \hat{\lambda}(j) & \text{for } v(\mathbf{D}, \mathbf{y}, \mathbf{z}_k, j) < -\hat{\lambda}(j) \end{cases}. \end{aligned}$$

A similar LUT result can be developed for any many other choices of the function $\rho(\cdot)$.

It is tempting to suggest an algorithm that applies the above procedure for $j = 1, 2, \ \ldots, \ , L$, updating one coefficient at a time in a sequential coordinate descent algorithm, and cycle such process several times. While such algorithm necessarily converges, and could be effective in minimizing the objective function using scalar shrinkage operations only, it is impractical in most cases. The reason is the necessity to draw one column at a time from $\mathbf{D}$ to perform this computation. Consider, for example, the curvelet dictionary. While the transform and its inverse can be interpreted as multiplications by the dictionary and its transpose (because it is a tight frame), this matrix is never explicitly constructed, and an attempt to draw basis functions from it or store them could be devastating. Thus we take a different route.

3.2. Parallel Approach

Given the current solution $\mathbf{z}_k$, let us assume that we use the above update formulation to update *all the coefficients* in parallel, rather than doing this sequentially. Obviously, this process must be slower in minimizing the objective function, but with this slowness comes a blessed simplicity that will be evident shortly.

First, let us convert the terms $v(\mathbf{D}, \mathbf{y}, \mathbf{z}_k, j)$ in Equation (6) to a vector form that accounts for all the updates at once. Gathering these terms for all $j \in [1, L]$, this reads

$$\mathbf{v}(\mathbf{D}, \mathbf{y}, \mathbf{z}_k) = \text{diag}^{-1}\left\{\mathbf{D}^T\mathbf{D}\right\}\mathbf{D}^T\left(\mathbf{y} - \mathbf{D}\mathbf{z}_k\right) + \mathbf{z}_k. \quad (8)$$

If the transform we use is such that multiplication by $\mathbf{D}$ and its adjoint are fast, then computing the above term is easy and efficient. Notice that here we do not need to extract some columns from the dictionary, and need not use these matrices explicitly in any other way. The normalization by the norms of the columns is simple to obtain and can be kept as fixed parameters of the transform, computed once off-line. In the case of tight frames, applying multiplications by $\mathbf{D}^T$ and $\mathbf{D}$ are the forward and the inverse transforms, up to a constant. For a non-tight frame, the above formula says that we need to be able to apply the adjoint *and not the pseudo-inverse* of $\mathbf{D}$.

There is also a natural weakness to the above strategy. One cannot take a shrinkage of the above vector with respect to the threshold vector $\lambda \cdot \text{diag}^{-1}\left\{\mathbf{D}^T\mathbf{D}\right\} \cdot \mathbf{1}$, and expect the objective function to be minimized well. While updating every scalar entry w_j using the above shrinkage formula is necessarily decreasing the function's value, applying all those at once is likely to diverge, and cause an ascent in the objective. Thus, instead of applying a complete shrinkage as Equation (7) suggests, we consider a relaxed step of the form

$$\mathbf{z}_{k+1} = \mathbf{z}_k + \mu\left[\mathcal{S}\left\{\mathbf{v}(\mathbf{D}, \mathbf{y}, \mathbf{z}_k)\right\} - \mathbf{z}_k\right] = \mathbf{z}_k + \mu\mathbf{h}_k. \quad (9)$$

This way, we compute the shrinkage vector as the formula suggests, and use it to define a descent direction. The solution is starting from the current solution $\mathbf{z}_k$ and updates it by "walking" towards the shrinkage result. For a sufficiently small $\mu > 0$, this step *must* lead to a feasible descent in the penalty function, because this direction is a non-negative combination of L descent directions.

We can apply a line search to find the proper choice for the value of μ. In general, a line search seeks the best step-size as a 1D optimization procedure that solves

$$\min_{\mu} \frac{1}{2} \cdot \|\mathbf{D}\left[\mathbf{z}_k + \mu \mathbf{h}_k\right] - \mathbf{y}\|_2^2 + \lambda \cdot \mathbf{1}^T \cdot \rho\left\{\mathbf{z}_k + \mu \mathbf{h}_k\right\}, \quad (10)$$

where $\mathbf{h}_k$ is a computable vector. As it turns out, the solution in this case is given also as a shrinkage-like procedure [6].

Looking at the first iteration, and assuming that the algorithm is initialized with $\mathbf{z}_0 = \mathbf{0}$, the term in Equation (8) becomes

$$\mathbf{v}(\mathbf{D}, \mathbf{y}, \mathbf{0}) = \text{diag}^{-1}\left\{\mathbf{D}^T\mathbf{D}\right\}\mathbf{D}^T\mathbf{y}. \quad (11)$$

The solution $\mathbf{z}_1$ is obtained by first applying shrinkage to the above vector, using $\lambda\text{diag}^{-1}\left\{\mathbf{D}^T\mathbf{D}\right\}\mathbf{1}$ as the threshold vector, and then relaxing it, as in Equation (9). The denoised outcome is thus

$$\mathbf{D}\mathbf{z}_1 = \mu\mathbf{D}\mathcal{S}\left\{\text{diag}^{-1}\left\{\mathbf{D}^T\mathbf{D}\right\}\mathbf{D}^T\mathbf{y}\right\}, \quad (12)$$

and the resemblance to the heuristic shrinkage is evident. In fact, for tight frames with normalized columns the above becomes exactly equal to the heuristic shrinkage [6].

4. EXPERIMENTAL RESULTS

We present here a simple set of experiments that corresponds to the case of a tight frame with normalized columns. Other experiments are reported in [6]. We build $\mathbf{D}$ as a union of 10 random unitary matrices of size 100×100 each. We synthesize a sparse representation $\mathbf{z}_0$ with 15 non-zeros in random locations and Gaussian i.i.d. entries, so as to match the sparsity prior we use. Thus the clean signal is defined as $\mathbf{x}_0 = \mathbf{D}\mathbf{z}_0$. This signal is contaminated by a Gaussian i.i.d noise $\sigma = 0.3$ (parallels an SNR of ≈ 1.3dB).

We apply several algorithms to the denoising task: (i) a heuristic shrinkage as described in the introduction; (ii) the IRLS algorithm, which is very heavy but minimizes the BPDN effectively, and thus is good as a reference method [6]; (iii) the sequential shrinkage algorithm developed above; and (iv) the parallel counterpart. We assume $\rho(z) = |z|$, and the results are reported in Figures 1- 3.

First, we show how effective are these algorithms in minimizing the objective in Equation (2). Figure 1 presents the value of the objective as a function of the iteration number. Here we have implemented the paralle shrinkage algorithm both with a fixed $\mu = 1/\alpha$ and with a line-search. As expected, the IRLS performs the best in terms of convergence speed. The sequential and the parallel (with line-search) coordinate descent are comparable to each other, being somewhat inferior to the IRLS.

When implementing these algorithms for denoising, we sweep through the possible values of λ to find the best choice. In assessing the denoising effect, we use the noise decay factor measure $r(\hat{\mathbf{x}}, \mathbf{x}_0, \mathbf{y}) = \|\hat{\mathbf{x}} - \mathbf{x}_0\|_2^2 / \|\mathbf{y} - \mathbf{x}_0\|_2^2$, which gives the ratio between the final reconstruction error and the error with $\mathbf{y}$ as our estimate. Thus, a value smaller than 1 implies a decay in the noise, and the closer it is to zero the better the result.

We compare the IRLS results (after the 1-st and the 5-th iterations) to the simple shrinkage algorithm. The simple shrinkage in

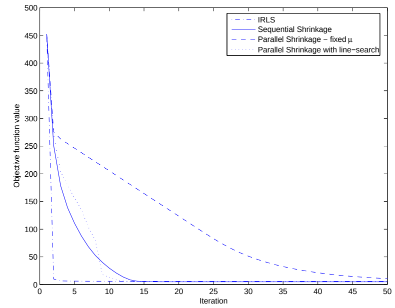

Fig. 1. The objective as a function of the iteration – algorithms B-D.

this case uses a threshold being $\lambda/\alpha = 10\lambda$, so as to match to the objective function that uses λ in its formulation. Figure 2 presents this comparison, showing the noise decay factor versus λ. Interestingly, it appears than the simple shrinkage manages to utilize most of the denoising potential, and 5 iterations of the IRLS give only slightly better results.

Figure 3 presents a similar comparison of the simple shrinkage with the parallel coordinate descent shrinkage with a fixed μ chosen as $\mu = 1/\alpha$. We see that the first iteration of the parallel shrinkage aligns perfectly with the simple shrinkage when $\mu = 1/\alpha$, as predicted, and having 5 iterations gives a slight improvement. Other experiments with the other algorithms are reported in [6], and were omitted from here because of space constraints.

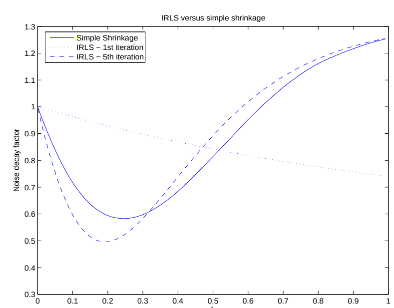

Fig. 2. The denoising effect of the IRLS versus simple shrinkage.

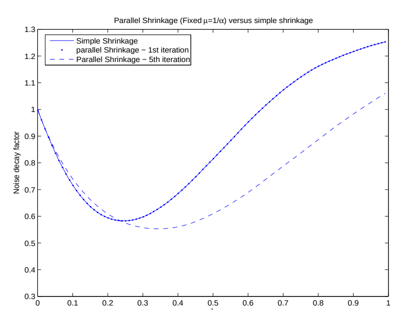

Fig. 3. The denoising effect of the parallel coordinate descent algorithm versus simple shrinkage.

5. RELATED WORK

Interestingly, a sequence of recent contributions proposed a similar sequential shrinkage algorithm. First, the work reported in [13, 14] uses such an algorithm for finding the sparsest representation over redundant dictionaries (such as the curvelet, or combination of dictionaries). These papers motivated such algorithm heuristically, relying on the resemblance to the unitary case, on one hand, and the block-coordinate-relaxation method, on the other [15].

Figueiredo and Nowak suggested a constructive method for image deblurring, based on iterated shrinkage [16]. Their algorithm aims at minimizing the penalty function

$$f_B(\mathbf{x}) = \frac{1}{2} \cdot \|\mathbf{K}\mathbf{x} - \mathbf{y}\|_2^2 + \lambda \cdot \mathbf{1}^T \cdot \rho\left\{\mathbf{W}\mathbf{x}\right\}, \qquad (13)$$

where $\mathbf{K}$ is a square matrix representing the blur, and $\mathbf{W}$ is a unitary wavelet transform. Their sequential shrinkage method is derived via expectation-maximization, and its structure is very similar to the method proposed in this work. However, their algorithm is restricted to the deblurring case, and cannot be generalized to handle the minimization of the objective posed in (2).

Most related to this paper is the work by Daubechies, Defrise, and De-Mol [17]. While addressing the same objective as posed in (13), their work copes with a general (and thus not necessarily square) operator $\mathbf{K}$. Thus, by defining $\mathbf{x}_W = \mathbf{W}\mathbf{x}$, the above penalty function becomes

$$\tilde{f}_B(\mathbf{x}_W) = \frac{1}{2} \cdot \|\mathbf{K}\mathbf{W}^T\mathbf{x}_W - \mathbf{y}\|_2^2 + \lambda \cdot \mathbf{1}^T \cdot \rho\left\{\mathbf{x}_W\right\}. \qquad (14)$$

Defining $\mathbf{D} = \mathbf{K}\mathbf{W}^T$, their method can handle the very same problem we have posed here. Indeed, their work proposes a sequential shrinkage procedure, very much like the one we propose. However, their way of developing the algorithm is entirely different, leaning on the definition of a sequence of surrogate functions that are minimized via shrinkage. Also, while the resulting algorithms are similar, they are not the same: the norms of the atoms play different roles in the two algorithms; the thresholds chosen in the shrinkage are somewhat different; and the choice of μ is done entirely different.

6. CONCLUSION

We have shown that the heuristic shrinkage has origins in Bayesian denoising, being the first iteration of a in a sequential shrinkage denoising algorithm. This leads to several consequences: (i) we are able to extend the heuristic shrinkage and get better denoising; (ii) we obtain alternative shrinkage algorithms that use the transform and its adjoint, rather than its pseudo-inverse; (iii) the new interpretation may help in addressing the question of choosing the threshold in shrinkage, and how to adapt it between scales; (iv) the obtained algorithm can be used as an effective pursuit for the BPDN for other applications; and (v) due to the close relation to [17], the proposed algorithm can handle general inverse problems of the form (here $\mathbf{KD}$ is the effective dictionary):

$$\tilde{f}(\mathbf{x}_T) = \frac{1}{2} \cdot \|\mathbf{K}\mathbf{D}\mathbf{x}_T - \mathbf{y}\|_2^2 + \lambda \cdot \mathbf{1}^T \cdot \rho\left\{\mathbf{x}_T\right\}. \qquad (15)$$

7. REFERENCES

[1] Donoho, D.L and Johnstone, I.M. (1994) Ideal spatial adaptation by wavelet shrinkage, *Biometrika* Vol. 81 No. 3, pp. 425–455, September.

[2] Donoho, D.L. (1995) De-noising by soft thresholding, *IEEE Transactions on Information Theory*, Vol. 41, No. 3, pp. 613–627, May.

[3] Simoncelli, E.P. and Adelson, E.H. (1996) Noise removal via Bayesian wavelet coring, Proceedings of the *International Conference on Image Processing*, Laussanne, Switzerland. September.

[4] Chambolle, A., DeVore, R.A., Lee, N.-Y., and Lucier, B.J. (1998) Nonlinear wavelet image processing: variational problems, compression, and noise removal through wavelet shrinkage, *IEEE Trans. Image Process.*, Vol. 7, No. 3, 319–335.

[5] Moulin, P. and Liu, J. (1999) Analysis of multiresolution image denoising schemes using generalized Gaussian and complexity priors, *IEEE Transactions on Information Theory*, Vol. 45, No. 3, pp. 909-919, April.

[6] M. Elad, Why simple shrinkage is still relevant for redundant representations?, submitted to the *IEEE Trans. On Information Theory* on January 2005.

[7] Starck, J.-L, Candes, E.J, and Donoho, D.L. (2002) The curvelet transform for image denoising, *IEEE Transactions On Image Processing*, Vol. 11, No. 6, pp. 670–684, June.

[8] Starck, J.-L., Elad, M., and Donoho D.L. (2004) Redundant multiscale transforms and their application for morphological component separation, *Advances in Imaging And Electron Physics*, Vol. 132, pp. 287–348.

[9] Do, M.N. and Vetterli, M. (2002) Rotation invariant texture characterization and retrieval using steerable wavelet-domain hidden markov models, *IEEE Transactions On Multimedia* Vol. 4, No. 4, pp. 517–527, December.

[10] Do, M.N. and Vetterli, M. (2003) Framing pyramids, em IEEE Transactions On Signal Processing, Vol. 51, No. 9, pp. 2329–2342, September.

[11] Portilla, J., Strela, V., Wainwright, M.J, and Simoncelli, E.P. (2003) Image denoising using scale mixtures of gaussians in the wavelet domain *IEEE Transactions On Image Processing*, Vol. 12, No. 11, pp. 1338–1351, November.

[12] Guleryuz, O.G. (2003) Weighted overcomplete denoising, Proceedings of the *Asilomar Conference on Signals and Systems*, Pacific Grove, CA, November.

[13] Starck, J.-L., Candes, E., and Donoho, D.L. (2003) Astronomical image representation by the curvelet transform, *Astronomy and Astrophysics*, Vol. 398, pp. 785–800.

[14] Starck, J.-L., Elad, M., and Donoho, D.L. (2004) Redundant multiscale transforms and their application for morphological component analysis, *Journal of Advances in Imaging and Electron Physics*, Vol. 132, pp. 287–348.

[15] Sardy, S., Bruce, A., and Tseng, P. (2000) Block coordinate relaxation methods for nonparametric wavelet denoising, *J. Comput. Graph. Stat.*, Vol. 9, pp. 361–379.

[16] Figueiredo, M.A. and Nowak, R.D. (2003) An EM algorithm for wavelet-based image restoration, *IEEE Trans. Image Process.* Vol. 12, No. 8, pp. 906–916, August.

[17] Daubechies, I., Defrise, M., and De-Mol, C. (2004) An iterative thresholding algorithm for linear inverse problems with a sparsity constraint, *Communications on Pure and Applied Mathematics*, Vol. LVII, pp. 1413-1457.