TR # 1153

We propose methods of both slow and rapid post-processing of signals for erasure of artifacts that arise in the process of thresholding and quantization. We use wavelets as tools to define constraints, and variational functionals as measures of complexity of signals. The methods come from analyses of different possibilities of blending variational calculus and wavelet multiresolution in ways that appear to be natural. The methods allow extensions to Ritz-Galerkin type algorithms for simulation of solutions to differential equations, which will be presented elsewhere.

Combining variational calculus and wavelets for image enhancement

R. Coifman[1] A. Sowa[2]

Research Report YALEU/DCS/RR-1153
June 8, 1998

This research was supported by DARPA/AFOSR F49620-97-1-0011.

Keywords: image processing, image enhancement, denoising, data compression, wavelet optimization, wavelet shrinkage, wavelet relaxation
AMS subject classifications: 42C15, 68U10, 65T99, 65K10

[1] Departments of Computer Science and Mathematics, Yale University, New Haven, CT 06520
[2] Department of Mathematics, Yale University, New Haven, CT 06520

1 Introduction

Given an imperfectly described signal, it is often the case that a few of its parameters are given with good precision whereas other parameters are known only vaguely or are *a priori* essentially unknown. If one believes however that the unknown parameters are somehow correlated with the known ones, then it is reasonable to try to extrapolate the unknown parameters from the available ones. This involves exploitation of additional, external principles of our choice — the ones that are believed to express relations between the two groups of parameters.

A good example of such a problem is provided by the contemporary wavelet and wavelet packet-based techniques for denoising and compression of one- and two-dimensional signals. In one form or another, all these procedures use the following two steps. First, the signal is represented in some wavelet packet basis. In the second step, the coefficients of the representation are purposefully *altered* either to suppress noise or compress the data, or both. This is usually done by *thresholding* or *quantization* of coefficients. Thresholding in its most basic form means setting those coefficients that are smaller in absolute value than a certain threshold to zero. Quantization in its most basic form means rounding the coefficients off to the nearest multiple of a chosen unit (the quanta).

Efficient as these methods can be, they will unavoidably introduce undesirable artifacts to the processed signals. The artifacts become more and more manifest as the thresholding or quantization procedures get coarser.

Many methods of enhancement of noisy images using partial differential equations have been recently developed and tested. Among them, the method of Perona and Malik, the method of Alvarez, Lions and Morel, and the method of Rudin and Osher are perhaps representative (cf. [5] for a review and further references). One common feature of all of these methods is that they use (nonlinear) heat flows with the noisy image as the initial condition. Since, generally speaking, one expects heat flow to be a regularizing, smoothing process, it is necessary to take special precautions to prevent smoothing of the edges in the images. In order to achieve this, the authors mentioned above design nonlinear terms, that enter the driving force sides of their equations, to detect edges and slow down or stop dispersion in their direct vicinity. Let us point out a few shortcomings of these methods.

- Heat flow is a slow process.

- Even if the methods converged very rapidly, so that it would suffice to perform just a few steps in their discretized form, the nonlinearity would cause each step to be computationally expensive.
- None of these methods is applicable to erasure of artifacts that arise in the process of quantization or thresholding of images with the use of singular wavelets. Indeed, these procedures introduce artifacts in the form of edges. Therefore, any method of cleaning must have a way of distinguishing between the edges that are features of the original image and those that have been introduced artificially.

One approach to restoration of signals that have been subject to damage by thresholding would be to devise a method for extrapolation of coefficients below the threshold from those that remain unchanged above it. We will present here two mathematical treatments of this problem. The first one may be seen as a generalization of the method introduced by Y. Bobichon and A. Bijaoui in their paper [2]. We believe that the second method is entirely new. It is in principle based on detection of wavelets (which are looked upon as "undercurrent" signals themselves) by matching with the use of specially designed correlation measures.

Since the signals to which we want to apply these techniques are finite dimensional, we will not discuss issues related to the infinite dimensional situation. We want to emphasize, however, that the scheme for minimization of energy functionals suggested at the end of the next paragraph is interesting as a tool for simulations of PDE's and naturally "lifts" to the infinite dimensions via the Galerkin method. Of course this poses many questions that have to be resolved. We will elaborate on the subject elsewhere.

2 The two approaches.

The algorithms presented in this section are (in their most interesting versions) nonlinear, and as such usually quite slow. Because of this we will limit our discussion to one-dimensional signals. We emphasize, however, that two-dimensional extensions of these algorithms are straightforward and, even though rather slow, they are still very interesting from the point of view of many particular applications. In the next section we will present a rapid algorithm that is a special version of algorithms presented here as the second type.

Methods of the First Kind

Suppose now we are given two (topological) functional spaces B_1 and B_2, and a (linear) transform $T : B_1 \to B_2$. We do not specify any properties of T, B_1, B_2 until later. However, we will think of this transform as a wavelet (wavelet-packet) type transform. In addition, we think of B_1 as the (physical) space of signals, and B_2 as the phase space. Furthermore, let $\Phi : B_1 \to R$ be a (energy) functional defined over B_1 with some regularity that will be requested later. Finally, let $M \in B_2$ be a constraint (subset) in B_2. We are interested in the following problem: Find a $u_0 \in B_1$ such that

$$\Phi(u_0) = min_{u \in B_1} \Phi(u),$$

subject to the condition

$$Tu_0 \in M.$$

Let, in addition, $P_M : B_2 \to M$ denote the orthogonal projection into M. If the functional Φ is sufficiently regular to posses (at least formally) gradient $grad\Phi$, one might attempt to solve this problem by the steepest descent method, i.e. by solving

$$\frac{\partial w}{\partial t} = -P_M T grad\Phi(T^{-1}w),$$

for $w \in B_2$ with the initial condition $T^{-1}w(0) = u_1 \in T^{-1}M$. Let $\chi_{[-Q,Q]}$ denote the characteristic function of the interval $[-Q, Q]$. In the case of thresholding the formula we do actually convert into an algorithm is

$$\frac{\partial w}{\partial t} = -\chi_{[-Q,Q]}(w) P_M T grad\Phi(T^{-1}w), \tag{1}$$

for $w \in B_2, T^{-1}w(0) = u_1 \in T^{-1}M$. In the case of quantization, the formula has to be modified to

$$\frac{\partial w}{\partial t} = -\chi_{[-Q,Q]}(w - Tu_1) T grad\Phi(T^{-1}w), \tag{2}$$

with the initial condition as above. The algorithms are as follows.

1. Choose an energy functional, say Φ.

2. Pick T to be the wavelet (wavelet packet) transform according to what basis has been used to threshold or quantize the image.

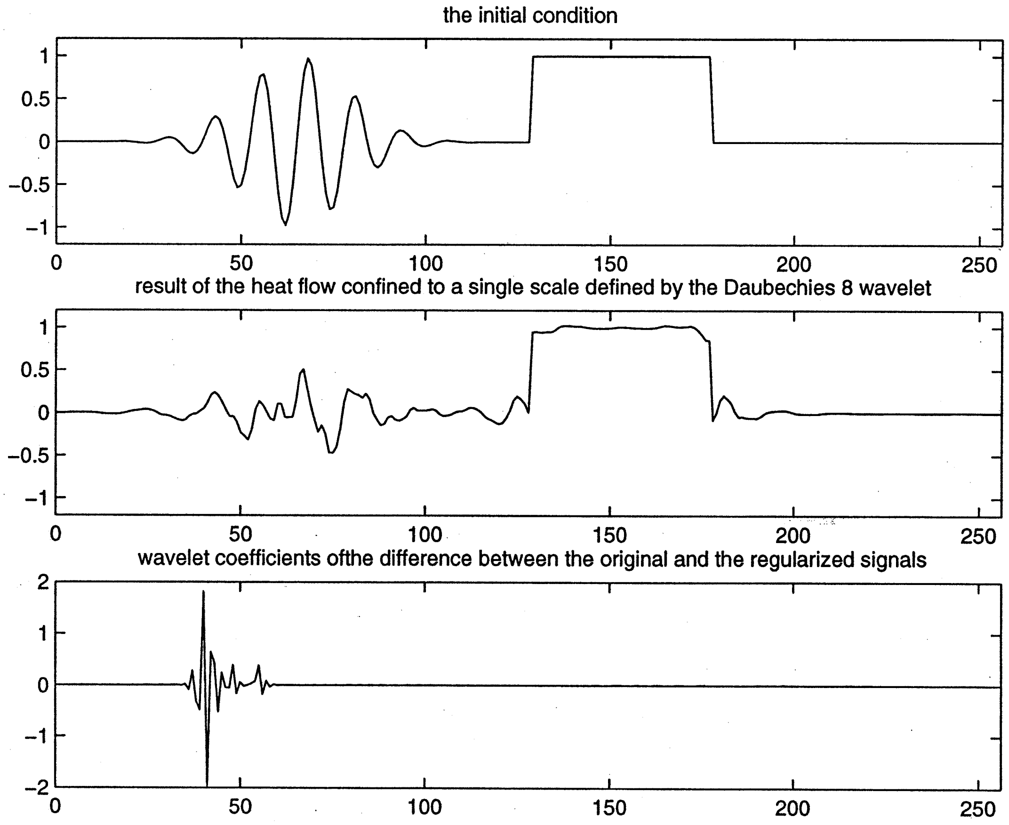
3. If repairing a thresholded image set $M = span\{\psi_i :< u_1, \psi_i >= 0\}$, and let P_M denote the orthogonal projection into M.

4. Chose Q to be equal either to half the quanta or the threshold.

5. Perform S steps of a discrete version of the evolution equation (1) in the case of thresholding and (2) in the case of quantization. S is to be chosen by experiment.

Remarks

- The algorithms presented above have one clear disadvantage. Since the gradient of the functional Φ does not have anything in common with the signal, the method of the steepest descent introduces its own artifacts. One-dimensional signals are usually smoothed, but on the other hand application of the algorithm can amplify the Gibbs phenomenon. For images we will observe edge and corner smearing, but also a propagation-of-textures kind of effect. All this depends heavily on the choice of a particular functional Φ, and the time of evolution. Below is a typical result of evolution confined to a fixed scale.



the initial condition

result of the heat flow confined to a single scale defined by the Daubechies 8 wavelet

wavelet coefficients ofthe difference between the original and the regularized signals

- In some cases improvement in quality of the enhanced image can be obtained by replacing the wavelet transform by the undecimated (redundant) wavelet transform.

- Furthermore, the method extends to other redundant descriptions as well. For instance, interesting effects have been obtained by simultaneous use of several bases. In this case equation (1) has to be replaced by a system of two (several) equations of the form

$$\begin{aligned} \tfrac{\partial w_1}{\partial t} &= -\chi_{[-Q_1,Q_1]}(w_1) P_{M_1} T_1 grad\Phi(T_2^{-1} w_2) \\ \tfrac{\partial w_2}{\partial t} &= -\chi_{[-Q_2,Q_2]}(w_2) P_{M_2} T_2 grad\Phi(T_1^{-1} w_1), \end{aligned} \qquad (3)$$

  with $w_i \in B_2^i$, $T_i : B_1 B_2^i$, and $P_{M_i} : B_2^i \to M_i$ for $i = 1, 2$ are as above, and the initial conditions $T_i^{-1} w_i(0) = u_1 \in T_1^{-1} M \cap T_2^{-1} M$ are consistent.

- Let us note that the effects of intertwining an elliptic differential operator with integral transforms as in $T grad\Phi T^{-1}$ are far from trivial, let alone the nonlinear multiplication by an expression of the type $\chi_{[-Q,Q]}(w)$. Indeed, it is a very instructive exercise to show that in the one-dimensional case with the discrete linear second derivative $(u'')_i = u_{i+1} - 2u_i + u_{i-1}$ and T given by the Haar wavelet transform at one level. In fact, it is easily seen that for $w = w_H + w_G$, where w_H (w_G) are the low-pass (respectively high-pass) components of w, we have

$$T((T^{-1}w)'') = \begin{pmatrix} \phi_{HH} & \phi_{HG} \\ \phi_{GH} & \phi_{GG} \end{pmatrix} * \begin{pmatrix} w_H \\ w_G \end{pmatrix},$$

  where $*$ denotes the convolution, and

$$\phi_{HH} = (-1, 2, -1),$$

$$\phi_{HG} = (1, 0, -1),$$

$$\phi_{GH} = (-1, 0, 1),$$

$$\phi_{GG} = (-1, -6, -1).$$

This means that although the effect of $T((T^{-1}w)'')$ will be just an application of the second derivative to the low-pass component, it acts on the high-pass component by convolution with a smoothing kernel. In addition, there is some interaction between the channels. Similar behavior is observed with other wavelets as well. Of course, we should be aware of the fact that when $grad\Phi$ is nonlinear, the interaction between channels is essentially beyond control.

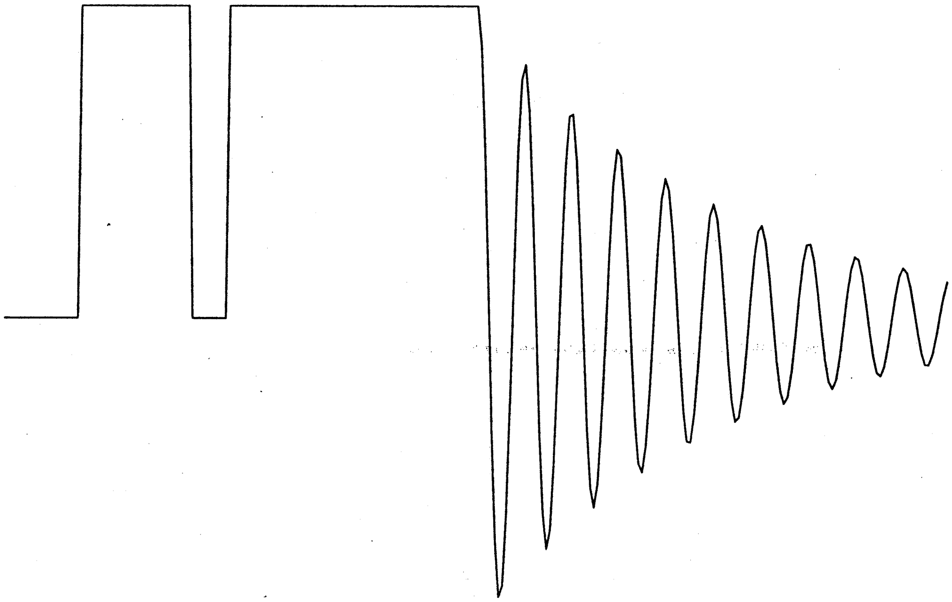
- On the other hand the flow given by equation (1) can be identified as the projection of an ordinary heat-flow to a (moving) subset. Indeed, Let us denote $u = T^{-1}w$, and let us apply T^{-1} to both the sides of the equation (1). If we denote $R_u = T^{-1} \circ \chi_{[-Q,Q]}(Tu)P_M \circ T$, we obtain
$$\frac{\partial u}{\partial t} = -R_u grad\Phi(u), \tag{4}$$
which is equivalent to the equation (1). We now check that R really is a projection. Indeed, since $\chi^2_{[-Q,Q]} = \chi_{[-Q,Q]}$ and $P^2_M = P_M$, we obtain
$$\begin{aligned} R_u \circ R_u &= T^{-1} \circ \chi_{[-Q,Q]}(Tu)P_M \circ T \circ T^{-1} \circ \chi_{[-Q,Q]}(Tu)P_M \circ T \\ &= T^{-1} \circ \chi_{[-Q,Q]}(Tu)P_M \circ T = R_u. \end{aligned}$$
In addition, since P_M, multiplication by $\chi_{[-Q,Q](Tu)}$, and for orthogonal wavelets the projection P_M are all self-adjoint, so is the projection R_u, i.e. $R^\star_u = R_u$ for all u. Thus the flow (1), which is equivalent to the flow (4) represents projection of the heat flow given by $grad\Phi$ to a constraint subset in B_1. We emphasize however that the constraint, as well as the projection R_u, depends on u, so generically they will change at every moment during the evolution. This sheds some light on how tremendously complex these processes are.

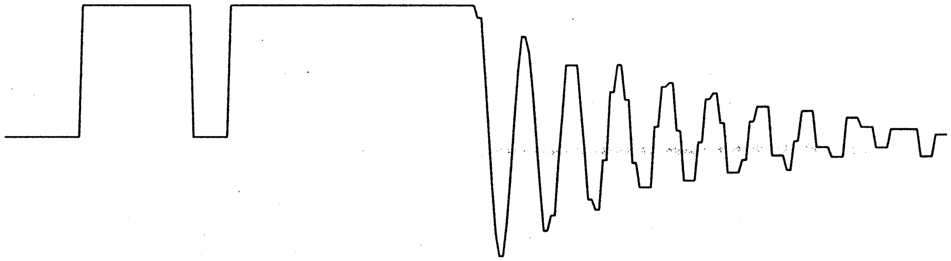
- In essence, a method roughly of this type (with $\Phi(u) = \int |\nabla u|^2$) was investigated in [2]. There, the algorithm is additionally required to proceed scale by scale and $grad\Phi$ is being evaluated not only in the physical space but also in the phase space.

Below we present a few examples of results obtained with the algorithm above. We will refer to the heat flow using ordinary second derivative as simply "heat flow". By the $L1$ heat flow we understand a regularized nonlinear parabolic flow determined by $\Phi(u) = \int |\nabla u|$, or more precisely
$$\frac{\partial u}{\partial t} = \left(\frac{u'}{(u'^2 + \varepsilon^2)^{\frac{1}{2}}} \right)',$$
where ε is a small number. (To assure regular behaviour, one needs to choose the time step unit less than at $\frac{\varepsilon}{10}$.) Of course the results depend on

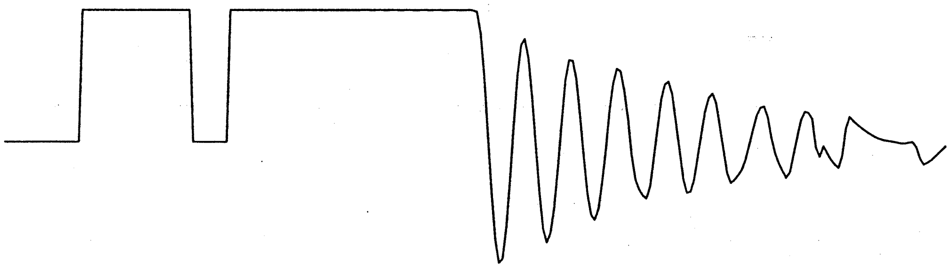
the stopping time, although, at least in principle, one expects convergence. The original signal used throughout this article is displayed below.

The original signal used throughout



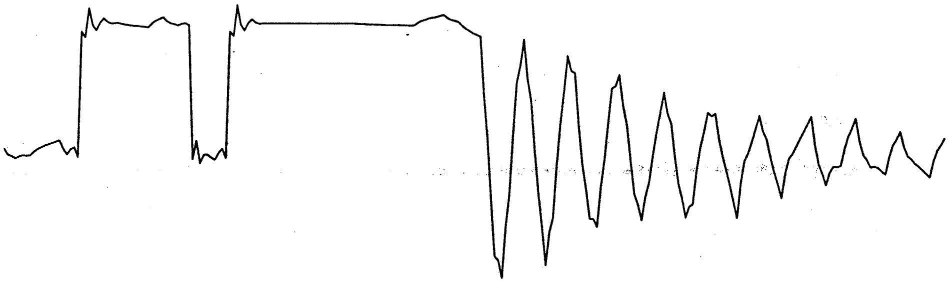


The thresholded signal, Haar basis,threshold = .2

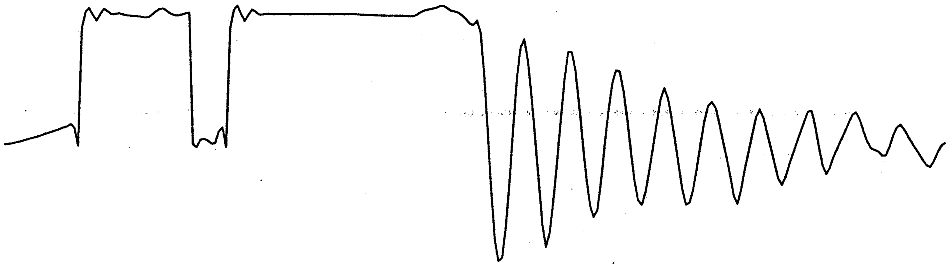


Signal restored with the heat flow

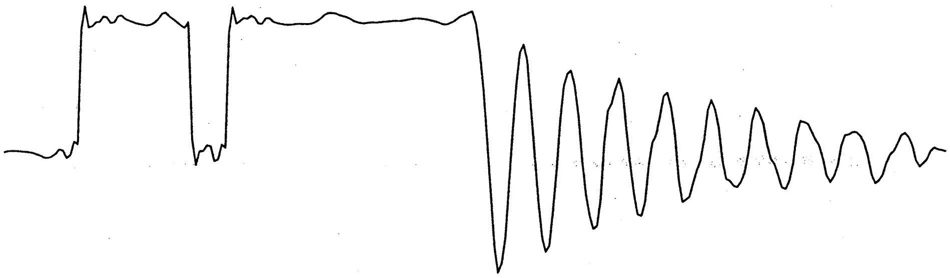
The thresholded signal, Daubechies4, threshold =.2



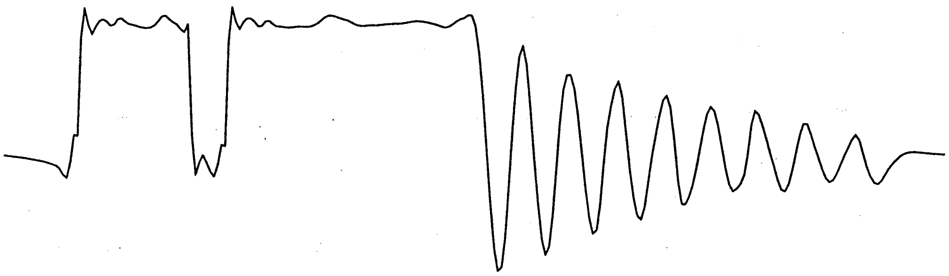
Signal restored, with the heat–flow



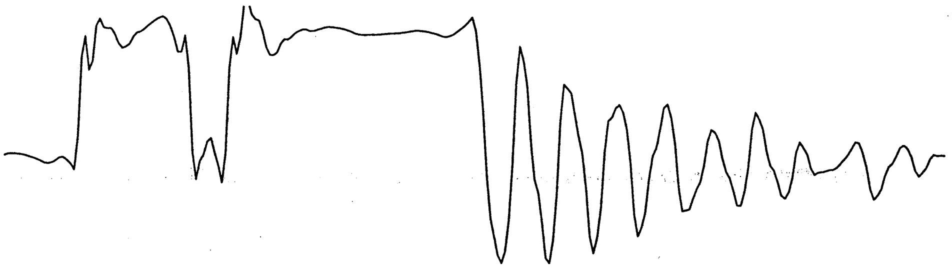
The thresholded signal, Daubechies8, threshold =.2



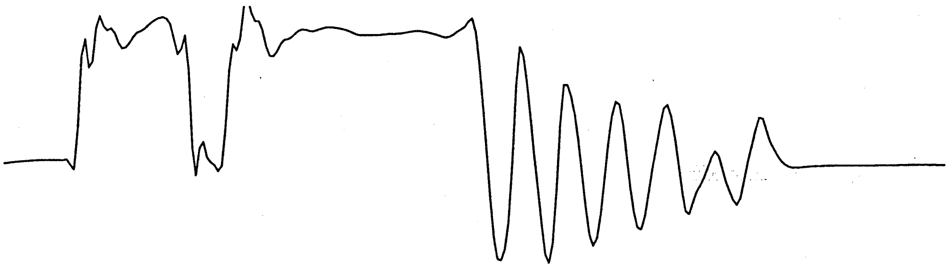
Signal restored, with the heat–flow



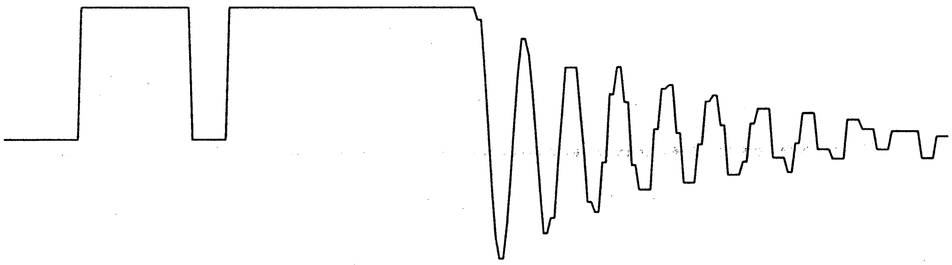
The thresholded signal, Daubechies8, threshold =.4



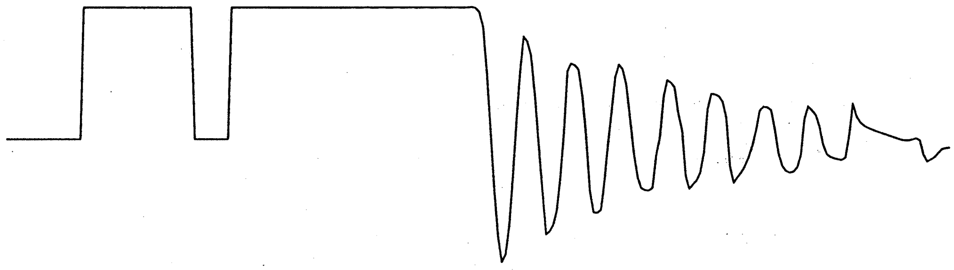
Signal restored, with the heat–flow



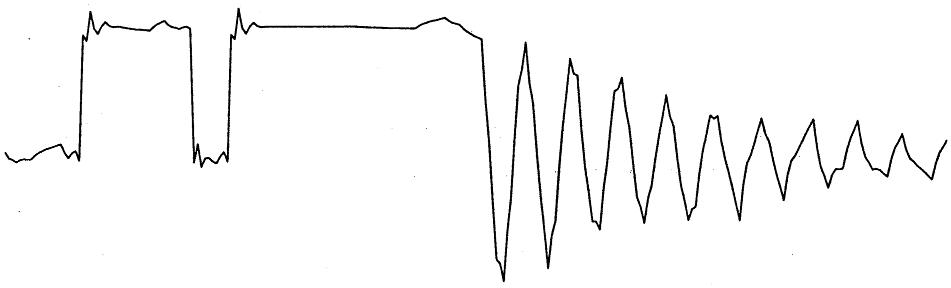
The thresholded signal, Haar basis, threshold =.2



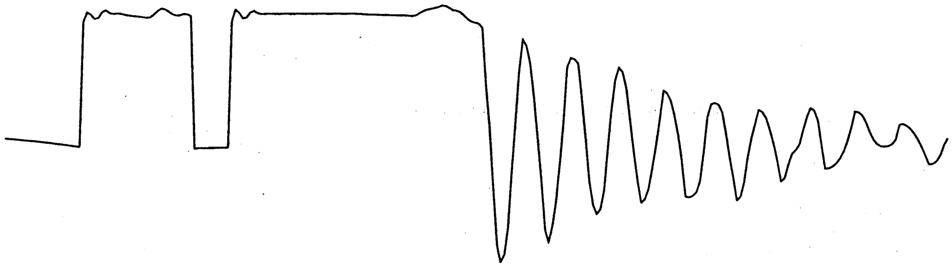
Signal restored, with the L1 heat–flow



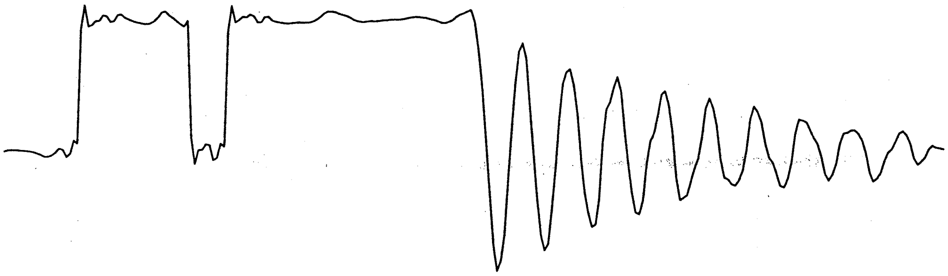
The thresholded signal, Daubechies4, threshold =.2



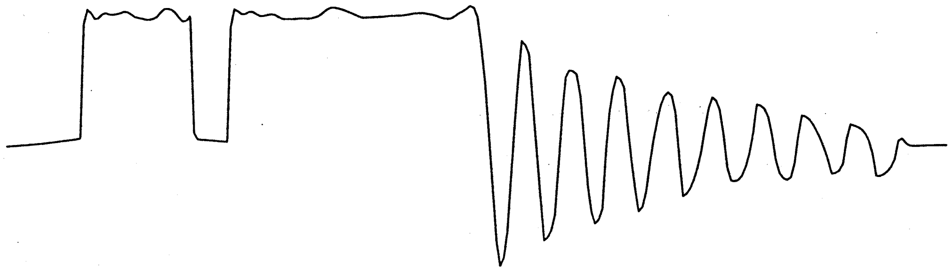
Signal restored, with the L1 heat–flow



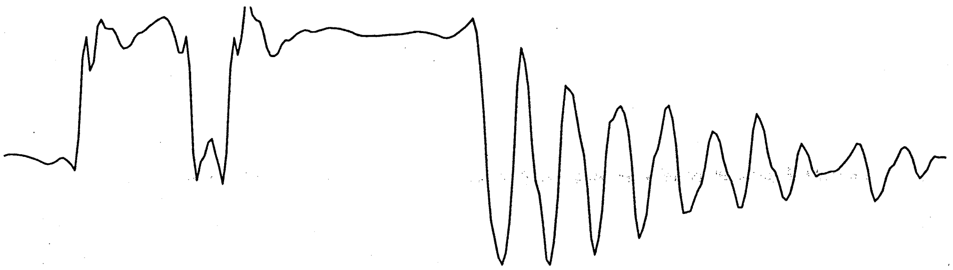
The thresholded signal, Daubechies8, threshold =.2



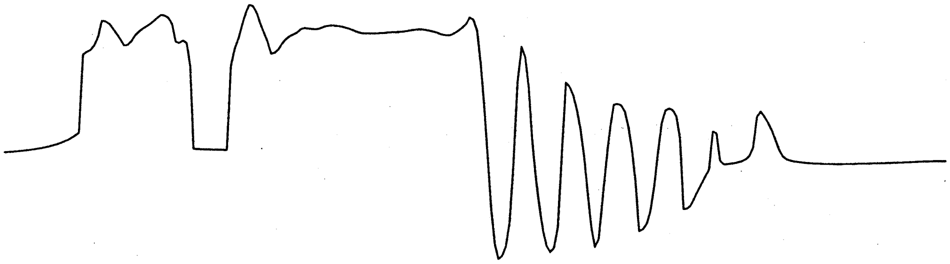
Signal restored, with the L1 heat–flow



The thresholded signal, Daubechies8, threshold =.4



Signal restored, with the L1 heat–flow



Methods of the Second Kind

The second method attempts to take advantage of the fact that artifacts introduced by thresholding often resemble, in shape, wavelets themselves. More strictly, given initial signal u_0, both thresholding and quantization will add a bunch of wavelets to it, and the resulting signal has the form $u_1 = u_0 + \sum a_i\psi_i$. Our task then is defined as detection of the ψ_i's. Moreover, we know that a_i must satisfy constraints of the form $|a_i| < Q$, where Q either does not depend on i , or we know how it depends on i. (The first possibility holds in the case of quantization and then $2Q$ is the quanta. The second possibility holds in the case of thresholding and then Q equals the threshold for those i for which $< u_1, \psi_i >= 0$, and $Q = 0$ otherwise.) We use many distinct ways of detecting correlation of shapes, but in all the cases, we proceed as follows.

1. Choose a measure of correlation, say C. $C(f, g)$ is a (bilinear or not) functional, which is asumed to measure how similar are (the shapes of) the two signals f and g.

2. Pick ε, consider $u_1 + \varepsilon\psi_i$ and $u_1 - \varepsilon\psi_i$. Consider all the ψ_i in the case of quantization and only those for which $< u_1, \psi_i >= 0$ in the case of thresholding. Choose a *threshold of correlation cor*.

3. If $C(u_1, \psi_i) > cor$ update $u_1 = u_1 + \varepsilon\psi_i$, else if $C(u_1, -\psi_i) > cor$ update $u_1 = u_1 - \varepsilon\psi_i$, else do nothing at this ψ_i, consider the next ψ_i. Try all the wavelets ψ_i (or those which have been set to zero by thresholding) to reconstruct one ε layer.

4. Repeat the previous step Q/ε times, i.e. go through the total of Q/ε layers of reconstruction.

In the experiments whose results are presented below, we have used the following measures of correlation.

1. $C1$ is given by
$$C1(u_1, \psi_i) = -\frac{\int < u_1', \psi_i' >}{\int |\psi_i'|^2},$$
where $(.)'$ denotes the derivative.

2. C2 is given by
$$C2(u_1, \psi_i) = \int \kappa_{u_1}\kappa_{\psi_i},$$

where κ_f denotes the geodesic curvature of f, i.e.

$$\kappa_f = \frac{f''}{(1+f'^2)^{\frac{3}{2}}}.$$

Of course, depending on the specific application one can design functionals other than those presented above.

Remarks

- The computational time required by the algorithm is $C\frac{M}{\varepsilon}N$. Here N is the number of ψ_i's we need to consider, which is equal to the signal length (or "area" for 2D) for quantization and smaller for thresholding. The constant C is basically the computational time required to perform step *3*, which depends only on the chosen Φ.

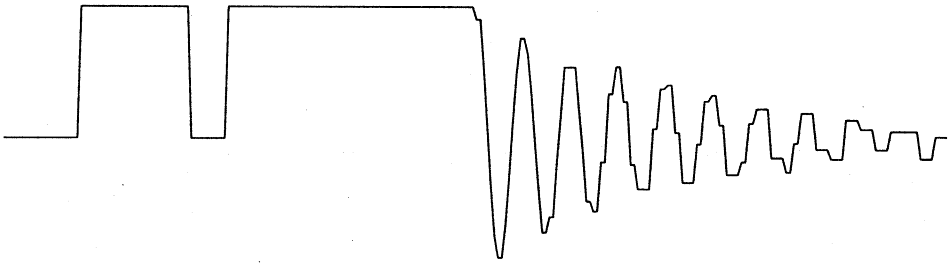
- The signal one obtains in this way is quantized with the quanta equal to ε. A priori, the smaller ε, the better the results, but in practice exceeding exactness gives negligible improvement in quality of the signal.

- It has to be noted that the algorithm has built in preference for $u_1+\varepsilon\psi$ over $u_1 - \varepsilon\psi$ or vice versa, depending on which is being tested first. This affects the evolution of u_1 only at points where we are at a local maximum or a saddlepoint of the functional Φ. However, experiment shows that the effects of this bias are completely negligible for signals one encounters in practice.

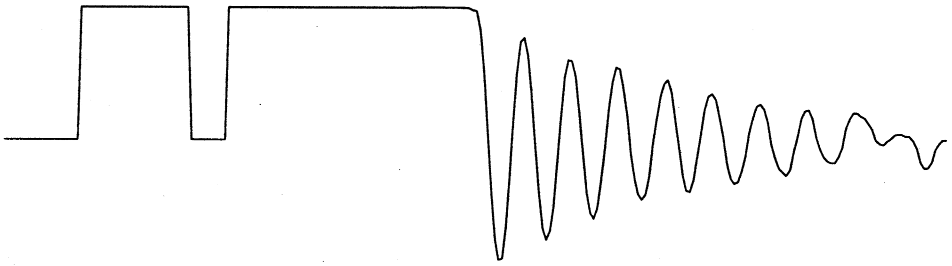
- It is an attractive and convenient feature of this algorithm that it does not require any regularity of the functional Φ, since we do not need to know its gradient. The shortcoming of this algorithm is that if one attempts to use it actually to minimize the functional Φ, one has to successively refine the ε quantization. This poses nontrivial (even in finite dimensions) questions about convergence to the minimizer and, needless to say, the answer will depend on the properties of Φ. We emphasize however that it is not our task in this paper to minimize functionals, but to detect and erase artifacts of wavelet compression. For our purposes here, the issues just raised are irrelevant. What is more, the self-imposed limitations of this method are designed to play to our advantage.

- For our purposes, it always pays to choose the functional Φ that will in some sense emphasize the role of the building blocks ψ_i. Below, we present the results obtained by application of this algorithm with two different energy functionals to one-dimensional signals.

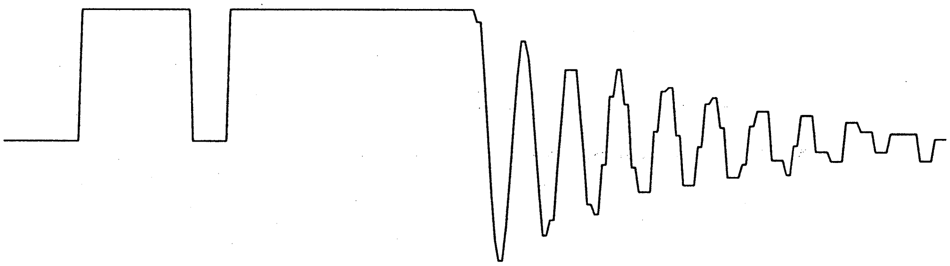
The thresholded signal, Haar basis, threshold =.2



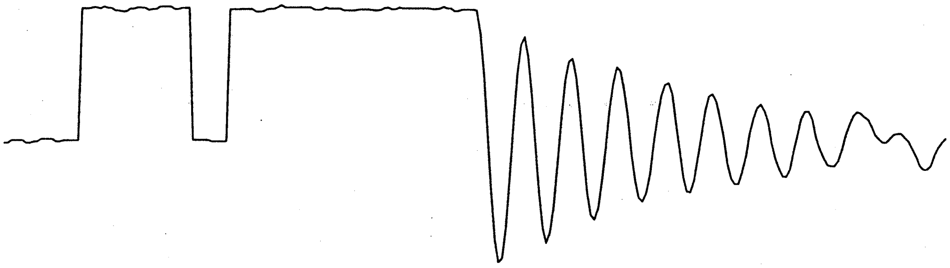
Signal restored, C1, cor =.1



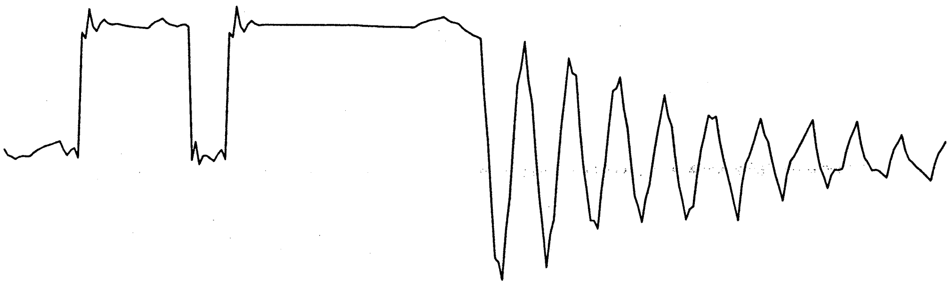
The thresholded signal, Haar basis, threshold =.2



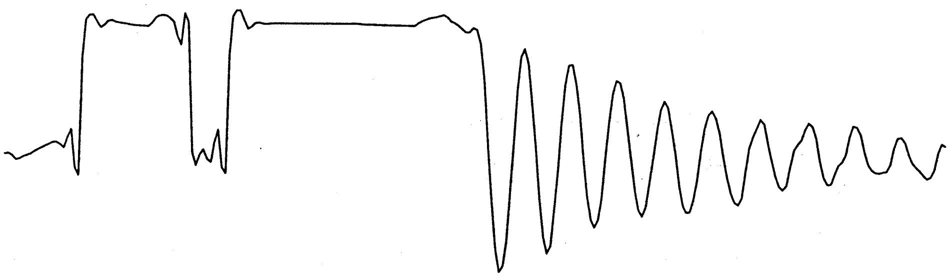
Signal restored, C1, cor =0



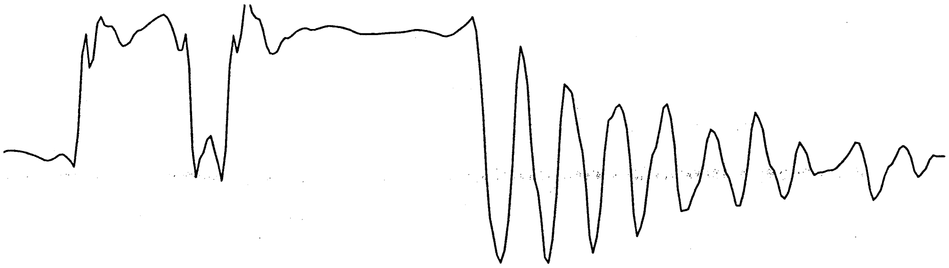
The thresholded signal, Daubechies4, threshold =.2



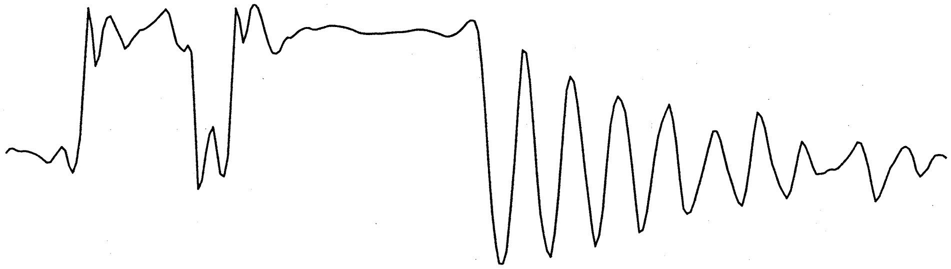
Signal restored, C2, cor =.01



The thresholded signal, Daubechies8, threshold =.4



Signal restored, C2, cor =.02



It has to be mentioned that in the case of, say, the functional $C1$, it is possible to use $C1(u_1, \psi_i)$ itself as a good candidate for the amplitude at ψ_i, and avoid the ε steps. We will explore this possibility in the next section. Let us now remark that in fact, even if we deal with an expression that is not bilinear, as for example $C2$, it is still possible to produce a good candidate for the amplitude using the "nonlinear projection", e.g. we can use the minimizer of $\inf_\lambda \int |\kappa_{u_1+\lambda\psi_i}|^p$. for some $p \geq 1$. It is, however, more computationally expensive to minimize this nonlinear expression than to perform a large number of ε steps. Moreover, the experiment shows no improvement is achieved with this method.

3 A rapid algorithm for enhancement of wavelet compressed images

Suppose we are given an $N \times N$ matrix u_1 representing an image. Although the algorithm presented above computes in time proportional to N^2, i.e. it is technically a fast $O(n)$-algorithm, the constant is in practice too large to make it useful for image enhancement in real time. We show below how to devise a really rapid version of this algorithm that proves to be especially effective for the Haar wavelet (making it a much more practical wavelet than you may have thought).

Here is how we go about it. First, if we want to construct a rapid algorithm, we have to get rid of the ε and try to do whatever the calculations we need simultaneously. This forces us to fix the amplitude of ψ_i at the optimal level once and for all. It seems natural to pick a_i such that

$$\Phi(u_1 + a_i\psi_i) = \inf_\lambda \Phi(u_1 + \lambda\psi_i).$$

This can be identified as performing one step towards the minimization of $\Phi(u)$ by the method of relaxation (cf. [3]). Secondly, in order to find the number a_i defined above numerically, we would usually have to spend a lot of time doing it, so it must be possible to obtain an explicit formula for a_i. Thirdly, even if we succeed in obtaining such a formula, it still does not guarantee that its numerical evaluation will be fast enough. If we want it to be fast, the formula for a_i can only depend on a few coefficients of u_1 and ψ_i. All these conditions are met if we pick $\Phi(u) = \int |\nabla u|^2$. Indeed, then

$$\Phi(u_1 + \lambda\psi_i) = \int |\nabla u_1|^2 + 2\lambda \int < \nabla u_1, \nabla\psi_i > + \lambda^2 \int |\nabla\psi_i|^2,$$

and the minimum is assumed for

$$\lambda = a_i = -\frac{\int < \nabla u_1, \nabla \psi_i >}{\int |\nabla \psi_i|^2}.$$

The constants $\int |\nabla \psi_i|^2$ depend only on wavelets and can be precomputed. Moreover, the constants depend only on "bands" in the multiresolution table, which makes it possible to perform even this precomputation rather fast. More important, the numerators in the formula above are easy to compute if we integrate by parts to obtain

$$a_i = \frac{\int < \Delta u_1, \psi_i >}{\int |\nabla \psi_i|^2}. \tag{5}$$

To justify integration by parts we make two remarks: First of all, we consider the pictures as functions on a two-dimensional torus, which is a closed manifold and therefore no boundary terms are present. We must add that we a use periodic wavelet transform in the applications, so that ψ_i's can be thought of as functions on the torus, too. Second of all, this integration by parts can be written down in the finitistic version with the surface integral replaced by a double sum. This makes it retain its meaning in the digitized version. In summary, let u_1 be the image before processing, and let u_2 denote the image after processing. Let WT and IWT denote respectively the wavelet transform (inverse wavelet transform) in a fixed wavelet (wavelet-packet) basis. The algorithm we propose consists of the following:

1. Find the wavelet coefficients $c_i = WT(u_1) =< u_1, \psi_i >$.

2. Evaluate the finite difference Laplacian Δu_1.

3. Find the wavelet coefficients of the Laplacian $a_i'' = WT(\Delta u_1) =< \Delta u_1, \psi_i >$.

4. Rescale the coefficients a_i'' according to the formula (5), dividing them by pre-computed constants $\int |\nabla \psi_i|^2$. Obtain a_i'.

5. Reset to zero those coefficients a_i' that do not satisfy the constraint. It means that if we are repairing a quantized image, we will put

   $$a_i = a_i' I_{\{|a_i'|<Q\}}.$$

   If on the other hand we are repairing a thresholded image, we will put

   $$a_i = a_i' I_{\{|a_i'|<Q\}} I_{\{c_i=0\}},$$

   where I_X is the characteristic function of the set X.

6. Define the restored image as

$$u_2 = IWT(c_i + a_i).$$

Remarks

- All the transforms and other computations are periodic.

- The computational cost of the algorithm is essentially the cost of application of a filter corresponding to the Laplacian (see below) and twice the wavelet (or waveletpacket) and the inverse wavelet (wavelet packet) transform — once for the input signal u_1 and once for its Laplacian $\triangle u_1$. In addition every coefficient of the wavelet (wavelet packet) transform of the Laplacian $\triangle u_1$ has to be rescaled exactly once by a precomputed constant that depends only on the wavelets used and does not depend on the signal itself.

- In experiments we use with good results the simplest possible finite-difference Laplacian, i.e. if U is a matrix then with the obvious notation

  $$(\triangle U)_{i,j} = U_{i+1,j} + U_{i,j+1} + U_{i-1,j} + U_{i-1,j-1} - 4U_{i,j}$$

  with the obvious periodic extension at the boundary.

- Since the numbers we use for rescaling $\int |\nabla \psi_i|^2$ depend on the shape of ψ_i and not their position, they are constant within a given scale (or scale and band if wavelet-packets are used).

- If we were to look at this algorithm as a version of discrete heat flow restricted to certain scales and possibly bands, we would have to point out two distinctions. Firstly, there is only one step in (discrete) time. The length of this step depends on the scale (band) and is given exactly by the number $(\int |\nabla \psi_i|^2)^{-1}$. We believe there is no way of justifying this selection other than it has been done above by means of "relaxation" in the direction of wavelets.

- Since multiplication of distributions is not well defined, the formula $\int |\nabla \psi_i|^2$ makes no sense for discontinuous Haar wavelets seen as functions of real variables. It can, however, be evaluated in the discrete version of wavelets, where ∇ denotes the left or right directional finite differences. (To avoid mistakes and resulting discrepancies between definitions of the Laplacian and the gradient, it is always safest to integrate by parts and compute these numbers in the form $\int < \Delta\psi_i, \psi_i >$.) The experiment shows that this works very well. In fact the existence of singularities may account for the fact that this method is most efficient for detection of artifacts left by the Haar wavelet.

Had we kept adding the wavelets one after another, we would have been sure that $\Phi(u_2) < \Phi(u_1)$. However, we have only used the functional Φ to compute the consecutive coefficients, and then we have added all the wavelets simultaneously. There is no a priori abstract reason for the value of the functional to go down. Below we prove that the L^2 variation will in fact decrease. For clarity we will start with the one-dimensional version of the algorithm. We will then show how to restate the theorem and refurnish its proof to obtain the two-dimensional version of the algorithm. We emphasize that the signals are considered to be defined on a circle (or a torus) and the analysis is periodic. In what follows we will be concerned with wavelets in a fixed band. More precisely, we consider a family of wavelets $\{\psi_i\}$ which are all shifts of a certain function $\psi \in C^2$ (the regularity condition can be weakened), i.e.

$$\psi_i(x) = \psi(x - i) \tag{6}$$

for $i = 1, 2, \ldots, N$ with the convention that the shift is circular, i.e.

$$\psi_i(x) = \psi_{N-i}(x). \tag{7}$$

Let us introduce the following notation. Let

$$M(k) = \frac{\int < \nabla\psi, \nabla\psi_k >}{\int |\nabla\psi|^2}. \tag{8}$$

Below we will use the discrete Fourier transform and its basic properties. Here we just remind the definition. Let $x = (x_0, x_1, \ldots, x_{N-1})$ be a (complex) vector. By its digital Fourier transform (DFT) we understand the vector $X = DFT(x)$ given by:

$$X_k = \sum_{l=0}^{N-1} x_l \exp(-\frac{2\pi\sqrt{-1}lk}{N}).$$

It is an (easy to verify) experimental fact that

Experimental Fact 1 *For all the (digitized periodic) compactly supported orthogonal wavelets ψ, the vector $M = (M(0), M(1), \ldots M(N-1))$ satisfies*

$$\sum_{k=0}^{N-1} |M(k)| < 2$$

and a fortiori

$$DFT(M) < 2.$$

Its continuous variable counterpart would be the inequality

$$\hat{M}(\xi) = (\int |\nabla\psi|^2)^{-1} |\xi \cdot \hat{\psi}(\xi)|^2 < 2.$$

We will show that this fact is equivalent to the property that the output signal has lower L^2 variation than the input of the algorithm above (cf. Theorem 1 below). In what follows, we will need the following lemma.

Lemma 1 *The quadratic form*

$$\Phi(x_1, x_2, \ldots, x_N) = -\sum_{i=1}^{N} x_i^2 + 2 \sum_{1 \leq i < j \leq N} x_i x_j M(j-i),$$

where $M(j-i)$ is as defined above, is negative definite, i.e.

$$\Phi(x_1, x_2, \ldots, x_N) \leq 0$$

with equality if and only if $x_1 = x_2 = \ldots = x_N = 0$.

Proof. Let us define a matrix A given by:

$$A = \begin{pmatrix} -1 & M(1) & M(2) & M(3) & \ldots & M(N-1) \\ M(1) & -1 & M(1) & M(2) & \ldots & M(N-2) \\ \vdots & \vdots & \vdots & \vdots & \ldots & \vdots \\ M(N-1) & M(N-2) & M(N-3) & M(N-4) & \ldots & -1 \end{pmatrix}$$

It suffices to show that all the eigenvalues of A are negative. We first note that A is circulant. Indeed, since ψ_i's are functions on a circle, we have

$$M(i) = M(N-i)$$

for all $i = 1, 2, \ldots, N$. Next, we use the well known fact that the eigenvalues of a circulant matrix are given by the discrete Fourier transform of one of its rows (cf. [4]). More precisely, if we denote the eigenvalues of A by λ_k for $k = 1, 2, \ldots, N$, then

$$\lambda_k = DFT(A_{1,j})$$

for $k = 1, 2, \ldots, N$. Next we will use the fact that the entries $A_{1,j}$ are given by

$$A_{1,j} = M(j) - 2\delta_0,$$

for $j = 0, 1, \ldots, N - 1$ where δ_0 denotes the Dirac delta supported on the first component. We now use the fact that $DFT(\delta_0) = 1$. Together with Experimental Fact 1 mentioned above before the Lemma 1, this completes the proof.

Now we can prove that the following theorem holds.

Theorem 1 *In one dimension, the energies of the output u_2 and the input u_1 of the algorithm above in both the case of thresholding and quantization (cf. point. 5 of the algorithm) satisfy the inequality*

$$\int |\nabla u_2|^2 \leq \int |\nabla u_1|^2,$$

and they are equal if and only if $u_2 = u_1$.

Proof. First we perform an analysis with the assumption that all the wavelets are taken from one scale (and band), so that the situation is as in the equation (6) above. Suppose further that all the coefficients a_i fall below the threshold (or within the quantization limit), i.e. the output u_2 is given by

$$u_2 = u_1 + \sum_{i=1}^{N} \frac{\int < \Delta u_1, \psi_i >}{\int |\nabla \psi|^2} \psi_i = u_1 - \sum_{i=1}^{N} \frac{\int < \nabla u_1, \nabla \psi_i >}{\int |\nabla \psi|^2} \psi_i.$$

We apply ∇ to both sides of this equation and obtain as a result

$$
\begin{aligned}
&\int |\nabla u_2|^2 = \int |\nabla u_1 + \sum_{i=1}^{N} a_i \nabla\psi_i|^2 = \\
&\int |\nabla u_1|^2 - 2\sum_i \frac{\int <\nabla u_1, \nabla\psi_i>^2}{\int |\nabla\psi|^2} \\
&+\sum_{i,j} \frac{\int <\nabla u_1, \nabla\psi_i> \int <\nabla u_1, \nabla\psi_j> \int <\nabla\psi_i, \nabla\psi_j>}{\int |\nabla\psi|^4} = \\
&\int |\nabla u_1|^2 - \sum_i \frac{\int <\nabla u_1, \nabla\psi_i>^2}{\int |\nabla\psi|^2} \\
&+2\sum_{i<j} \frac{\int <\nabla u_1, \nabla\psi_i> \int <\nabla u_1, \nabla\psi_j> \int <\nabla\psi_i, \nabla\psi_j>}{\int |\nabla\psi|^4} = \\
&\int |\nabla u_1|^2 + \frac{1}{\int |\nabla\psi|^2}\Big(-\sum_{i=1}^{N} x_i^2 + 2\sum_{1\le i<j\le N} x_i x_j M(j-i)\Big),
\end{aligned}
\tag{9}
$$

where $x_i = \int < \nabla u_1, \nabla\psi_i >$. Thus the claim now follows from the Lemma 1. We will now examine what happens if some of the coefficients a_i are obtained by a nontrivial application of thresholding to the original correlation coefficients, i.e. let

$$a_i' = \frac{\int < \nabla u_1, \nabla\psi_i >}{\int |\nabla\psi|^2},$$

and let

$$a_i = a_i' I_{\{|a_i|<Q\}} I_{\{c_i=0\}},$$

in the case of thresholding and

$$a_i = a_i' I_{\{|a_i|<Q\}},$$

in the case of quantization, both for a given threshold Q (respectively quanta $2Q$). In both cases we denote $y_i = \int < \nabla u_1, \nabla\psi_i >$, and $x_i = \int |\nabla\psi|^2 a_i$. Below we are going to use the property that $x_i y_i = x_i^2$. A calculation as above shows that now

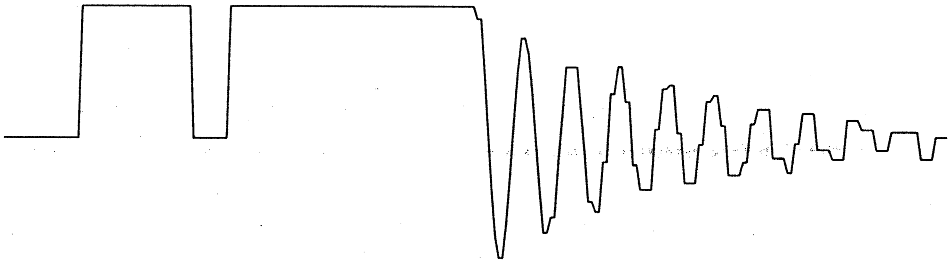
$$
\begin{aligned}
&\int |\nabla u_2|^2 - \int |\nabla u_1|^2 = \\
&\frac{1}{\int |\nabla\psi|^2}\Big(-2\sum_{i=1}^{N} x_i y_i + \sum_{i,j} x_i x_j M(j-i)\Big) = \\
&\frac{1}{\int |\nabla\psi|^2}\Big(-2\sum_{i=1}^{N} x_i^2 + \sum_{i,j} x_i x_j M(j-i)\Big) = \\
&\frac{1}{\int |\nabla\psi|^2}\Big(-\sum_{i=1}^{N} x_i^2 + 2\sum_{1\le i<j\le N} x_i x_j M(j-i)\Big) \le 0,
\end{aligned}
\tag{10}
$$

where the last inequality is justified as before, and again the equality holds only if u_2 and u_1 are identical. Finally, let us remark that if we perform the algorithm with wavelets at many different scales (bands), the analyses can still be carried out in a scale-by-scale (or band-by-band) fashion. This time one decomposes the difference $\int |\nabla u_2|^2 - \int |\nabla u_1|^2$ into a sum of several quadratic forms and then estimates each one separately exactly as above. The estimates hold as long as Experimental Fact 1 is satisfied for wavelets (wavelet-packets) at those scales and bands. This concludes the proof.

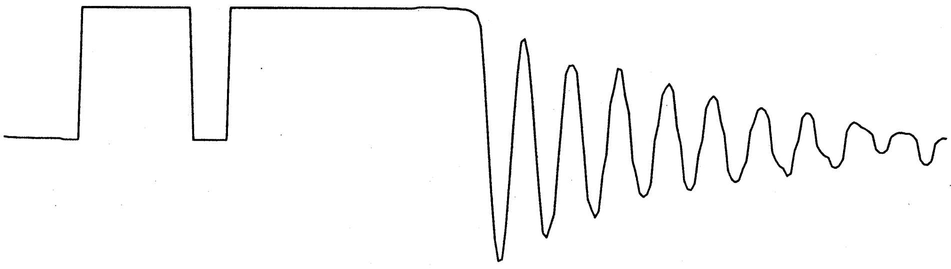
Remarks

- We emphasize that the theorems above do in fact depend on the properties of the wavelets used throughout. Indeed, consider the following problem: Suppose
$$u_2 = u_1 + c \triangle u_1 \tag{11}$$
for a certain constant c. Can c be chosen independently of u_1 in such a way that we would always have
$$\int |\nabla u_2|^2 \leq \int |\nabla u_1|^2 ? \tag{12}$$
The answer is NO, which is easily seen if one plugs into equation (11) the eigenfunctions of the Laplacian corresponding to higher and higher eigenvalues. Next, it suffices to evaluate the integrals $\int |\nabla u_i|^2$ to see that the constant c will have to decrease to zero as the eigenvalues increase in order to guarantee that the above inequality holds. (The reader unfamiliar with the general notions can try $u_1 = \sin(2\pi kx)$ as eigenfunctions on the unit circle to obtain that (12) is now equivalent to $c < (2\pi k)^{-1}$.)

We now present results of application of the fast algorithm above in a few cases of thresholded one-dimensional signals. It is clear that the improvement is more spectacular when the artifacts are stronger, i.e.for less regular wavelets. The same rule governs the case of images (cf. below).

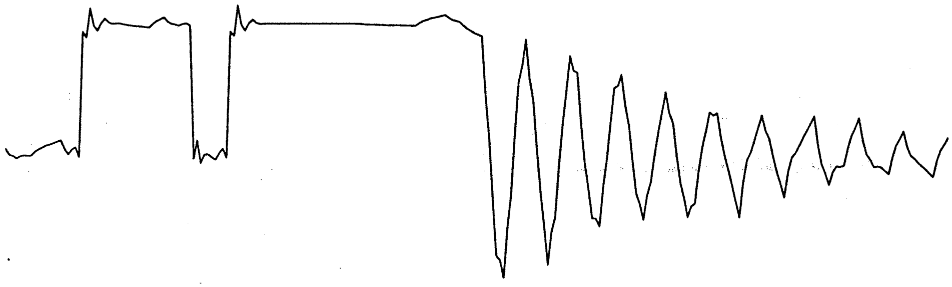


The thresholded signal, Haar basis, threshold =.2

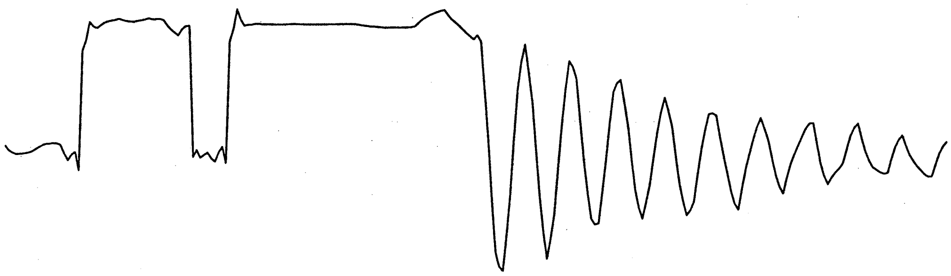


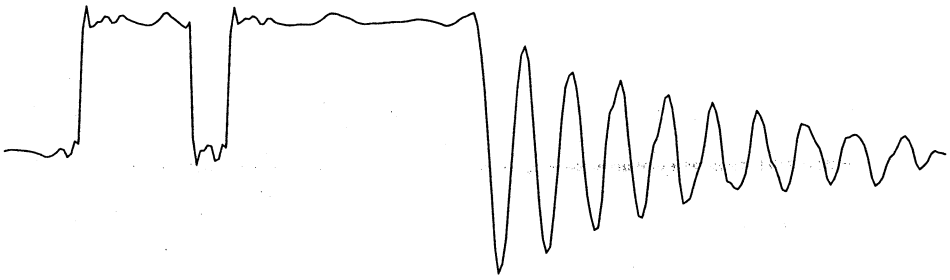
Signal restored

The thresholded signal, Daubechies4, threshold =.2

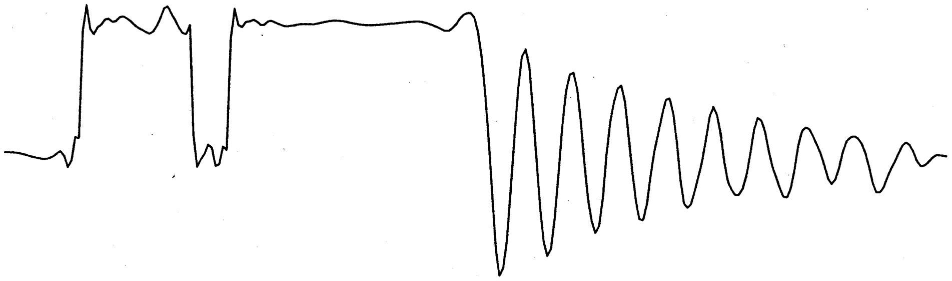


Signal restored





The thresholded signal, Daubechies8, threshold =.2



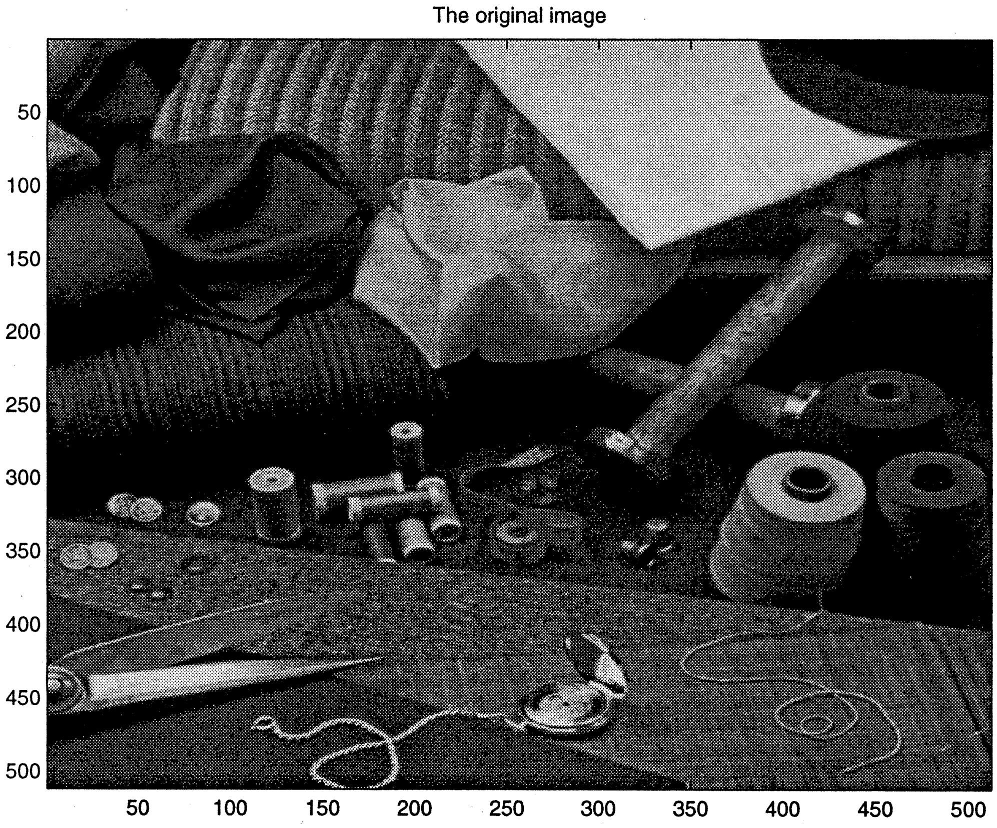
Signal restored

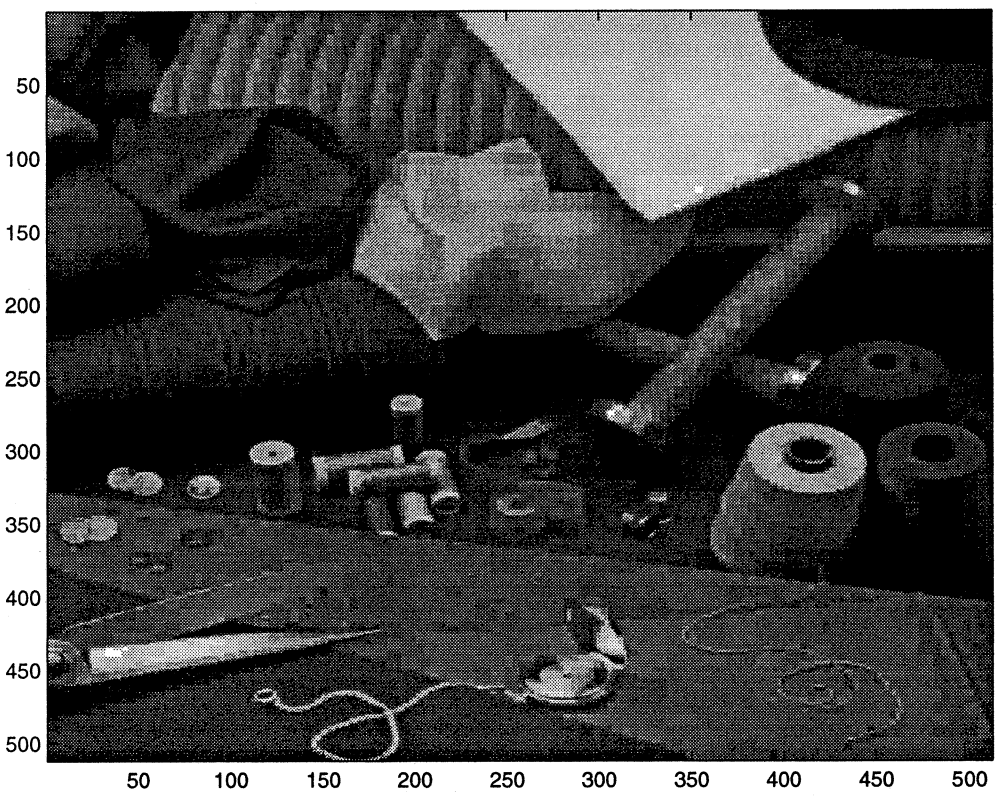
4 Applications

We will show applications of the algorithm presented above to enhancement of images that have been thresholded or quantized, magnification of images, and last but not least, deblocking of images compressed with JPG software.

Dethresholding and dequantization.

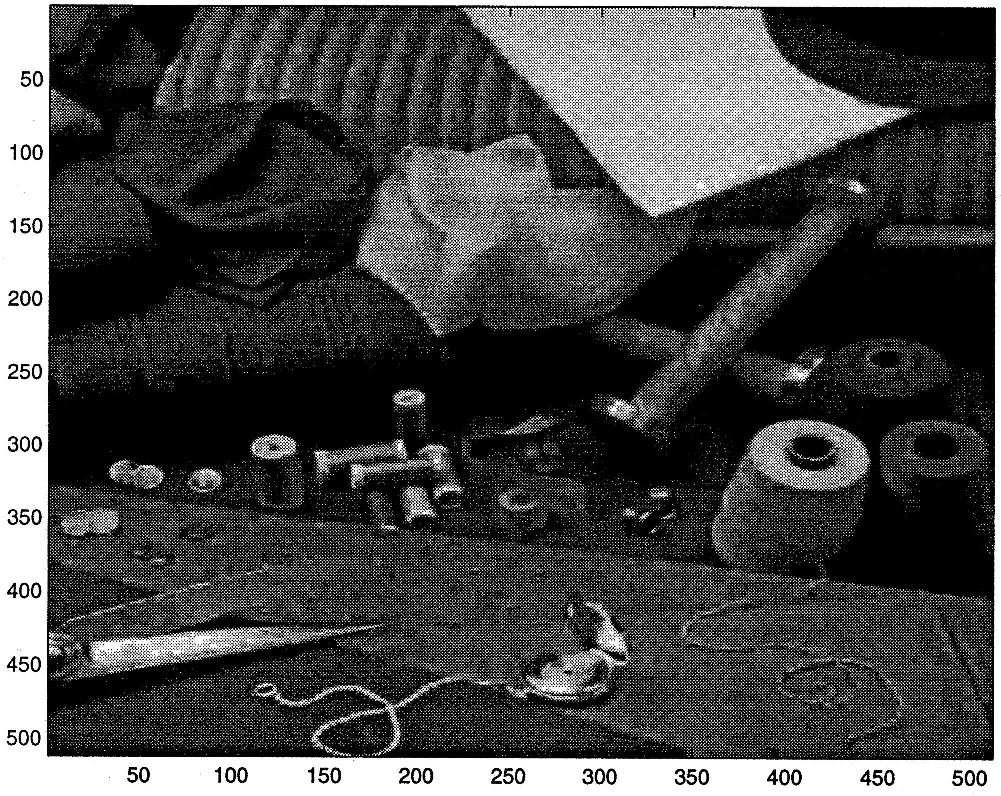
Below we present the result of application of the algorithm presented above to a picture compressed using a few low-regularity wavelet bases. Experiment shows that the quality of restoration does not depend on whether the artifacts have been introduced by thresholding or quantization. The method does not depend on whether we use wavelet bases or wavelet packet bases, either. Similarily as in the case of one-dimensional signals, the improvement in quality of the image becomes less spectacular as the wavelets used for compression become more regular and blocking artifacts are replaced by ringing phenomena. This suggests an application of this method in processes that require high speed of computation (especially on the transmitter side) with reasonable image quality.

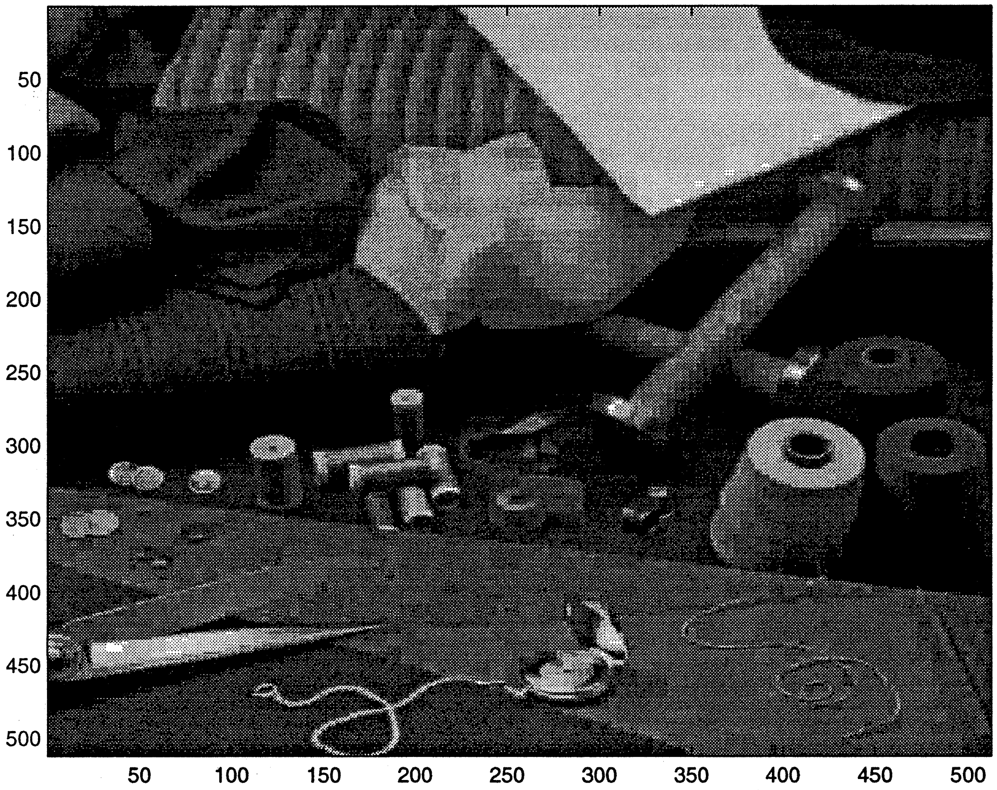

The original image
50
100
150
200
250
300
350
400
450
500
50
100
150
200
250
300
350
400
450
500



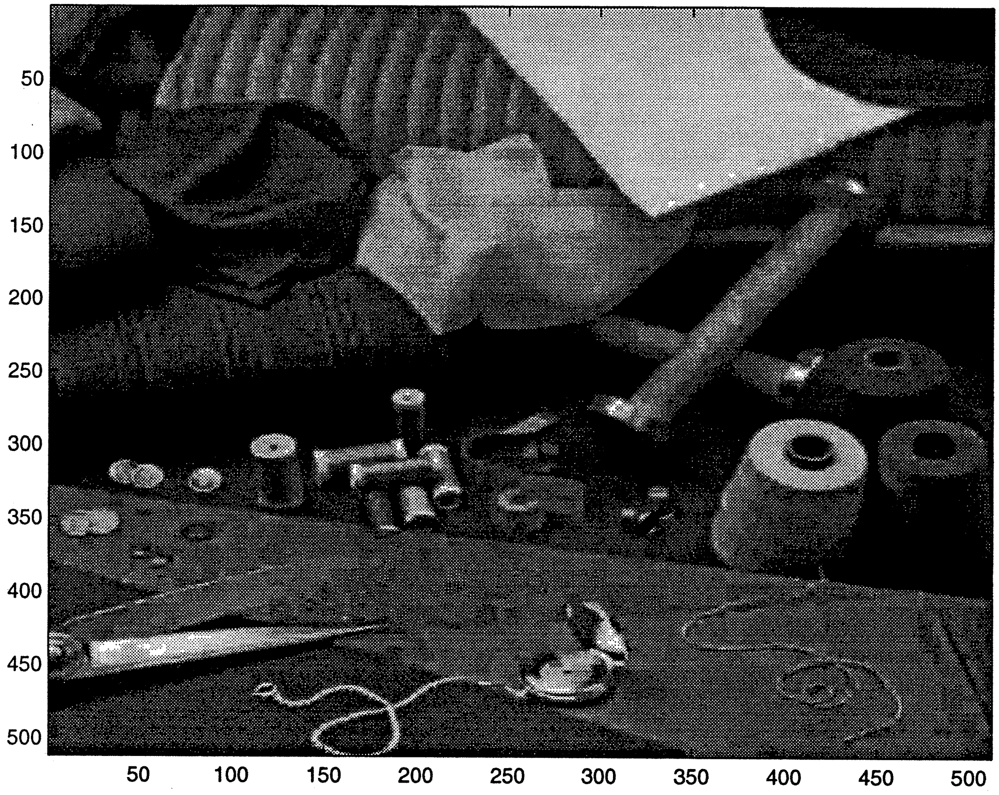
The image compressed in the Haar wavelet basis

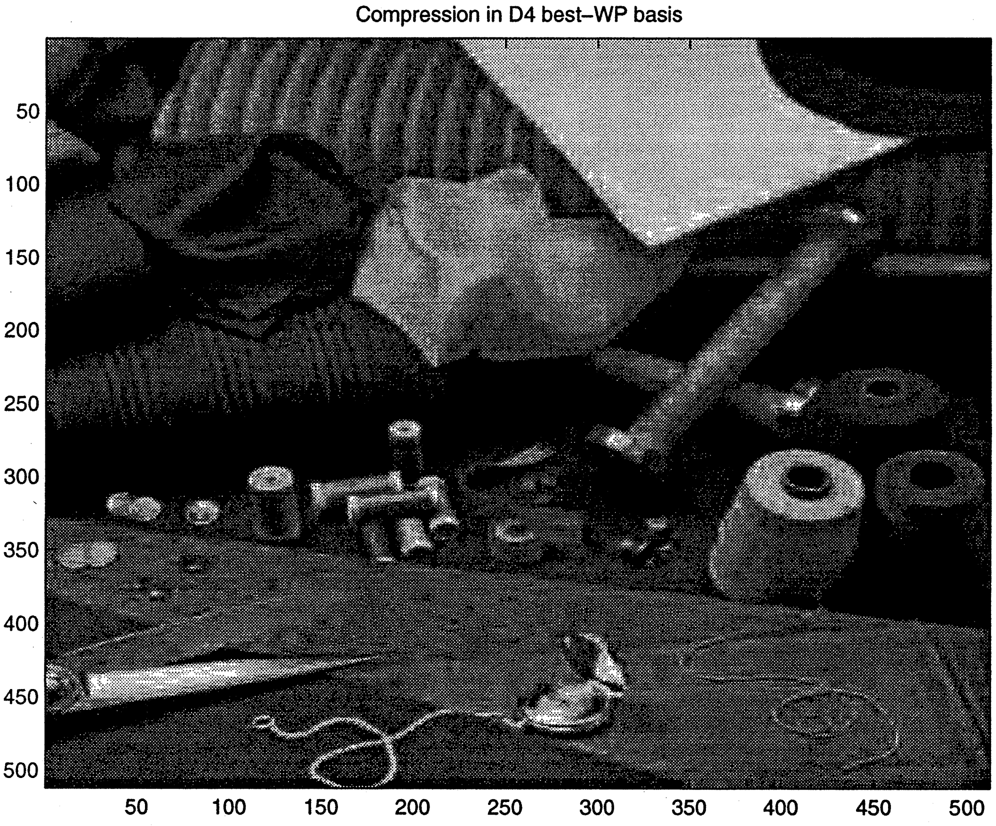
The image restored with the algorithm presented above

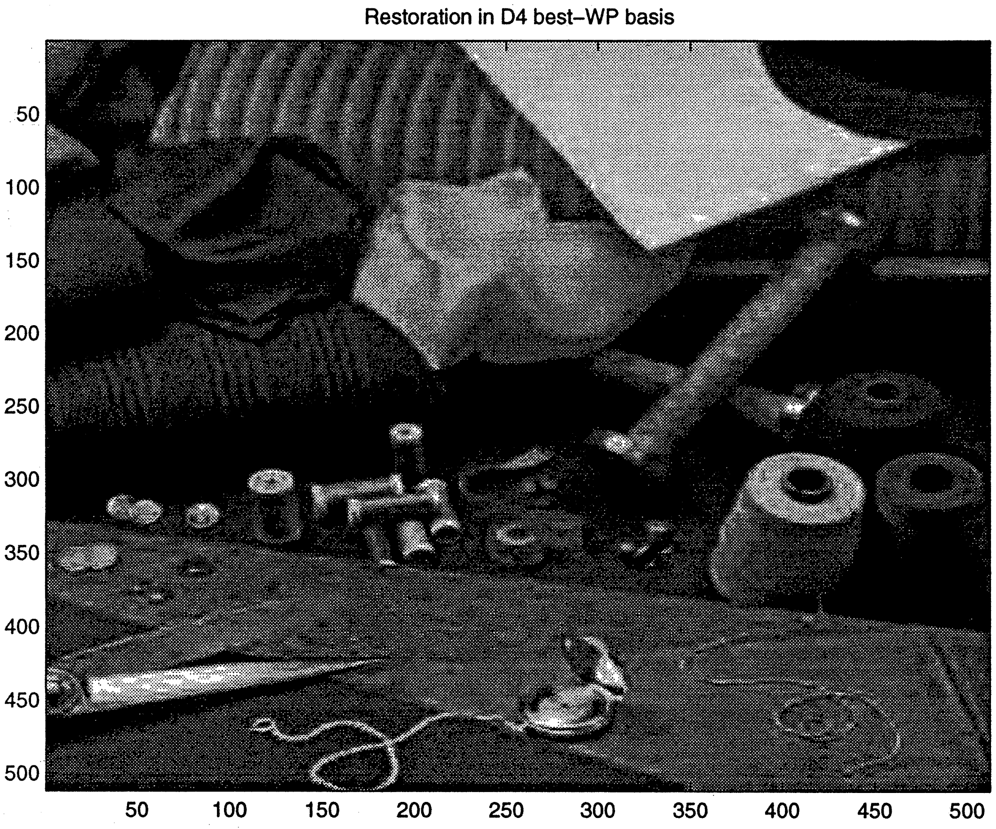


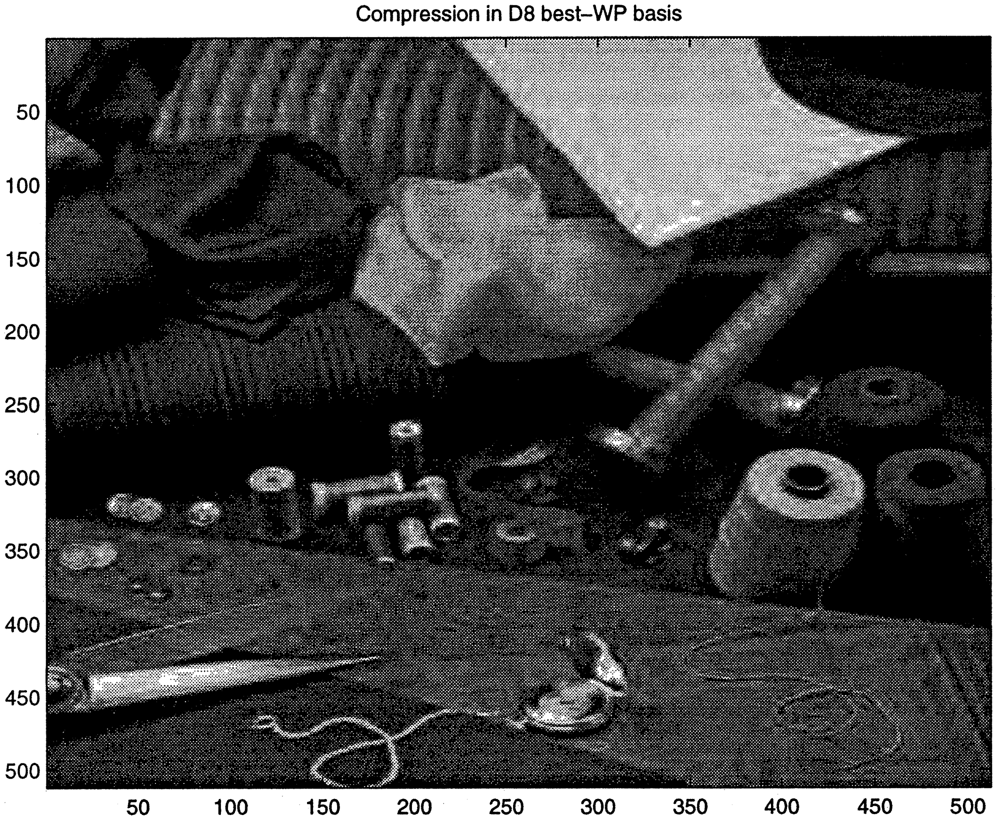

The image compressed in the Haar best-wavelet-packet basis

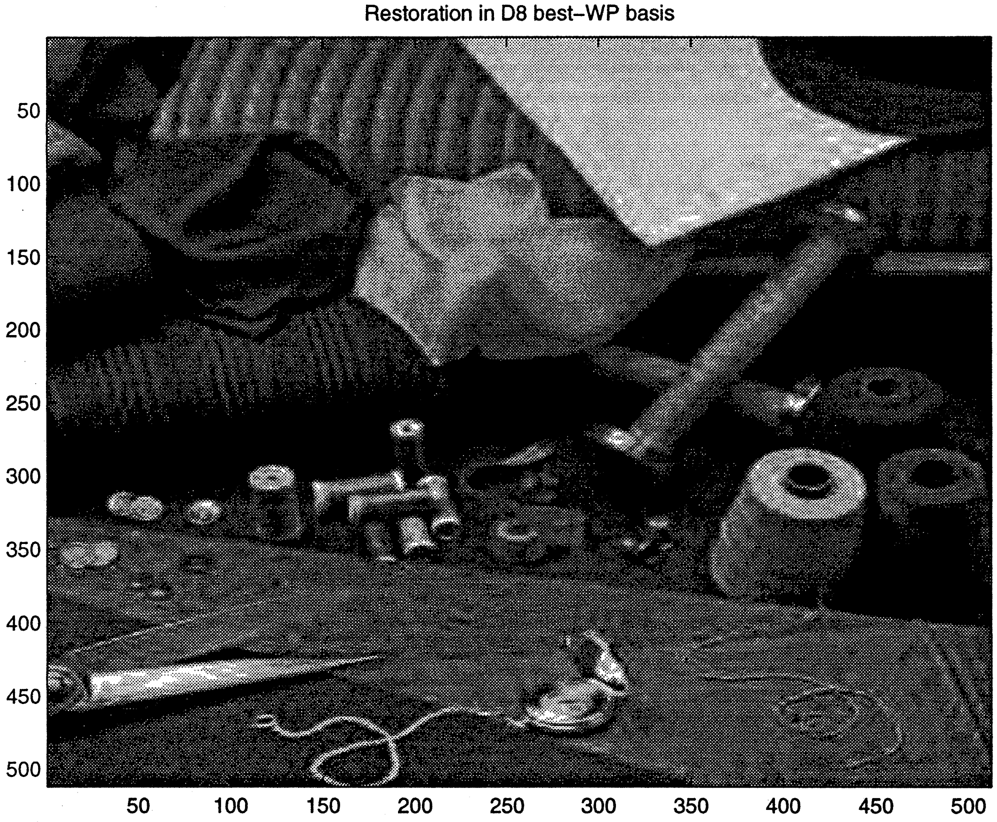
The image restored with the wavelet–packet version of the algorithm

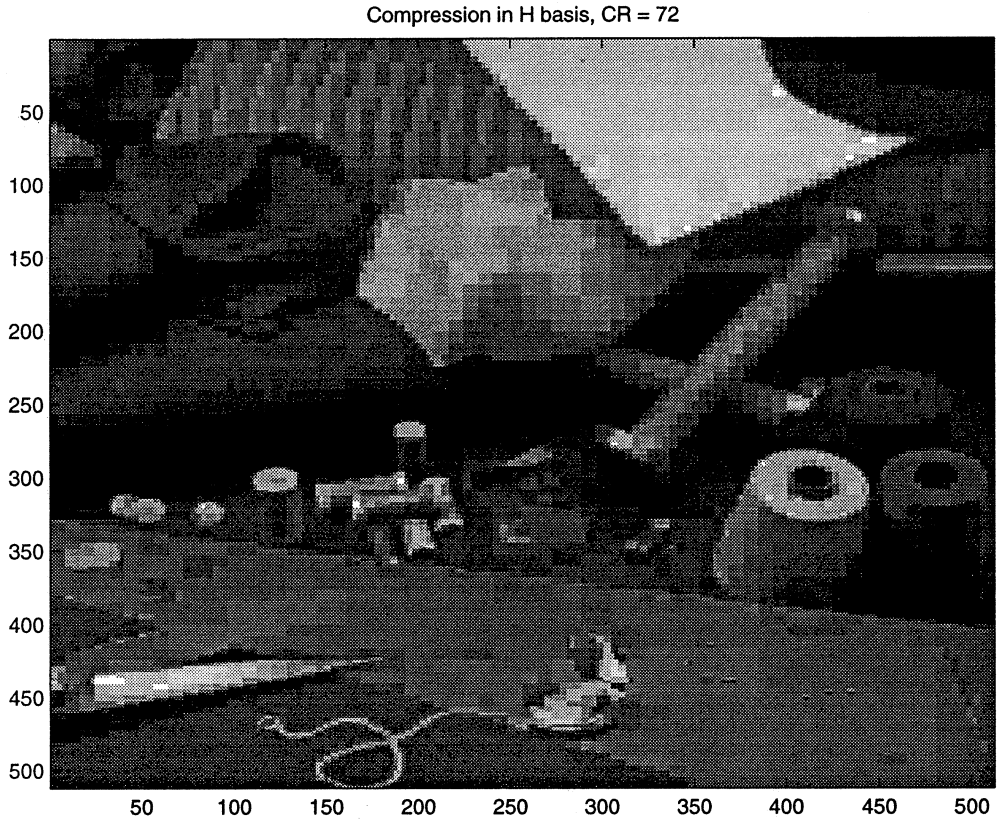



Compression in D4 best–WP basis
50
100
150
200
250
300
350
400
450
500
50
100
150
200
250
300
350
400
450
500

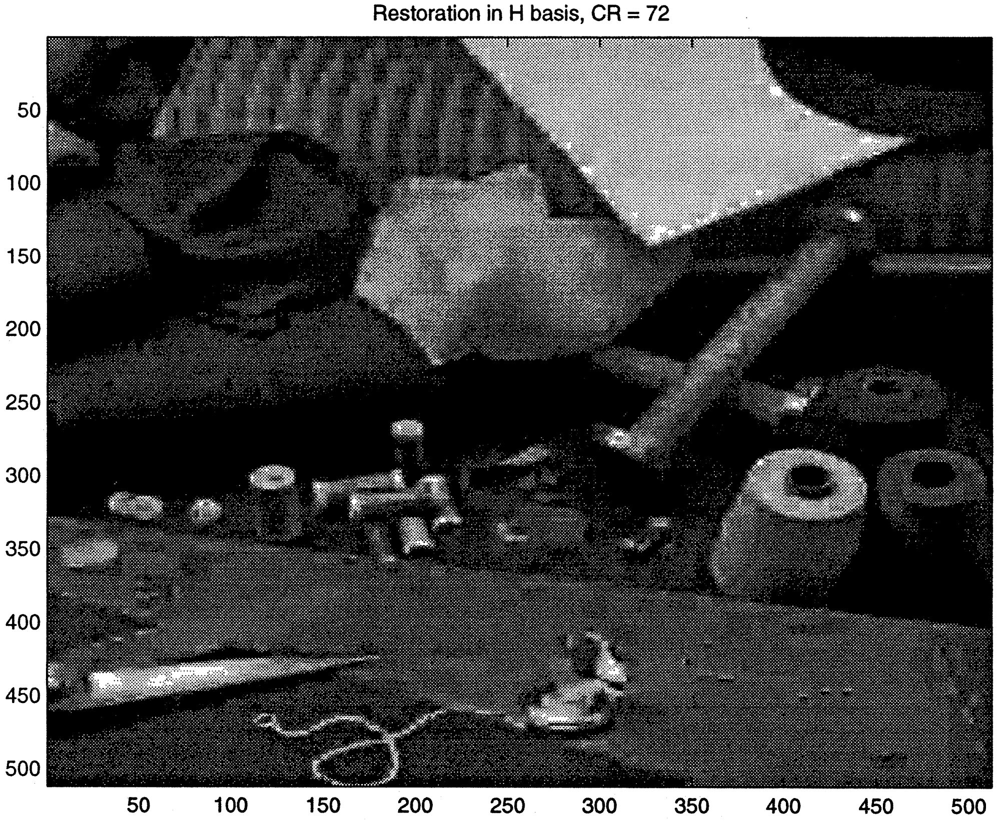

Restoration in D4 best–WP basis
50
100
150
200
250
300
350
400
450
500
50
100
150
200
250
300
350
400
450
500


Compression in D8 best–WP basis
50
100
150
200
250
300
350
400
450
500
50
100
150
200
250
300
350
400
450
500


Restoration in D8 best–WP basis
50
100
150
200
250
300
350
400
450
500
50
100
150
200
250
300
350
400
450
500

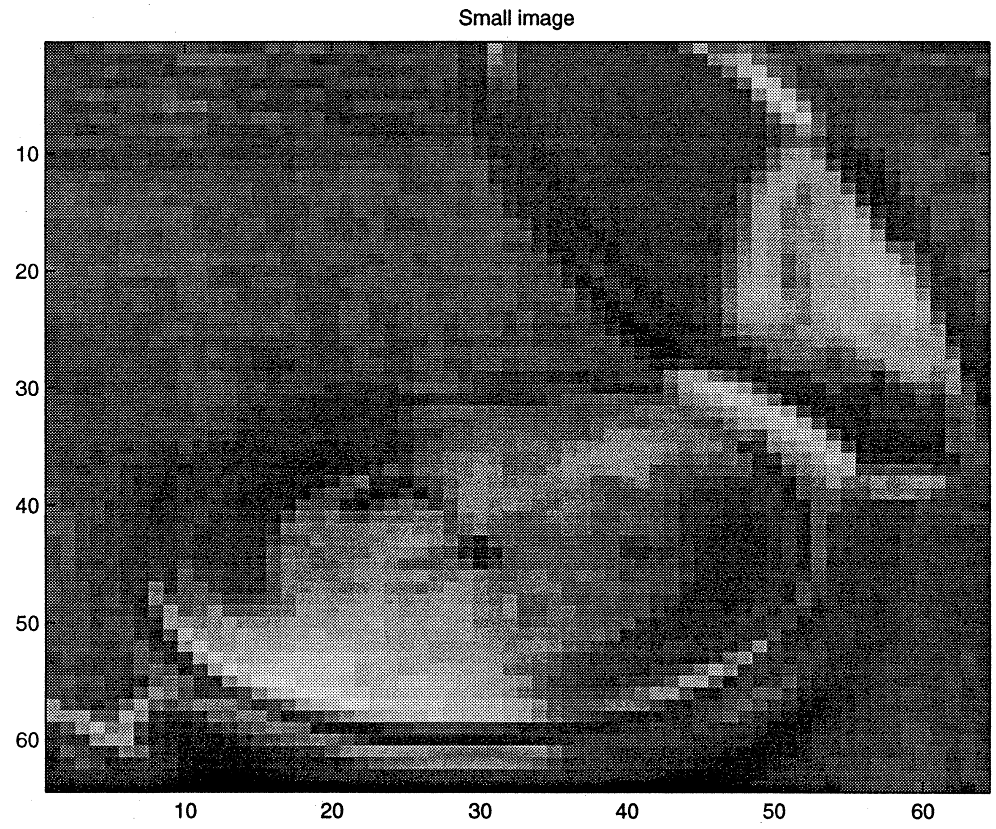


Compression in H basis, CR = 72

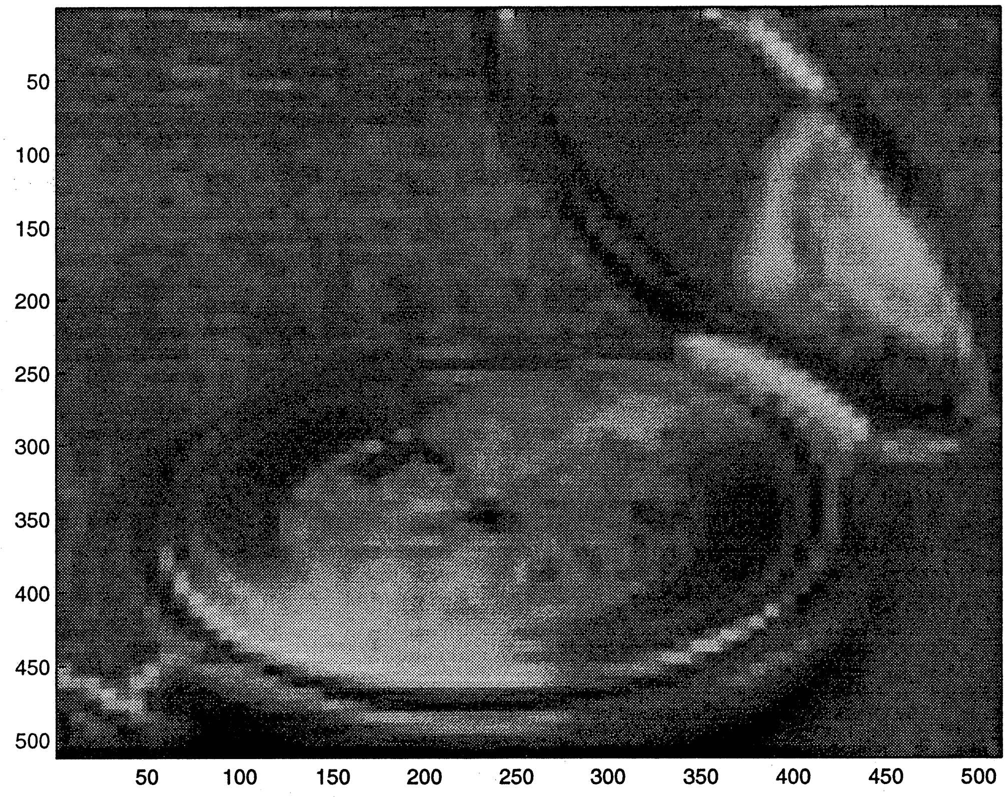

Restoration in H basis, CR = 72
50
100
150
200
250
300
350
400
450
500
50
100
150
200
250
300
350
400
450
500

Magnification

For magnification of images, one takes the image as the low-pass of the Haar wavelets. After having performed the magnification by the inverse Haar transform, one applies the algorithm described above. No thresholding is necessary. The image has to be renormalized to amplify the energy.


Small image
10
20
30
40
50
60
10
20
30
40
50
60

Bigger image



DE-JPG

By DE-JPG we understand an adaptation of our algorithm that allows us to *rapidly* deblock images compressed with JPG software. In fact the algorithm in this version:

- Performs three levels of the Haar transform (3 because the JPG blocks have size 8-by-8).
- Assumes the low-pass coefficients as accurate (they are the average values of the image inside the 8-by-8 blocks).
- Extrapolates new high-pass coefficients using the principle explained in the previous section.
- Automatically estimates the appropriate threshold level for a given image. (This is done using an empirical formula given in terms of the level of quantization of the low-pass coefficients of the JPG compressed image.)
- The newly extrapolated coefficients are thresholded using the above estimate and are then added to the existing high-pass coefficients (but only where the latter ones are originally below the threshold).
- The new image pixel values are reset between 1 and 255.

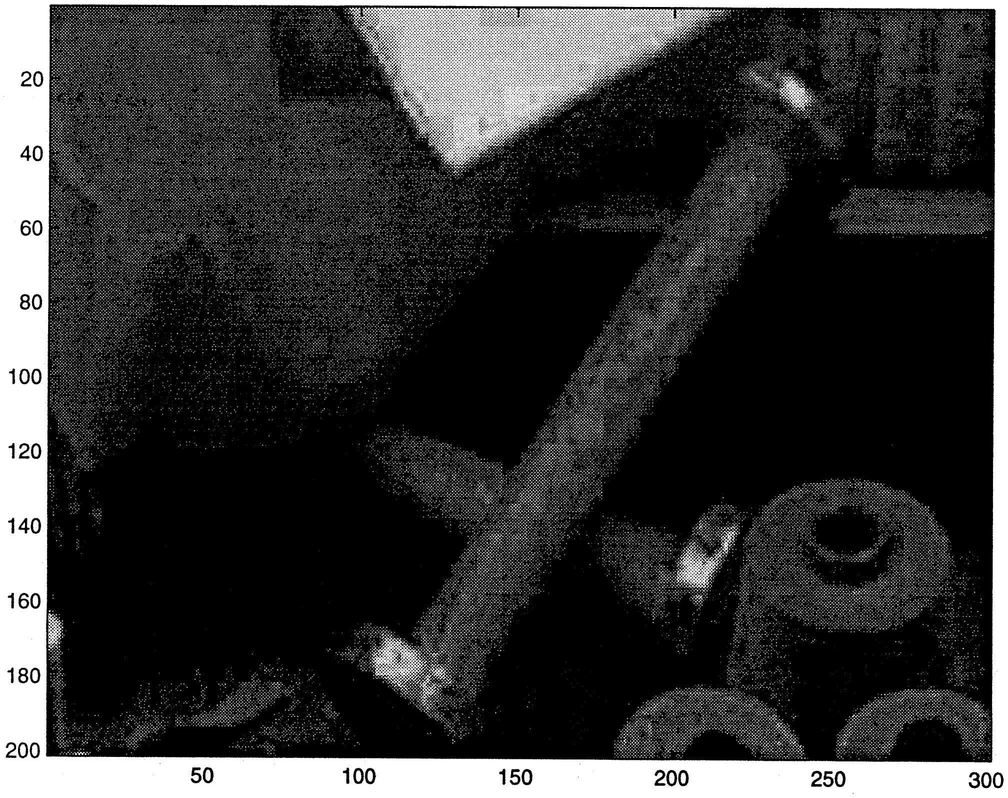
The automatic detection of the threshold can be optionally replaced by manual selection of the threshold. Again, as seen in the theorem above, the bigger the threshold the more smoothing should be expected.

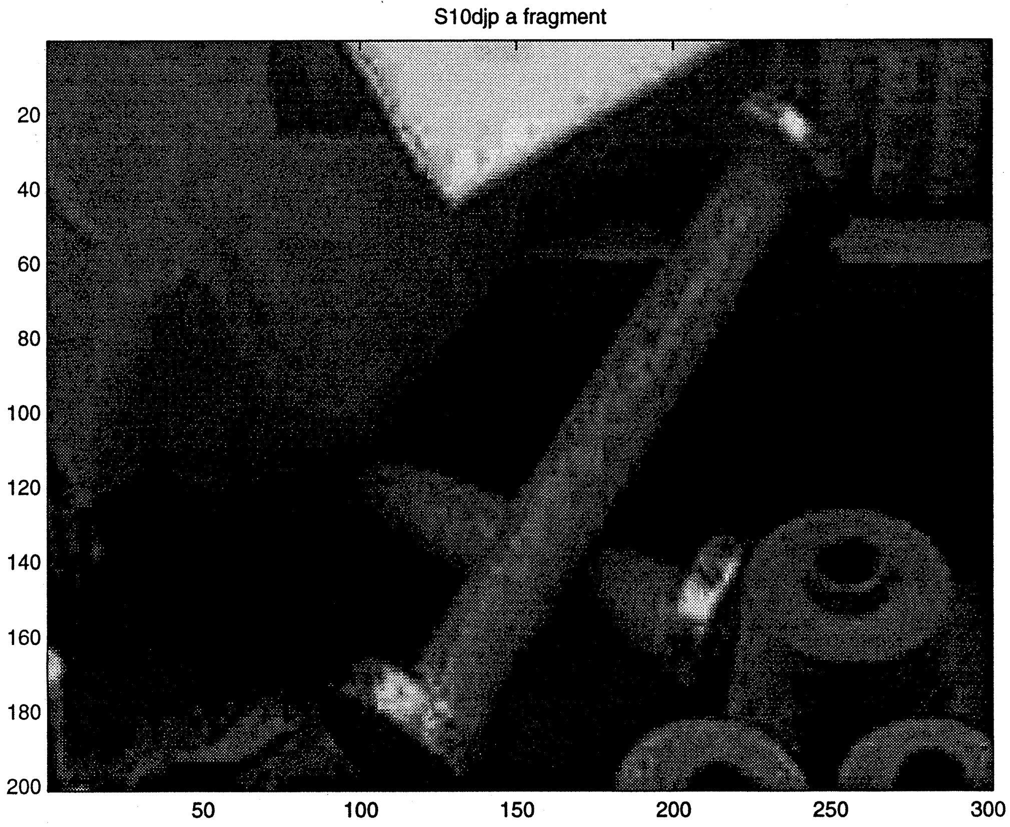
The automatic threshold is set so that it increases PSNR of the image in the known cases. Striking as it may seem at first, this claim can be substantiated theoretically. The JPG-compressed images that have been post-processed with DE-JPG can be sharpened, e.g.using the standard sharpening tool in 'xv' software, without emphasizing the 8-by-8 grid. This is in contrast to JPG compressed images, even for very low compression rates (like, say, 4 for JPG quality approximately 90.)

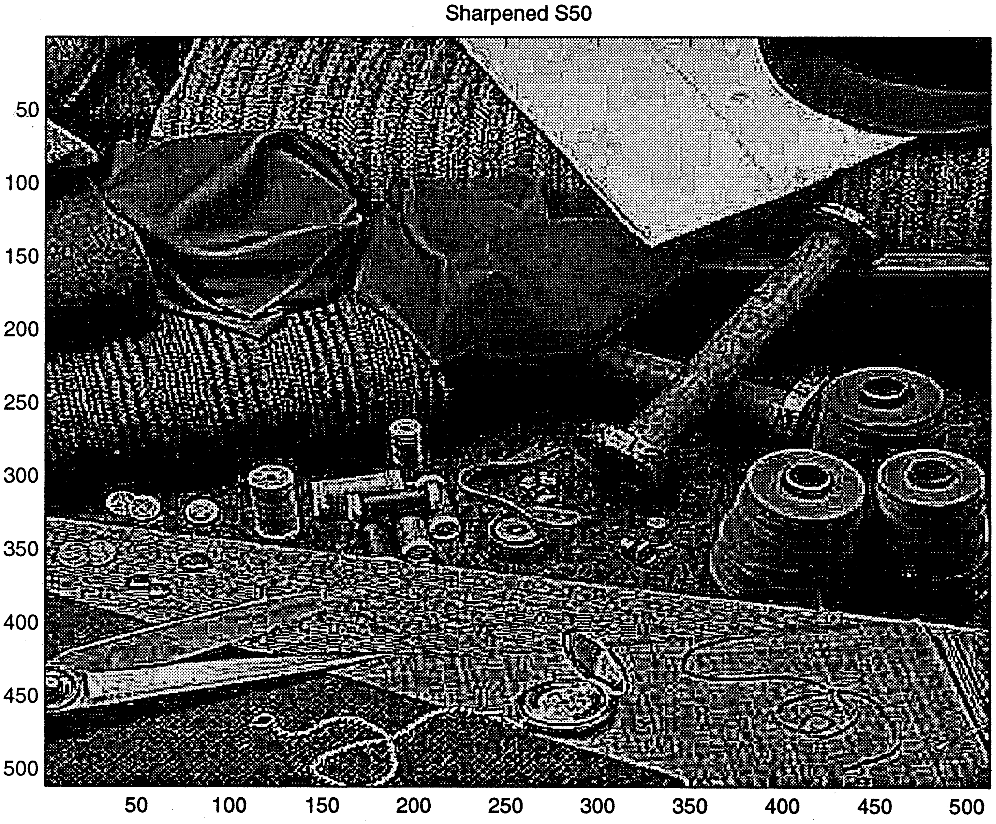
Below we present the results obtained in an application of the rapid deblocking algorithm DE-JPG. 'Sx' (the Pegasus-JPG-compressed image with luminance quality equal to 'x') is the input, while 'Sxdjp' is the output of the algorithm. This is the completely automatic version of the procedure with default choice of the threshold (given as a function of the image). The user can optimize the output further (either for the PSNR or other measures), by adjusting the threshold parameter manually. Large threshold gives smoothing effects.

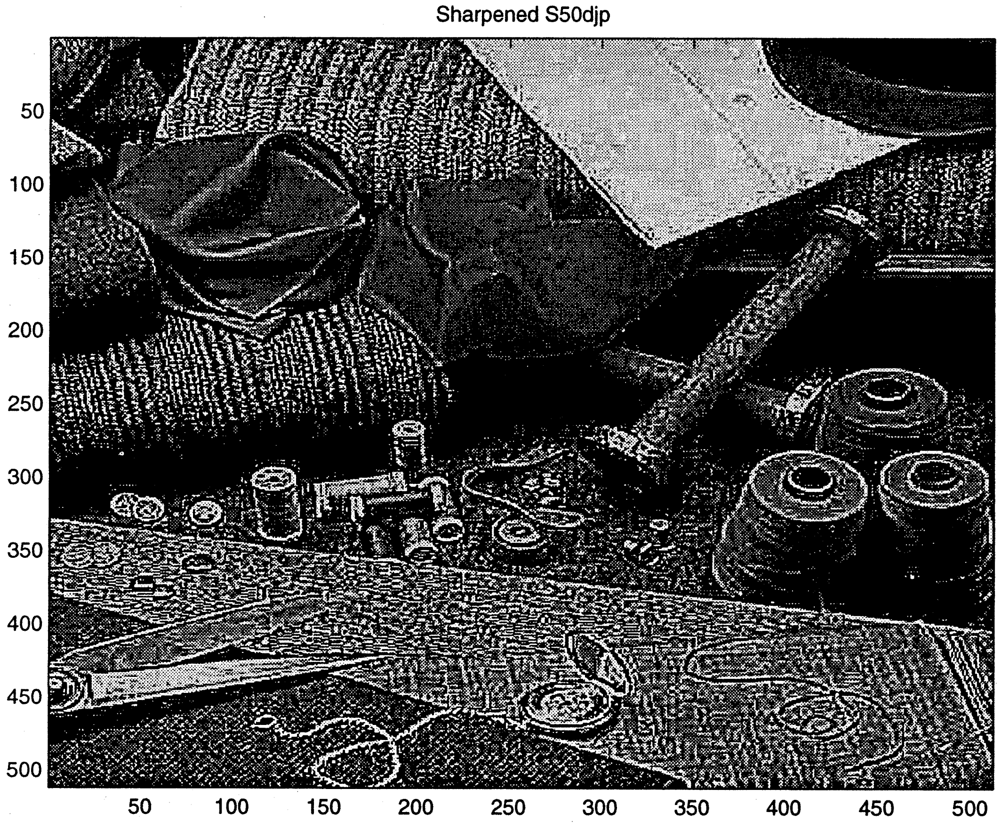
Original and De-blocked	Compression Ratio	Power Signal to Noise
S1	40.1	28.7431
S1djp		28.8353
S3	35.1	29.2821
S3djp		29.4355
S5	32.2	29.6247
S5djp		29.7802
S10	27.6	30.1728
S10jp		30.3099
S15	24.3	30.6373
S15djp		30.7585
S20	21.7	31.0266
S20djp		31.1269
S30	17.9	31.7030
S30djp		31.7795
S40	14.8	32.3777
S40djp		32.4434
S50	12.2	33.0968
S50djp		33.1331
S70	8.0	35.1843
S70djp		35.1997
S80	6.0	36.9916
S80djp		37.0042
S90	4.0	40.2701
S90djp		40.2825

S10 a fragment




S10djp a fragment
20
40
60
80
100
120
140
160
180
200
50
100
150
200
250
300


Sharpened S50
50
100
150
200
250
300
350
400
450
500
50
100
150
200
250
300
350
400
450
500


Sharpened S50djp
50
100
150
200
250
300
350
400
450
500
50
100
150
200
250
300
350
400
450
500

Acknowledgments

The authors would like to thank Christopher Hatchell for editing the manuscript.

References

[1] L. Alvarez, P.-L. Lions, J.-M. Morel, Image Selective Smoothing and edge detection by nonlinear diffusion, II, SIAM J. NUMER ANAL., Vol. 29, No. 3, pp. 845–866, June 1992

[2] Y. Bobichon and A. Bijaoui, Regularized multiresolution methods for astronomical image enhancement.

[3] Philippe G. Ciarlet, Introduction to numerical linear algebra and optimization, Cambridge University Press, Cambridge, New York 1989

[4] S. Lawrence Marple Jr., Digital Spectral Analysis, Prentice-Hall, Englewood Cliffs, 1987

[5] Jean-Michel Morel, Sergio Solimini, Variational Methods in Image Segmentation, Birkhäuser, Boston, Basel, Berlin 1995

[6] S. Osher, L. I. Rudin, Enhancement using shock filters, SIAM J. NUMER ANAL., Vol. 27, No. 4, pp. 919–941, August 1990

[7] P. Perona and J. Malik, Scale space and edge detection using anisotropic diffusion, Proc. IEEE Comput. Soc. Workshop on Comput. Vision, 1987