

L0-based sparse approximation: Two alternative methods and some applications

Javier Portilla^a and Luis Mancera^b

^aDept. of Images and Vision, Instituto de Óptica, CSIC, Madrid, Spain;

^bDept. of Computer Science and Artif. Intell. (DECSAI), Universidad de Granada, Spain

ABSTRACT

We propose two methods for sparse approximation of images under ℓ_2 error metric. First one performs an approximation error minimization given a ℓ_p -norm of the representation through alternated orthogonal projections onto two sets. We study the cases $p = 0$ (sub-optimal) and $p = 1$ (optimal), and find that the ℓ_0 -AP method is neatly superior, for typical images and overcomplete oriented pyramids. Given that ℓ_1 -AP is optimal, this shows that it is not equivalent in practical image processing conditions to minimize one or the other norm, contrarily to what is often assumed. The second method is more powerful, and it performs gradient descent onto decreasingly smoothed versions of the sparse approximation cost function, yielding a method previously proposed as a heuristic. We adapt these techniques for being applied to image restoration, with very positive results.

Keywords: sparse approximation, ℓ_0 -norm minimization, Parseval frames, image restoration

1. INTRODUCTION

Given an observed vector and a set of vectors defining an overcomplete dictionary, the sparse approximation problem can be stated as minimizing the vector's approximation error as a linear combination of a given number of vectors from the dictionary. This type of problem seems to be very relevant for both living beings and artificial systems that analyze and process stimuli. It very likely plays a role in verbal communication, vision, and, in general, in tasks involving mixed source identification and efficient coding/synthesis. An exact solution of this problem requires a combinatorial method. Thus, some authors have explored more tractable variants.¹ In particular, if we minimize the sum of the absolute values of the coefficients (ℓ_1 -norm) instead of minimizing the number of active vectors (ℓ_0 -norm), then the problem becomes a convex optimization.^{2,3} It has also been shown that, under some constraints, ℓ_0 and ℓ_1 minimizations are equivalent.⁴ However, no one has shown that such equivalence holds under practical conditions for image processing. Here we show the opposite.

On the other hand, during this decade, some other authors have proposed heuristical methods for solving the sparse approximation problem. They use the following empirical observation: if we set to zero the smaller amplitudes of a representation vector and then impose perfect reconstruction (PR) on the thresholded vector, iteratively, we achieve an increasingly sparse representation, until convergence (e.g.⁵⁻⁹). Many more or less ad-hoc variants have been added to this simple scheme, including the important improvement of dynamically modify the threshold applied at each iteration.⁷ Some other authors have proposed similar schemes but using soft-thresholding instead, and they have successfully applied them to image processing tasks (e.g.^{10,11}). However, all the referred methods have in common the lack of a direct theoretical justification.

Here, we firstly present a simple and robust optimization method which minimizes the MSE of the reconstruction from a representation vector (using a Parseval frame) given a ℓ_p -norm and using alternated orthogonal projections onto the ℓ_p -ball and onto the set of vectors with perfect reconstruction. We call it the ℓ_p -AP method. Secondly, we present a theoretical frame that justifies the use of iteratively hard-thresholding and imposing perfect reconstruction until convergence as a means to find a local minimum on a continuous (though equivalent) version of the sparse approximation cost function. Moreover, we also justify the use of a dynamic threshold in terms of applying a common type of deterministic annealing strategy.¹² Though this strategy does not guarantee, in general, the achievement of the global optimum of the original cost function, it greatly helps to

During the development of this work, JP (portilla@io.cfmac.csic.es) and LM (mancera@decsai.ugr.es) were supported by TIC2003-01504 and TEC2006/13845/TCM Spanish Government grants. JP is under the "Ramón y Cajal" Program.

avoid non-favorable local optima. In fact, as we show here, the proposed method achieves close to optimal performance, at least in the vicinity of the theoretical PR asymptote corresponding to use the same number of active coefficients as pixels exist in the image.

2. SPARSE APPROXIMATION IN OVERCOMPLETE FRAMES

Let Φ be a $N \times M$ matrix with $M > N$ and $\text{rank}(\Phi) = N$, representing the synthesis operator of an overcomplete frame. Then, for an observed image $\mathbf{x} \in \mathbb{R}^N$, the system $\Phi \mathbf{a} = \mathbf{x}$ has infinite solutions in $\mathbf{a} \in \mathbb{R}^M$. The minimum ℓ_2 -norm solution, $\mathbf{a}^{LS} = \Phi^\# \mathbf{x}$, is often chosen because it is linear and, thus, it is easy to compute. Here, $\Phi^\#$ is the pseudoinverse of Φ , and corresponds to the analysis operator of the frame. In this paper we consider Parseval frames, where $\Phi^\# = \Phi^T$. We aim to solve the sparse approximation problem using a ℓ_2 -norm cost for the approximation error, which is formulated as follows:

$$\hat{\mathbf{a}}_0(\lambda) = \arg \min_{\mathbf{a}} \{ \|\mathbf{a}\|_0 + \lambda \|\Phi \mathbf{a} - \mathbf{x}\|_2^2 \}, \quad (1)$$

where $\|\mathbf{a}\|_0$ means the ℓ_0 -norm of \mathbf{a} , i.e., the number of its non-zero elements, and $\lambda \in \mathbb{R}^*$ controls the accuracy of the approximation (the larger is λ , the more accurate the solution will be, at the price of reducing the sparseness). Unfortunately, direct thresholding of \mathbf{a}^{LS} does not provide any result close to satisfy Eq. (1) for any λ . Thus, more sophisticated non-linear mechanisms than a single thresholding are required.

3. ALTERNATED PROJECTIONS METHOD

We can generalize Eq. (1) by:

$$\hat{\mathbf{a}}_p(\lambda) = \arg \min_{\mathbf{a}} \{ \|\mathbf{a}\|_p + \lambda \|\Phi \mathbf{a} - \mathbf{x}\|_2^2 \}, \quad (2)$$

where $\|\mathbf{a}\|_p = (\sum_{i=1}^M |a_i|^p)^{\frac{1}{p}}$ denotes the ℓ_p -norm of \mathbf{a} for some norm $p > 0$. Note that, for a fixed p and for each λ value, $\hat{\mathbf{a}}_p(\lambda)$ will have some ℓ_p -norm, R . Then, to solve Eq. (2) for a given λ is equivalent to minimize the approximation error for a given $\|\hat{\mathbf{a}}_p(\lambda)\|_p = R$. Being the ℓ_p -ball $B_p(R) = \{\mathbf{a} \in \mathbb{R}^M : \|\mathbf{a}\|_p \leq R\}$:

$$\hat{\mathbf{a}}_p(\lambda) = \hat{\mathbf{a}}_p(R) = \arg \min_{\mathbf{a} \in B_p(R)} \|\Phi \mathbf{a} - \mathbf{x}\|_2. \quad (3)$$

We have chosen to solve this problem via alternating orthogonal projections¹³ on two sets. The first set is the one including all the representation vectors (synthesis coefficients) that are solutions to the original system of equations, and it is defined as $S(\Phi, \mathbf{x}) = \{\mathbf{a} \in \mathbb{R}^M : \Phi \mathbf{a} = \mathbf{x}\}$. This is an affine subspace, and thus, it is convex. The second set is the ℓ_p -ball, $B_p(R)$, for given p and radius R . This is convex only if $p \geq 1$. When both sets are convex, the projections converge on $B_p(R)$ to the global minimum of the distance to $S(\Phi, \mathbf{x})$ in that ball. Otherwise, the iterations converge to a local minimum. Here we assume that the starting vector for the iterations has a ℓ_p -norm larger than desired (as it happens in practice), thus ensuring that the achieved solution will lie on the boundary of the ℓ_p -ball, and, therefore, that Eq. (3) will hold.

Let $P_C^\perp(\mathbf{v})$ be the orthogonal projection of the vector \mathbf{v} onto a set C . Then, projecting \mathbf{a} onto $S(\Phi, \mathbf{x})$ yields:

$$P_{S(\Phi, \mathbf{x})}^\perp(\mathbf{a}) = \mathbf{a} + \Phi^T(\mathbf{x} - \Phi \mathbf{a}). \quad (4)$$

This requires a synthesis operation, a subtraction in the image domain, an analysis operator, and an addition in the representation domain, as we are adding to the vector being projected the analysis coefficients of its reconstruction error. The expression of the $P_{B_p(R)}^\perp(\mathbf{a})$, on the other hand, depends on p .

Then, for two given p and R values, the ℓ_p -AP method is implemented through the following iterations:

$$\begin{aligned} \hat{\mathbf{a}}_p(R)^{(0)} &= P_{B_p(R)}^\perp(\mathbf{a}^{LS}), \\ \hat{\mathbf{a}}_p(R)^{(n+1)} &= P_{B_p(R)}^\perp(P_{S(\Phi, \mathbf{x})}^\perp(\hat{\mathbf{a}}_p(R)^{(n)})). \end{aligned}$$

In our implementation, we choose to stop the iterations when the accumulated SNR increase of the approximation for the last ten iterations is below 0.02 dB. Now we study the $p = 1$ and $p = 0$ cases in detail.

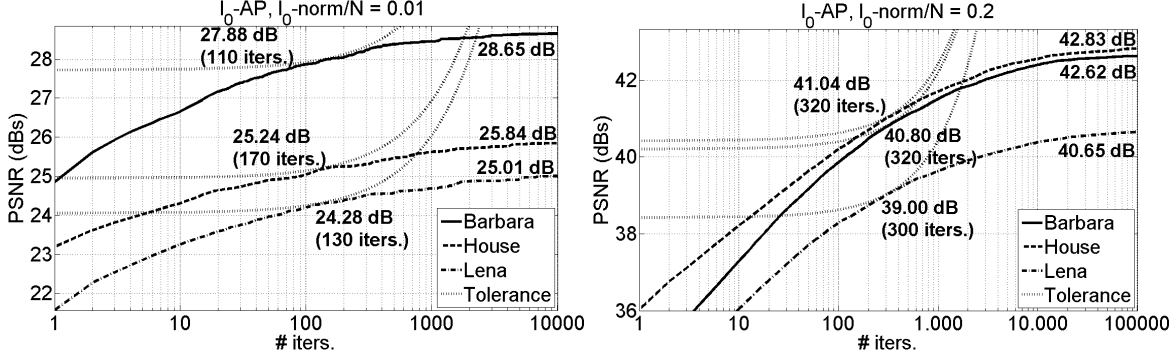


Figure 1. Convergence plot for ℓ_0 -AP using three images and two sparseness levels with 8-scales DT-CWT. The Peak Signal-to-Noise Ratio in dB ($10 \log_{10} \frac{255^2}{MSE}$) at convergence is shown. Dotted line corresponds to the tolerance criterium (a straight line if plotted using a linear abscissa). Tangency points mark the stop of the iterations, whose number is indicated, together with the PSNR at that point

3.1. ℓ_0 -AP

In this case it is straightforward to see that $P_{B_0(R)}^\perp(\mathbf{a})$ is a hard-thresholding preserving the R largest coefficients in amplitude:

$$P_{B_0(R)}^\perp(\mathbf{a}) = \mathbf{a}^h,$$

$$a_i^h = \begin{cases} a_i, & |a_i| > \tau_h(\mathbf{a}, R) \\ 0, & \text{otherwise.} \end{cases}$$

Here, $\tau_h(\mathbf{a}, R)$ is the smallest threshold preserving the $R - n_0$ largest amplitudes, being n_0 the smallest non-negative integer that guarantees that a solution exists ($n_0 = 0$ if there are no repeated amplitude values in the interval of interest). In our implementation, we find this threshold through a golden search.

It is easy to show that this method leads to a local optimum in the image domain. Note that, from Eq. (4):

$$\|\mathbf{a} - P_{S(\Phi, \mathbf{x})}^\perp(\mathbf{a})\|_2 = \|\Phi^T(\mathbf{x} - \Phi\mathbf{a})\|_2 = \|\mathbf{x} - \Phi\mathbf{a}\|_2, \quad (5)$$

where the last step holds because Φ^T is a Parseval frame. Provided that $\hat{\mathbf{a}}_0(R)$ is a local minimum in $B_0(R)$ of the distance to $S(\Phi, \mathbf{x})$, then $\exists \delta > 0$ such that $\forall \mathbf{a} \in B_0(R)$, if $\|\mathbf{a} - \hat{\mathbf{a}}_0(R)\|_2 < \delta$, then $\|\mathbf{a} - P_{S(\Phi, \mathbf{x})}^\perp(\mathbf{a})\|_2 \geq \|\hat{\mathbf{a}}_0(R) - P_{S(\Phi, \mathbf{x})}^\perp(\hat{\mathbf{a}}_0(R))\|_2$. Using Eq. (5) we obtain that $\|\mathbf{x} - \Phi\mathbf{a}\|_2 \geq \|\mathbf{x} - \Phi\hat{\mathbf{a}}_0(R)\|_2$. That is, $\Phi\hat{\mathbf{a}}_0(R)$ is a local minimum in $B_0(R)$ of the Euclidean distance to \mathbf{x} . Regarding the convergence properties, the method quickly evolves towards the solution at the first few iterations, and then it progressively slows down, as shown in Figure 1. Note that the convergence speed also depends on the degree of sparseness being imposed (the higher sparseness, the faster). In this work we have been interested in exploring the absolute performance limits of the studied methods. This has required making a few hundreds iterations for each sparse approximation experiment. In a practical implementation, though, much fewer iterations could have been made with comparable results. Our method can also be seen as an independently developed version of the one described in,⁷ but with the difference that in that work the proposed procedure was not formally described as an optimization method.

3.2. ℓ_1 -AP

It is easy to prove that $P_{B_1(R)}^\perp(\mathbf{a})$ is a soft-thresholding operation, which results from removing the sign of \mathbf{a} , alternatively orthogonally project onto the set of vectors whose elements sum R and onto the positive quadrant, and finally restoring the sign of the original vector.¹⁴ We could use these operations directly to do the projection, but we show here a more efficient alternative, consisting of first computing the required soft-threshold, $\tau_s(\mathbf{a}, R)$, given R , and then applying it to \mathbf{a} . Note that we can directly compute R given \mathbf{a} and the threshold* τ_s :

$$R = \sum_{\Upsilon(\mathbf{a}, \tau_s)} (|a_i| - \tau_s) = \left(\sum_{\Upsilon(\mathbf{a}, \tau_s)} |a_i| \right) - \tau_s \text{card}(\Upsilon(\mathbf{a}, \tau_s)),$$

*For notation clarity, we drop here the dependence of τ_s on (\mathbf{a}, R) .

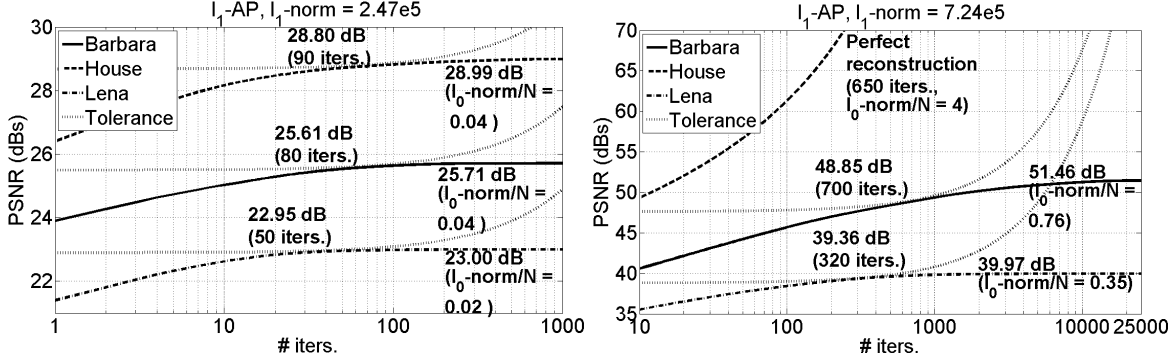


Figure 2. Semi-logarithmic convergence plots for ℓ_1 -AP for three images and two ℓ_1 -norm levels. Details are similar to those of Figure 1. It is also indicated the normalized ℓ_0 -norm of the solution at convergence.

where $\Upsilon(\mathbf{a}, \tau)$ is the set of coefficient indices of \mathbf{a} whose amplitudes are strictly above τ . If we try to solve for τ_s in the previous expression, we get:

$$\tau_s = \frac{\left(\sum_{\Upsilon(\mathbf{a}, \tau_s)} |a_i| \right) - R}{\text{card}(\Upsilon(\mathbf{a}, \tau_s))},$$

which we can not solve directly because the expression on the right still depends on τ_s . However, we can apply instead the following iterative method:

$$\begin{aligned} \tau^{(0)} &= 0, \\ \tau^{(n+1)} &= \frac{\left(\sum_{\Upsilon(\mathbf{a}, \tau^{(n)})} |a_i| \right) - R}{\text{card}(\Upsilon(\mathbf{a}, \tau^{(n)}))}. \end{aligned}$$

This sequence, as it is easy to prove,¹⁴ converges to the unique solution τ_s imposing the desired ℓ_1 -norm R to the soft-thresholded vector.

We next demonstrate that ℓ_1 -AP provides the global optimal solution also in the image domain. First note that $\hat{\mathbf{a}}_1(R)$ is the global minimum in $B_1(R)$ of the Euclidean distance to $S(\Phi, \mathbf{x})$ (because the two involved sets in the projections are convex). Then, $\forall \mathbf{a} \in B_1(R)$, it holds that $\|\mathbf{a} - P_{S(\Phi, \mathbf{x})}^\perp(\mathbf{a})\|_2 \geq \|\hat{\mathbf{a}}_1(R) - P_{S(\Phi, \mathbf{x})}^\perp(\hat{\mathbf{a}}_1(R))\|_2$. Applying Eq. (4) and being Φ^T a Parseval frame, we obtain that $\|\mathbf{x} - \Phi \mathbf{a}\|_2 \geq \|\mathbf{x} - \Phi \hat{\mathbf{a}}_1(R)\|_2$. That is, $\Phi \hat{\mathbf{a}}_1(R)$ is the global minimum for $\mathbf{a} \in B_1(R)$ of the Euclidean distance to \mathbf{x} . Figure 2 illustrates convergence properties of ℓ_1 -AP. As there is no local solutions, the convergence is more regular than with ℓ_0 -AP and the number of iterations needed to converge is much smaller. Note that we have included an example (*House* image in right panel) where perfect reconstruction is achieved.

4. GRADIENT DESCENT ON A GRADUALLY DE-SMOOTHED CRITERION

In this section we present yet another non-convex sparse approximation technique, which is both more powerful and more computationally efficient than the one described in the previous section. This new technique is also fully embedded in a mathematical optimization frame. This technique starts from the simplest optimization question: is it possible to do gradient descent on the criterion to be minimized in Eq.(1)? The answer, due to the discontinuous nature of the ℓ_0 -norm, has to be "no". However, we show here how we can write the same criterion in a different way which allows us to compute iteratively a gradient to descend on. Then, to avoid getting trapped in non-favorable local minima, we apply a type of deterministic annealing technique. Similar non-convex optimization strategies have been proposed in different contexts.^{10, 12, 17}

4.1. An alternative formulation with a continuous cost function

Here we re-formulate the optimization problem of Eq.(1), from current form, which involves an unconstrained discontinuous cost function, to a form based on a constrained continuous cost function. This will allow us to do

gradient descent on the constrained cost function by means of projecting the cost gradient onto the constraint set. Consider the following optimization problem equivalent to Eq. (1):

$$(\hat{\mathbf{a}}, \hat{\mathbf{r}}) = \arg \min_{\mathbf{a}, \mathbf{r}} \{ \|\mathbf{a}\|_0 + \lambda \|\mathbf{r}\|_2^2 \text{ s.t. } \Phi(\mathbf{a} + \mathbf{r}) = \mathbf{x} \} \quad (6)$$

We have used here two terms: a sparse approximation term (\mathbf{a}) and a residual term (\mathbf{r}). The sum of both provides perfect reconstruction of the original. Next we show that the solutions $\hat{\mathbf{a}}$ and $\hat{\mathbf{r}}$ have a very special structure and that we can impose this structure a priori to the vectors involved in the minimization problem, achieving a more convenient form. First note that if $\hat{a}_i \neq 0$, then $\hat{r}_i = 0$ (otherwise we could use $\hat{a}'_i = \hat{a}_i + \hat{r}_i$ and $\hat{r}'_i = 0$, which would decrease the cost function). Also note that \hat{a}_i is either larger in amplitude than $\lambda^{-1/2}$ or zero (otherwise we could use $\hat{a}'_i = 0$ and $\hat{r}'_i = \hat{a}_i$ and decrease so the cost function). Naming $\mathbf{b} = \mathbf{a} + \mathbf{r}$, and taking into account this constraints, we can express \mathbf{a} as $\Theta(\mathbf{b}, \lambda^{-1/2})$, i.e., the result of hard-thresholding \mathbf{b} with threshold $\theta = \lambda^{-1/2}$, and \mathbf{r} as the residual $\mathbf{b} - \mathbf{a}$. As a result of this manipulation, we obtain an equivalent cost function, but which depends on a single vector \mathbf{b} :

$$C(\mathbf{b}) = \sum_{i=1}^M \min \left(\left(\frac{b_i}{\theta} \right)^2, 1 \right). \quad (7)$$

This cost function is counting the terms in \mathbf{b} above the threshold θ (which corresponds to $\|\mathbf{a}\|_0$ in Eq. (6)) and adding the quadratic contribution of all the terms below the threshold (which corresponds to $\lambda \|\mathbf{r}\|_2^2$ in Eq. (6)). Then we obtain $\hat{\mathbf{a}}$ through hard-thresholding from the previous expression:

$$\begin{aligned} \hat{\mathbf{b}} &= \arg \min_{\mathbf{b}} C(\mathbf{b}) \text{ s.t. } \Phi \mathbf{b} = \mathbf{x} \\ \hat{\mathbf{a}} &= \Theta(\hat{\mathbf{b}}, \theta). \end{aligned} \quad (8)$$

5. COST FUNCTION MINIMIZATION

The gradient of the (unconstrained) cost function of Eq. (7) is:

$$\nabla C(\mathbf{b}) = 2\lambda (\mathbf{b} - \Theta(\mathbf{b}, \theta)),$$

and its projection onto the PR affine subspace $\Phi \mathbf{b} = \mathbf{x}$ is $\nabla^{PR} C(\mathbf{b}) = (\mathbf{I} - \Phi^T \Phi) \nabla C(\mathbf{b})$. Because this projection is the null component of the gradient with respect to Φ , it ensures that $\mathbf{b} - \alpha \nabla^{PR} C(\mathbf{b})$ will also provide perfect reconstruction, no matter which α value we use. A necessary (and, in our case, also sufficient) condition for achieving a local minimum of the constrained cost function is that $\nabla^{PR} C(\mathbf{b}^*) = \mathbf{0}$. This is the convergence point of the iterations:

$$\mathbf{b}^{(n+1)} = \mathbf{b}^{(n)} - 2\alpha\lambda (\mathbf{I} - \Phi^T \Phi) (\mathbf{b}^{(n)} - \Theta(\mathbf{b}^{(n)}, \theta)).$$

The choice $\alpha = \frac{1}{2\lambda}$ minimizes the unconstrained cost function of Eq. (7), giving:

$$\mathbf{b}^{(n+1)} = \Theta(\mathbf{b}^{(n)}, \theta) + \Phi^T \Phi (\mathbf{b}^{(n)} - \Theta(\mathbf{b}^{(n)}, \theta)).$$

The previous expression describes a common iterative procedure (e.g.,⁶) for sparsifying a representation vector: hard-threshold the vector coefficients with a fixed threshold and then project the result onto the PR subspace (see Eq. (4)), until convergence. Here we have demonstrated that such a procedure provides a local minimum of the classical sparse approximation criterion function of Eq. (1) using $\lambda = \theta^{-2}$ (with θ the fixed hard-threshold used). However, choosing the α that minimizes in a single step the unconstrained cost function is not, in general, optimal in convergence speed terms. We have experienced that significantly faster convergence is obtained by using instead $\alpha \sim 1.85/(2\lambda)$. Figure 3 shows some convergence plots using fixed thresholds and different α values using 8-scales DT-CWT^{15†}, whose redundancy factor is 4. We can see that, despite the appealing simplicity

[†]Unless indicated, we have used this representation throughout the paper. We have experienced that other Parseval frames, as *Curvelets*,¹⁶ perform qualitatively similar.

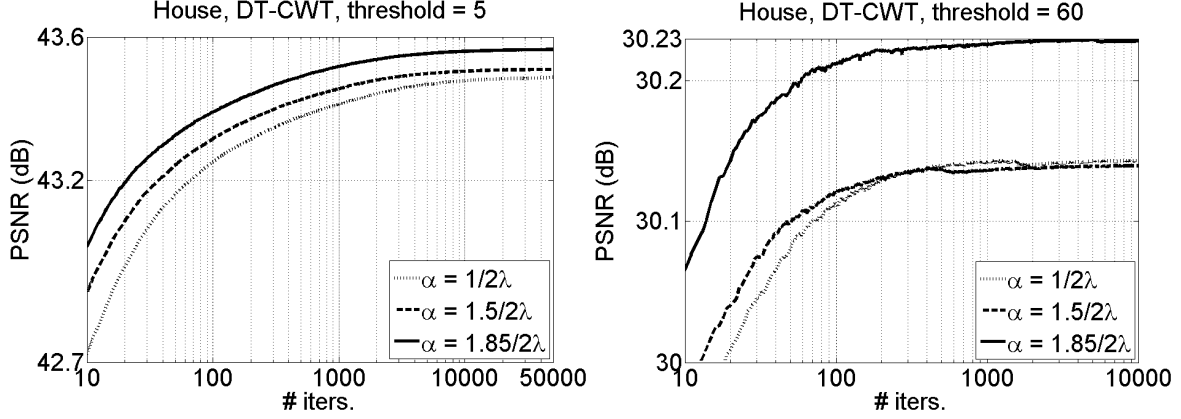


Figure 3. *Left*, convergence plots of ℓ_0 -FT with a low threshold and three different α values. We have used *House* and 8-scales DT-CWT. *Right*, same for a higher threshold.

of this method, doing gradient descent for a given λ value until convergence is reached results far too costly in computational terms. Furthermore, we do not know how good the local minima achieved this way will be. We label this method ℓ_0 -FT (initials come from *Fixed Thresholding*). We propose next an efficient alternative inspired in deterministic global optimization techniques which dramatically reduces the cost of the optimization, also improving the quality of the solution.

5.1. Global minimization: gradual de-smoothing

The cost function associated to Eq. (7), for a given $\theta = \lambda^{-1/2}$, can be written as:

$$C(\mathbf{b}, \theta) = \sum_{i=1}^M (1 - h(b_i/\theta)), \quad (9)$$

where we have defined $h(x) = \max(1 - x^2, 0)$. That is, an inverted parabolic arc centered at zero, going from -1 to 1 and reaching its maximum the value 1 . We can thus rewrite our optimization problem as:

$$\hat{\mathbf{b}}(\theta) = \arg \max_{\mathbf{b}} C'(\mathbf{b}, \theta),$$

$$C'(\mathbf{b}, \theta) = \sum_{i=1}^M h(b_i/\theta) = M - C(\mathbf{b}, \theta).$$

Now it is easy to rewrite the previous expression in terms of an infinitely sharp cost function convolved with a smoothing kernel:

$$C'(\mathbf{b}, \theta) \propto C_\delta(\mathbf{b}) * H(\mathbf{b}/\theta),$$

where "*" represents convolution, $H(\mathbf{b}) = \prod_{i=1}^M h(b_i)$, the proportionality factor is $A(\theta)^{-M+1}$, with $A(\theta) = \int_{-\theta}^{\theta} h(x/\theta) = 4\theta/3$ and $C_\delta(\mathbf{b}) = \sum_{i=1}^M \delta(b_i)$. The $A(\theta)^{-M+1}$ scaling factor results from integrating $H(\mathbf{b}/\theta)$ along all dimensions except for the one of the delta (in which dimension a 1-D convolution is performed), and it is irrelevant in terms of minimization in vector \mathbf{b} . It becomes clear now that it must be faster to find a local optimum when λ is small (θ large, smoother cost function), which corresponds to the case when we give priority to obtain a highly sparse approximation than to obtain a high fidelity of the approximation. Figure 3 shows this effect: convergence is much faster for high thresholds than for low thresholds.

Having a good candidate of the global optimum for a given small λ , we can expect a good result by searching from there to the nearest optimum corresponding to a similar, slightly higher λ . The method consists of fixing a small lambda value, doing gradient descent until reaching convergence, then fixing a slightly higher λ , do gradient descent from the previous convergence point, and so on, until getting to the desired λ value. We call this method

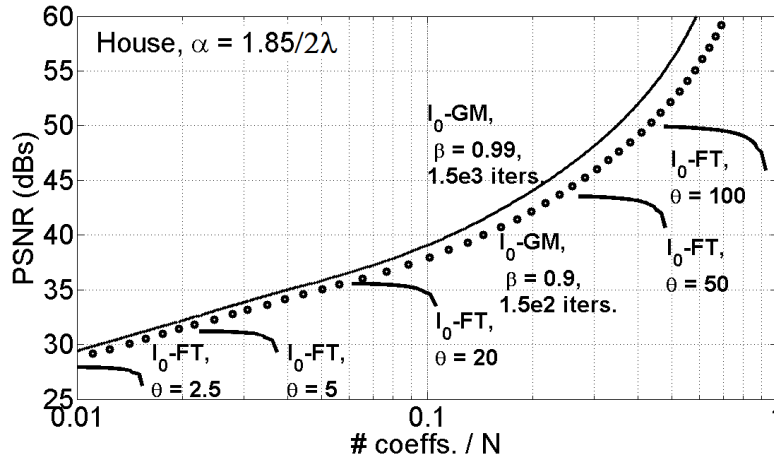


Figure 4. Fidelity-sparseness results of ℓ_0 -GM vs. fixed thresholding (ℓ_0 -FT) for *House* image using several thresholds for ℓ_0 -FT and two β values for ℓ_0 -GM using 8-scales DT-CWT. We show $5 \cdot 10^4$ iterations for each ℓ_0 -FT curve.

ℓ_0 -GM (initials come from *Gradual Minimization*). A simplified, faster, version of this consists of increasing very slowly λ at each single iteration. This corresponds to start with the highest possible threshold and go slowly down at each iteration, until achieving the desired threshold. Note that both methods become the same in the limit when the increase of λ at each iteration becomes infinitesimal. The idea of smoothing a cost function to avoid getting trapped into unfavorable local optima is closely related to other *deterministic annealing* schemes, such as.^{12,17} Some authors had already proposed this idea as an heuristical fast sparsifying algorithm, either using soft-thresholding,¹⁰ or hard-thresholding.^{6,8} However, the referred authors did not address this problem mathematically in optimization terms.

In Figure 4 we have plotted, on the one hand, the convergence trajectories of ℓ_0 -FT for different thresholds (which corresponds to search for local optima doing gradient descent, for fixed λ). On the other hand, we have plotted two trajectories corresponding to exponential decay of the threshold $\theta^{(n)} = \theta^{(0)}\beta^n$ for two different β values. We (as other authors^{7,10}) have experienced in practice that the exponential dynamic threshold update provides a better cost-quality trade-off than other decreasing functions (such as a linear decay).

We can see that following the gradually de-smoothing approach we are not only reducing dramatically the number of iterations, but we also obtain significantly higher fidelity for the same sparseness level. In Figure 5 (left) we show a family of fidelity-sparseness curves for different β 's. The fidelity is expressed as Peak Signal-to-Noise Ratio (PSNR), defined as $10 \cdot \log_{10}(\frac{255^2}{MSE})$, where MSE is the average mean square error for a set of five standard 256×256 images (*Einstein*, *Barbara*, *Boat*, *Lena*, *Peppers*)*. We can appreciate how the obtained fidelity values get close to ideal (they become asymptotical around $R = N$) as β gets close to $\frac{1}{2}$.

5.2. A single solution for all λ values?

If we optimize for a set of λ values, we end up with multiple solutions. Which criterion should we follow to choose a particular solution? Would our choice optimize the performance for a given λ at the cost of sacrificing the performance for other λ values? The (somehow surprising) answer is "not necessarily". We have explained before how, to achieve a good solution for a certain $\lambda_i > \lambda_j$ it is good to "pass by" the λ_j solution, and, from there, refine it until reaching the λ_i solution. This seems to imply that good solutions for higher λ 's should be reasonably good also for lower λ values. In Figure 5 (right) we have shown the fidelity-sparseness curves obtained applying the cost function, for the different λ values, to the solution obtained with the highest λ value

*We have cropped *Boat* and *Barbara* to 256×256 starting at pixel (200,100) and (150,50) respectively. The black border surrounding *Einstein* has been removed by repeating the neighbor lines.

[‡]As pointed out before, achieving the global optima is much less likely for low sparseness levels than for high sparseness levels, because the number of local optima increases rapidly with λ . This makes the asymptotical behavior for $R = N$ even more significant.

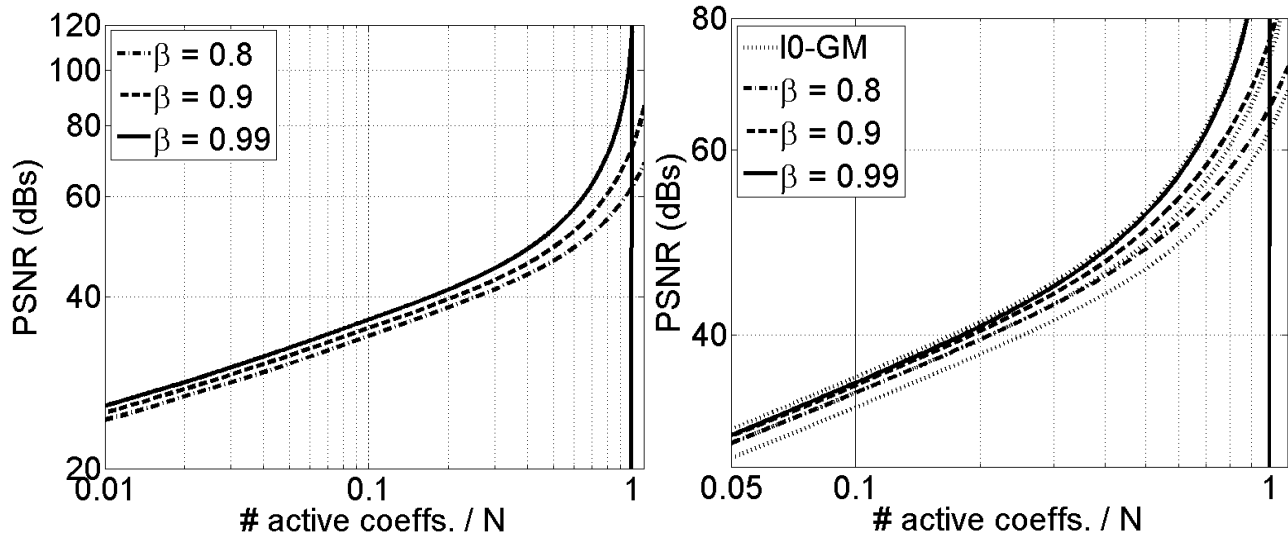


Figure 5. *Left*, averaged fidelity-sparseness results in our test set using ℓ_0 -GM and different β values with 8-scales DT-CWT. *Right*, reconstruction quality from the largest coefficients of the optimized result after the de-smoothing process of ℓ_0 -GM for different β values. Dotted curves correspond to those in left panel. Ordinate axis has been re-scaled for visibility. See more details in the text.

in the considered range. As we can see, for $\beta < 0.99$ the results not only have not worsened, but they have even improved. This means that a single sparseness optimization solution may be enough for all considered λ factors. The idea of using the solution of decreasing a threshold until reaching a very small value to make a representation sparse is not new.^{8,10} However, to the best of our knowledge, nobody has analyzed the reasons why this solution is so favorable in terms of minimizing an objective criterion.

6. A COMPARISON OF SPARSE APPROXIMATION RESULTS

Figure 6 (left) shows a comparison of the average fidelity vs. sparseness performance achieved by ℓ_1 -AP and ℓ_0 -AP. We observe that ℓ_0 -AP performs significantly better, except for the very low (of little practical use) sparseness regime. We have also plotted the performance achieved by MSE-optimizing a posteriori¹⁴ the active coefficients obtained by ℓ_1 -AP. We label this method ℓ_1 -AP+OP. Now the results improve slightly those of ℓ_0 -AP.

Figure 6 (right) compares the ℓ_0 -GM, using an exponential decay for the threshold, to our ℓ_0 -AP method and also with ℓ_1 -AP+OP and ℓ_0 -FT. The performance increase with respect to our previous result is very noticeable, whereas our two methods perform significantly better than ℓ_0 -FT. We see that ℓ_0 -GM clearly outperforms the ℓ_1 -norm post-optimized method for the whole sparseness range. Note also that the new method requires much less computation and no prior choice of the sparseness level. Figure 7 shows a visual comparison of the methods using $0.04 \cdot N$ DT-CWT coefficients. An important difference with the alternated projections method is that it allows us to sweep "on the fly" all the sparseness levels, instead of running long iterations for each one. Table 1 shows the numeric results from Figure 6.

7. APPLICATION TO IMAGE RESTORATION

Consider now that our observation $\mathbf{y} \in \mathbb{R}^N$ is incomplete. For instance, some pixels, some color components in a trichromatic picture, some bits, etc., are missing. We assume that from the observation we can discriminate which elements of the image are lost[§], and so we can replicate the associated degradation, that we will term $f_{\mathbf{y}}(\mathbf{x})$. We define $R(\mathbf{y})$ as the set of images consistent with that observation, $R(\mathbf{y}) = \{\mathbf{x} \in \mathbb{R}^N : f_{\mathbf{y}}(\mathbf{x}) = \mathbf{y}\}$. Finding the orthogonal projection onto the consistency set is trivial for a wide class of "deterministic a posteriori" degradations, by simply forcing the preserved components to have the observed values.

[§]This is not always a realistic assumption, but it seems a reasonable starting point for the model.

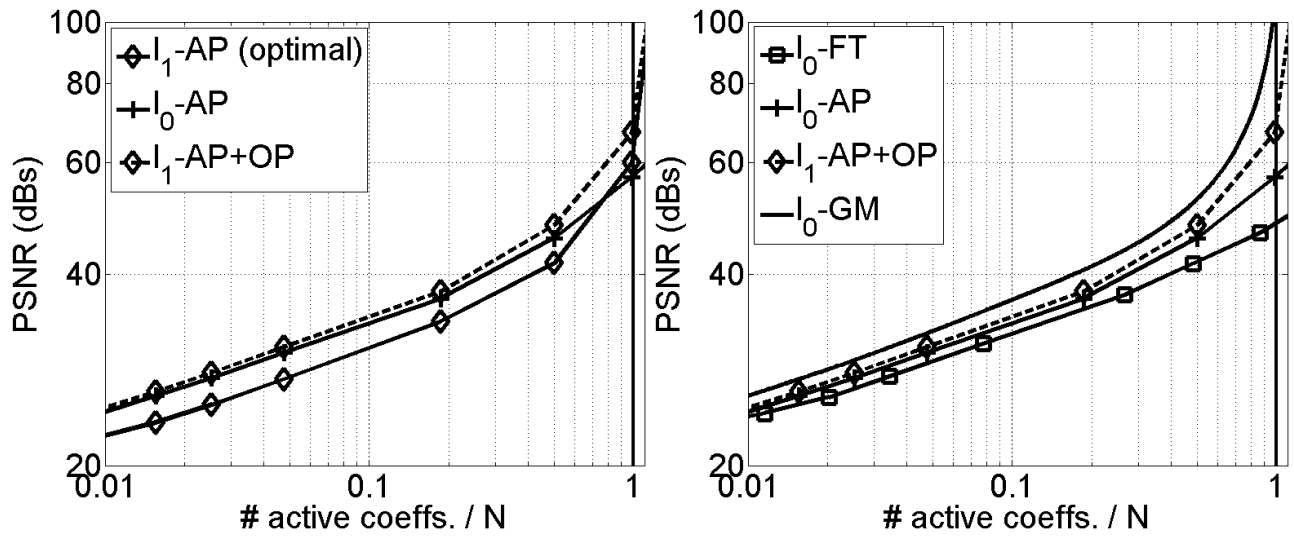


Figure 6. *Left*, averaged compaction results in the test set for our alternated projections-based methods using DT-CWT (8-scales). *Right* Averaged compaction results in the test set for our non-convex optimization method.

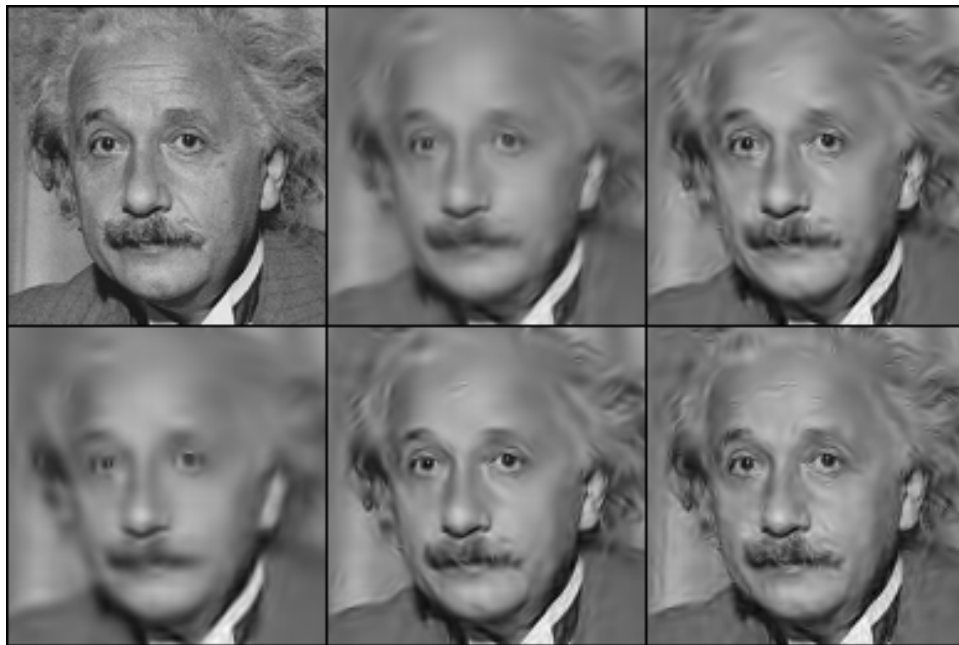


Figure 7. 64×64 crop of the 8-scales DT-CWT reconstruction of *Einstein* image using $0.04 \cdot N$ active coefficients for the methods presented in the paper. *Top left*, original image. *Bottom left*, reconstruction from analysis vector (29.09 dB). *Top center*, reconstruction using l_1 -AP (29.70 dB). *Bottom center*, result of l_1 -AP+OP (32.38 dB). *Top right*, result using l_0 -AP (31.97 dB). *Bottom right*, using l_0 -GM (33.20 dB).

		Normalized ℓ_0 -norm (R/N)					
Image	Method	8.68e-3	2.54e-2	4.76e-2	1.86e-1	5.02e-1	9.87e-1
Barbara	ℓ_0 -FT	23.16	27.25	29.89	35.68	44.12	48.56
	ℓ_0 -AP	24.89	28.67	31.89	40.39	47.63	55.64
	ℓ_1 -AP+OP	25.24	29.38	32.60	41.57	51.22	>100
	ℓ_0 -GM	26.18	30.47	33.93	43.05	54.93	>100
House	ℓ_0 -FT	24.08	30.35	32.91	38.14	44.61	50.75
	ℓ_0 -AP	27.33	31.50	34.16	38.40	49.25	58.98
	ℓ_1 -AP+OP	27.75	32.01	34.73	39.20	51.25	62.31
	ℓ_0 -GM	28.85	33.18	35.65	43.47	56.12	>100
Boat	ℓ_0 -FT	16.82	24.74	27.09	32.07	39.30	45.65
	ℓ_0 -AP	21.57	25.76	27.86	33.52	43.29	58.91
	ℓ_1 -AP+OP	21.96	26.12	28.37	34.25	45.02	>100
	ℓ_0 -GM	23.77	27.32	30.13	38.09	50.10	>100
Lena	ℓ_0 -FT	17.27	26.22	28.93	34.33	42.30	48.21
	ℓ_0 -AP	23.62	27.20	30.09	36.85	45.66	55.51
	ℓ_1 -AP+OP	24.17	27.96	30.95	37.79	47.95	63.72
	ℓ_0 -GM	25.19	29.29	32.34	40.97	52.97	>100
Peppers	ℓ_0 -FT	16.76	25.15	27.93	33.98	41.19	48.66
	ℓ_0 -AP	22.61	26.39	29.17	37.20	44.84	57.09
	ℓ_1 -AP+OP	22.81	26.85	29.74	38.41	47.13	>100
	ℓ_0 -GM	24.03	28.43	31.57	39.93	51.63	>100

Table 1. Fidelity (PSNR, in dB) for several sparseness levels, images and sparse approximation methods. We have used 8-scales DT-CWT. Bold numbers indicate the method giving the best approximation. Columns correspond, respectively, to 569, 1662, 3120, 12221, 32905 and 64682 active coefficients. We have directly extracted the PSNR values from experiments, except those for ℓ_0 -FT, which have been linearly interpolated.

7.1. Estimation using ℓ_p -AP

Sparseness (expressed with some ℓ_p -norm with $0 \leq p \leq 1$) has been widely used as a prior information for solving inverse problems (e.g.,^{2,18,19}). We can pose our restoration problem as:

$$\hat{\mathbf{a}} = \arg \min_{\mathbf{a}} \{ \|\mathbf{a}\|_p + \lambda \|P_{S(\mathbf{y})}^\perp(\mathbf{a}) - \mathbf{a}\|_2^2 \},$$

where $S(\mathbf{y}) = \{\mathbf{a} \in \mathbb{R}^M : f_{\mathbf{y}}(\Phi\mathbf{a}) = \mathbf{y}\}$ is the set of vectors whose reconstruction is consistent with the observation. As our aim is to achieve an estimation which is consistent with \mathbf{y} , it seems natural to set λ to infinite, so we are minimizing the ℓ_p -norm while forcing $\hat{\mathbf{a}} \in S(\mathbf{y})$. Thus, $\hat{\mathbf{a}}$ will have some ℓ_p -norm, R . Then, given \mathbf{y} , p and R , the problem at hand is equivalent to:

$$\hat{\mathbf{a}} = P_{S(\mathbf{y}) \cap B_p(R)}^\perp(\Phi^T \mathbf{y}).$$

We can use the ℓ_p -AP method to solve this problem just substituting $S(\Phi, \mathbf{x})$ by $S(\mathbf{y})$.

7.2. Estimation using ℓ_0 -GM

7.2.1. Analysis-sense vs. Synthesis-sense sparseness

Though perfectly legitimate (and reasonably successful) one could object the lack of a direct empirical basis for the use of *sparse synthesis priors*. Usual Bayesian approaches to image restoration are based on building image priors that reflect the typical behavior of the signals in many previous observations. But, is the sparseness, in the sense we have used it through this work, *directly* observable? The answer is "no". Whereas the responses to a signal of a set of linear kernels are directly observable, that is not the case for the synthesis coefficients of an optimal sparse representation. To differentiate among the two sparseness concepts, we call the latter

synthesis-sense sparseness (SS), as opposed to the analysis sense sparseness (AS). Elad et al. have addressed the theoretical properties and the different uses we can give to both concepts.²⁰

Following this discussion, it seems conceptually more correct to use a statistical prior based on direct observations (AS) rather than using a statistical description of the image based on non-observable optimally sparse synthesis coefficients. The first approach is a natural extension of many previous works (e.g.,²¹). Moreover, some authors have implemented AS-based methods in image processing, with very positive results (e.g.^{10,11}). We have experienced a generally better performance of AS vs. SS for image restoration, as previously shown in.²⁰

7.3. Analysis-sense sparseness restoration

When we talk about sparse linear responses, strict sparseness is in general no longer attainable. We can, instead, consider that typical responses to natural images concentrate on a small proportion of coefficients a large proportion of the total signal energy, and, thus, model the linear representation as a strictly sparse vector representing the highest amplitude responses (\mathbf{a}), plus a Gaussian correction term (\mathbf{r}). Then, if we define $S_A(\mathbf{y})$ as the set of linear responses whose reconstruction is consistent with the observation, $S_A(\mathbf{y}) = \{\mathbf{b} \in R^M : \exists \mathbf{x} \in R(\mathbf{y}), \Phi^T \mathbf{x} = \mathbf{b}\}$, we look for:

$$(\hat{\mathbf{a}}, \hat{\mathbf{r}}) = \arg \min_{\mathbf{a}, \mathbf{r}} \{\|\mathbf{a}\|_0 + \lambda \|\mathbf{r}\|_2^2 \text{ s.t. } (\mathbf{a} + \mathbf{r}) \in S_A(\mathbf{y})\}. \quad (10)$$

Following a completely parallel path to the one we used for the sparse approximation case, we end up with an expression depending only on $\mathbf{b} = \mathbf{a} + \mathbf{r}$ that is equivalent to that of Eq. (8) substituting the constraint set by $S_A(\mathbf{y})$. Note that now the constraint set is no longer affine in general, and, thus, we have to consider its curvature by projecting the gradient onto its tangent hyperplane at every frontier point \mathbf{b} . This projection can be computed as the limit:

$$\nabla^{S_A(\mathbf{y})} C(\mathbf{b}) = \lim_{\alpha \rightarrow 0} \frac{P_{S_A(\mathbf{y})}^\perp(\alpha \nabla C(\mathbf{b}))}{\alpha},$$

where $P_{S_A(\mathbf{y})}^\perp$ is the orthogonal projector on $S_A(\mathbf{y})$. The gradient descent method is:

$$\mathbf{b}^{(n+1)} = \mathbf{b}^{(n)} - \alpha \nabla^{S_A(\mathbf{y})} C(\mathbf{b}). \quad (11)$$

However, in practice it is more convenient to use in the estimation loop a more simple computation:

$$\mathbf{b}^{(n+1)} = P_{S_A(\mathbf{y})}^\perp \left(\mathbf{b}^{(n)} - \alpha \nabla C(\mathbf{b}) \right)$$

which ensures that the updated vector belongs to the consistency set $S_A(\mathbf{y})$ for any α value. This updating rule is identical to that of Eq. (11) if the projection is linear. For calculating $P_{S_A(\mathbf{y})}^\perp$ we realize that Φ^T is a Parseval frame (so it preserves the Euclidean norm) and, as a consequence:

$$P_{S_A(\mathbf{y})}^\perp(\mathbf{b}) = \Phi^T P_{R(\mathbf{y})}^\perp(\Phi \mathbf{b}).$$

Because of the similar structure of the described minimization problem for the sparse approximation (Eq. (6)) and for the AS restoration problem (Eq. (10)), the same results on global minimization strategies apply now. This implies in practice that significantly better restoration performance is achieved by using an exponentially decaying threshold until reaching the desired λ , than using a fixed threshold corresponding to that λ . We have experienced that the optimal final θ in our described optimization method for restoration application is usually close to zero (as in^{10,11}), so an arbitrarily small threshold value can be used in practice as a stopping criterion for the iterative method.

7.4. Some examples

For each example, we only show the results from the best of our methods. Figure 8 shows an example of performance of ℓ_0 -AP to the removal of spatial quantization artifacts. We compare the performance of ℓ_1 -AP and ℓ_0 -AP using both 8-scales DT-CWT and 6-scales Curvelets. We can see that ℓ_1 -AP results are too smooth, even decreasing the PSNR with respect to the degraded observation. On the other hand, performance of ℓ_0 -AP

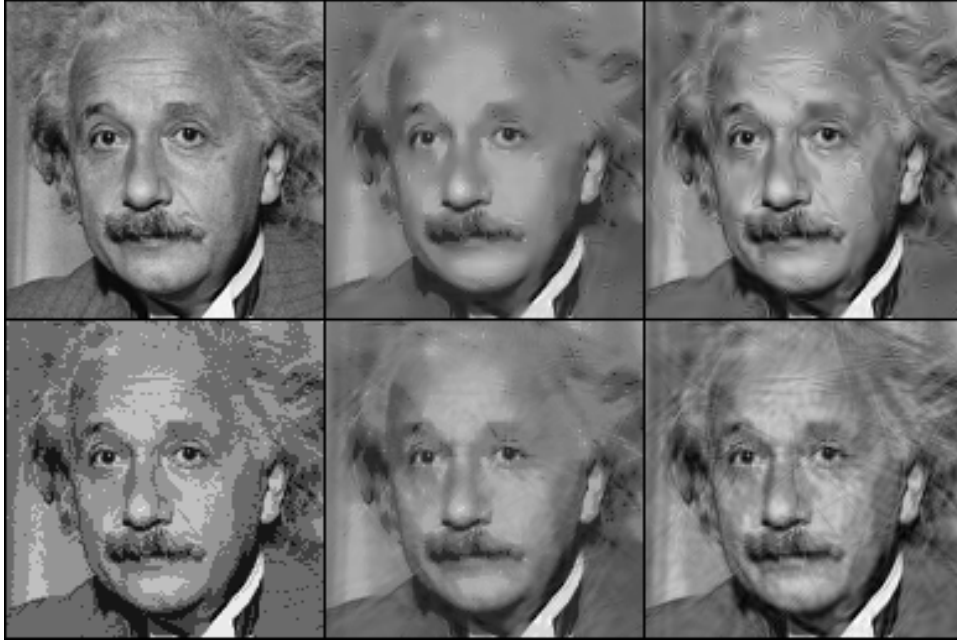


Figure 8. Application of ℓ_1 -AP and ℓ_0 -AP to removing quantization artifacts. *Top left*, Einstein image, cropped to 128×128 starting at (71,41). *Bottom left*, 3-bits quantized observation (29.58 dB). *Top center*, ℓ_1 -AP results using 8-scales DT-CWT (28.14 dB). *Bottom center*, same result using 6-scales Curvelets (28.61 dB). *Top right*, ℓ_0 -AP results using DT-CWT (31.24 dB). *Bottom right*, same using Curvelets (31.29 dB).

is much more satisfactory. Our choice of R (see Section 7.1) is useful for this example, but it will vary from one application to another (see, e.g., in-painting application in¹⁸). Figure 9 shows an example of application of ℓ_0 -GM to filling in missing pixels, where it outperforms dynamic soft-thresholding.¹¹ Figure 10 shows application of ℓ_0 -GM to super-resolution, where observation is obtained by averaging 2×2 blocks and subsampling. Our method reconstructs significantly better both edges and smooth surfaces than pixel replication and linear interpolation. Figure 11 is an example of application to de-mosaicing. Here, ℓ_0 -GM compares favorably not only to basic bilinear interpolation but also with an advanced recently proposed method.²³ For this application, we sparsify the YUV channels of the image. In order to favor the chromatic regularity of the image, we have used an extra projection that minimizes the high frequency components of the U and V channels by means of setting them to zero at each iteration (2 highest frequency scales of 3-scales Curvelets).

8. CONCLUSIONS

We have studied the sparse approximation of images using Parseval frames from a practical point of view through formal optimization methods. Firstly, we have developed a method that minimizes the approximation error given a ℓ_p -norm of the representation using alternated orthogonal projections. We find that using $p = 0$ (suboptimal) is clearly superior to using $p = 1$ (optimal) for the sparse approximation problem. This shows that it is not equivalent in practical image processing conditions to minimize one or the other norm. However, we have also shown that the selection of active coefficients resulting from ℓ_1 -AP is generally better than the one made by ℓ_0 -AP. The second method performs gradient descent onto decreasingly smoothed versions of the sparse approximation cost function. Similar methods were previously proposed as heuristics, but here we justify it in standard optimization terms. This method gets close to current state-of-the-art in sparse approximation. It is easy to adapt both methods for doing restoration when the degradation is a localized information loss. We have shown successful examples for missing bits, pixels, chromatic components, and averaging plus downsampling.

REFERENCES

1. J.F. Claerbout, F. Muir, “Robust modeling with erratic data,” *Geophysics* **38**(51), p. 826.



Figure 9. Application of ℓ_0 -GM to in-painting. *Top left*, original *Barbara* image, cropped to 128×128 starting at (301,50). *Top right*, degraded observation. *Bottom left*, result using dynamic soft-thresholding¹¹ (PSNR of the estimated pixels is 23.31 dB). *Bottom right*, our result (23.61 dB). Representation used is 6-scales Curvelets.

2. S. Chen, D. Donoho, and M. Saunders, "Atomic decomposition by Basis Pursuit," *SIAM* **20**(1), p. 33.
3. J.A. Tropp, "Just relax: Convex programming methods for identifying sparse signals in noise," *Trans. Inf. Theory*, **52**(3), p. 1030.
4. D. Donoho, "For most large undetermined systems of linear equations the minimal ℓ_1 -norm solution is also the sparsest solution," *Tech. Rep.*, 2004.
5. S. Fischer and G. Cristobal, "Minimum entropy transform using Gabor wavelets for image compression," in *Int'l. Conf. on Image Anal. and Proc.*, (Palermo, Italy), 26-28 Sep. 2001.
6. T.H. Reeves and N. Kingsbury, "Overcomplete image coding using iterative projection-based noise shaping," in *Int'l. Conf. on Image Proc.*, (Rochester, NY), 23-25 Sep. 2002.
7. N. Kingsbury and T. Reeves, "Redundant representation with Complex Wavelets: How to achieve sparsity," in *Int'l. Conf. on Image Proc.*, (Barcelona, Spain), 14-18 Sep. 2003.
8. O.G. Guleryuz, "Nonlinear approximation based image recovery using adaptive sparse reconstructions and iterated denoising-part I: Theory," *IEEE Trans. on Image Proc.*, **15**(3), p. 539, Mar. 2006.
9. R.M. Figueras i Ventura and E.P. Simoncelli, "Statistically driven sparse image approximation," in *Int'l. Conf. on Image Proc.*, (San Antonio, TX), 16-19 Sep. 2007.
10. J.L. Starck, "Morphological Component Analysis," in *Proc. of the SPIE*, **5914**, (San Diego, CA), Aug. 2005.
11. M. Fadili and J. Starck, "EM algorithm for sparse representation-based image inpainting," in *Int'l. Conf. on Image Proc.*, (Genoa, Italy), 11-14 Sep. 2005.
12. A. Blake and A. Zisserman, "Graduated Non-Convexity," in *Visual Reconstruction*, MIT Press, Cambridge, MA, 1987.
13. D.C. Youla, "Generalized image restoration by the method of Alternating Orthogonal Projections," *IEEE Trans. on Circuits and Syst.*, **CAS-25**(9), 1978.

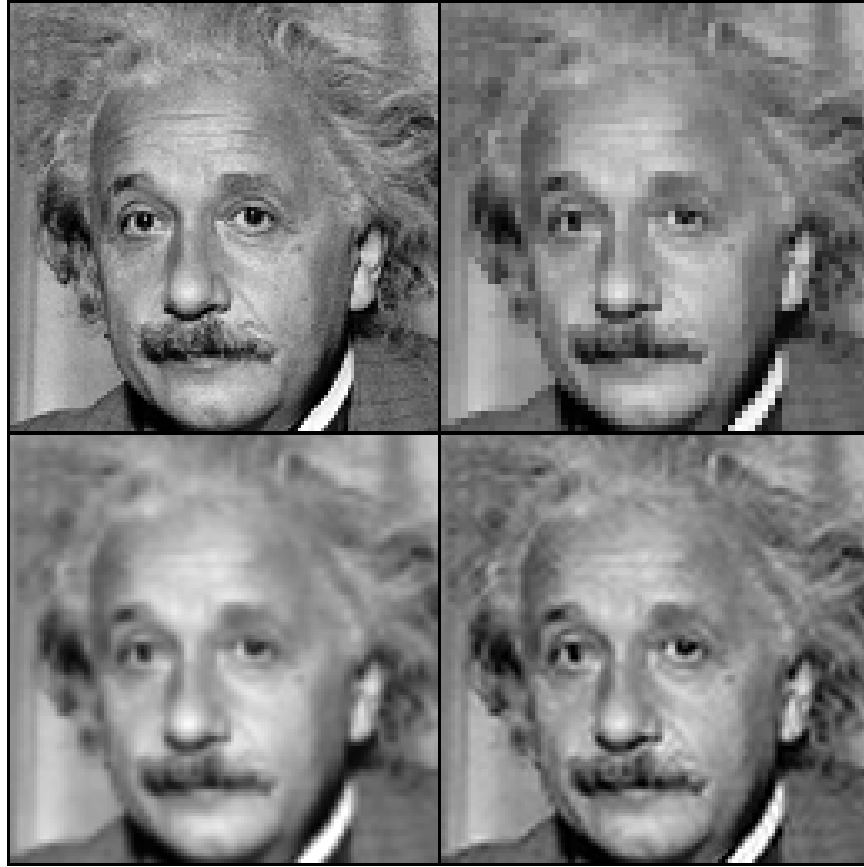


Figure 10. Application of ℓ_0 -GM to super-resolution. *Top-left*, original image cropped to 128×128 starting at (71,31). Observation is obtained by averaging 2×2 original blocks and subsampling. *Top-right*, result from replicating observed pixels (30.49 dB). *Bottom-left*, linear interpolation (30.99 dB). *Bottom-right*, ℓ_0 -GM (32.63 dB).

14. L. Mancera, J.A. Guerrero-Colon and J. Portilla, "Sparse approximation via orthogonal projections: beyond greed and convexity," *Submitted to IEEE Trans. Image Proc.*, Jul. 2007.
15. N. Kingsbury, "Complex Wavelets for shift-invariant analysis and filtering of signals," *J. of App. and Comp. Harmonic Anal.*, **10**(3), p. 234, May 2001.
16. E. Candes and D. Donoho, "Curvelets - a surprisingly effective nonadaptive representation for objects with edges," in *Curves and Surfaces IV*, L. S. et al., ed., Vanderbilt University Press, Nashville, TN, 1999.
17. S. Kirkpatrick, C. Gelatt, and M. Vecchi, "Optimization by Simulated Annealing," *Science* **220**(4598), p. 671.
18. L. Mancera and J. Portilla, "L0-norm-based representation through Alternate Projections," in *Int'l. Conf. on Image Proc.*, (Atlanta, GE), 8-11 Oct. 2006.
19. M.A.T. Figueiredo, J.M. Bioucas-Dias and R.D. Nowak, "Majorization-Minimization algorithms for wavelet-based image restoration," *Submitted*, 2007.
20. M. Elad, P. Milanfar, R. Rubinstein, "Analysis versus synthesis in signal priors," *Inverse Problems*, **23**, p. 947, 2007.
21. E.P. Simoncelli and E.H. Adelson, "Noise removal via Bayesian wavelet coring," *Int'l. Conf. on Image Proc.*, (Laussane, Switzerland), Sep. 1996.
22. 40 scanned images. "Eastman Kodak(c) photographic color image database", 1993.
23. X. Li, "Demosaicing by successive approximation," *IEEE Trans. on Image Proc.* **14**(3), p. 370.

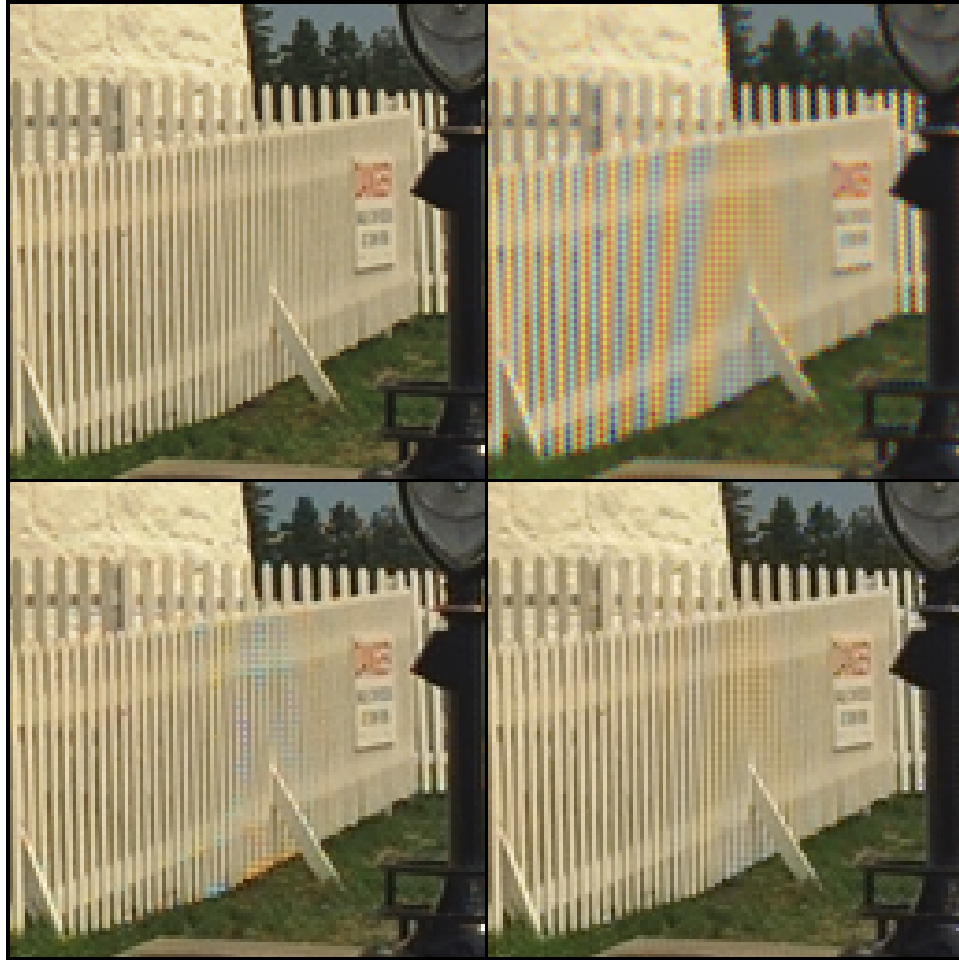


Figure 11. Application of ℓ_0 -GM to de-mosaicing. *Top left*, Image 15 from Eastman Kodak database.²² *Top right*, interpolation using Bilinear method. PSNR red channel (R): 22.17 dB, PSNR green channel (G): 27.30 dB, PSNR blue channel (B): 21.29 dB. *Bottom left*, result from Li's method²³ (PSNR R,G,B: 34.60, 38.18 and 32.86 dB). *Bottom right*, our result (PSNR R,G,B: 35.20, 38.25 and 33.97 dB). Images are cropped to 128×128 starting at pixel (228,323) for artifacts visibility. Representation used is 6-scales Curvelets.