

COMPRESSION ARTIFACTS REDUCTION USING VARIATIONAL METHODS : ALGORITHMS AND EXPERIMENTAL STUDY

Pierre Weiss¹, Laure Blanc-Féraud¹

Thomas André², Marc Antonini²

¹Ariana Reseach Group - INRIA/I3S, CNRS
2004, routes des Lucioles, BP93
06902 Sophia Antipolis, Cedex France
E-mail={pierre.weiss, laure.blanc_feraud}@inria.fr

² Creative project - I3S, CNRS/UNSA
2000 route des Lucioles, BP 121
06903 Sophia Antipolis Cedex France
E-mail={andret, am}@i3s.unice.fr

ABSTRACT

Many compression algorithms consist of quantizing the coefficients of an image in a linear basis. This introduces compression noise that often look like ringing. Recently some authors proposed variational methods to reduce those artifacts. They consists of minimizing a regularizing functional in the set of antecedents of the compressed image. In this paper we propose a fast algorithm to solve that problem. Our experiments lead us to the conclusion that these algorithms effectively reduce oscillations but also reduce contrasts locally. To handle that problem, we propose a fast contrast enhancement procedure. Experiments on a large dataset suggest that this procedure effectively improves the image quality at low bitrates.

Index Terms— Image coding, Image restoration, Variational methods, Convex optimization

1. INTRODUCTION

State-of-the-art image compression algorithms share the same general principle : transform the image in a linear space where its representation is sparse, quantize the coefficients in this space, perform a lossless compression on the quantized coefficients. The first basis used were the *local cosine transform* in JPEG. The new compression standard uses a *wavelet* transform in JPEG2000. More recently *wedgelets*, *bandlets* and *contourlets* were proposed and seem to show better compression performances at low bitrates.

At high bitrate, there is almost no perceptible difference between the original image and the compressed image. At low bitrate, some details of the image disappear, and noticeable compression artifacts appear. Most bases for coding are constructed using smooth oscillatory functions, and thus induce artifacts similar to gibbs effect (oscillations localized near edges).

It seems difficult to recover lost details, but the compression artifacts can be reduced. Some authors proposed to minimize regularizing functionals - like the total variation - in the set of antecedents of the compressed image [1, 2, 3]. In the following, we call this procedure *decompression*. The first contribution of this paper is to propose a *new convergent and fast algorithm* to solve this problem.

Our experiments led us to the conclusion that this decompression removes the oscillations, but also strongly reduces the contrast of small details, annihilating the interest of the method. We thus introduce a *fast contrast enhancement procedure*. This is the second contribution of this paper.

The outline of the paper is as follows: first we formalize the problem of image decompression. Second, we detail the proposed algorithm, give its convergence rate and show that it outperforms classical schemes. Third, we introduce the contrast enhancement procedure. Finally, we give some results for grayscale images.

2. THE DECOMPRESSION MODEL

Let $f_{ex} \in \mathbb{R}^n$ be an image composed of n pixels.
Let $A : \mathbb{R}^n \rightarrow \mathbb{R}^n$ be some linear invertible transform.
Let $q : \mathbb{R} \rightarrow \{a^1, a^2, \dots, a^n\}$ be a quantizer defined by

$$q(x) = a^j \text{ if } x \in [a^j - \frac{a^j - a^{j-1}}{2}, a^j + \frac{a^{j+1} - a^j}{2}] \quad (1)$$

Let $Q = (q_1, q_2, \dots, q_n)$ be a set of n different quantizers. Using these notations, a compressed image f can be expressed as

$$f = A^{-1}(Q(Af_{ex})) \quad (2)$$

Suppose now that we have a compressed image f and that the quantizers q_i are known or can be estimated. This hypothesis is not too restrictive as in many compression schemes, the quantizers are defined only by one parameter: the quantization step.

Under these conditions, we can determine the set of antecedents of f , denoted K .

$$K = \{u \in \mathbb{R}^n, Q(Au) = Af\} \quad (3)$$

$$= \{u \in \mathbb{R}^n, \alpha_i \leq (Au)_i - (Af)_i < \beta_i \quad \forall i\} \quad (4)$$

where α_i and β_i can be easily determined given the expressions of $(Af)_i$ and q_i . The closure \bar{K} of this set is

$$\bar{K} = \{u \in \mathbb{R}^n, \alpha_i \leq (Au)_i - (Af)_i \leq \beta_i \quad \forall i\} \quad (5)$$

In order to remove the compression artifacts, we can look for the image in the set \bar{K} which minimizes some regularizing criterion J [1, 2, 3]. Examples of such criterion include total variation ($J(u) = \sum_{i=1}^n |(\nabla u)_i|$) or any other convex, edge preserving energy of the form

$$J(u) = \sum_{i=1}^n \phi(|(\nabla u)_i|) \quad (6)$$

where ϕ is convex, grows linearly at infinity and is preferably differentiable at 0 (for numerical purposes). Finally, the problem of decompression by variational methods consists of determining

$$\inf_{u \in \tilde{K}} (J(u)) \quad (7)$$

where J is a convex criterion, and \tilde{K} is convex, closed and bounded. In the next section, we propose an algorithm to solve this problem efficiently.

3. A FAST DECOMPRESSION ALGORITHM

Recently, Y. Nesterov [4] introduced an algorithm that allows to minimize any convex, Lipschitz differentiable functional under convex constraints. The interesting fact about this scheme is that it is optimal. No scheme can achieve a better rate of convergence than it, only using the gradient of the functional. We detail it in this section.

Let us introduce some notations :

$\langle \cdot, \cdot \rangle$ is the canonical scalar product of \mathbb{R}^n .

For $x \in \mathbb{R}^n$, $|x|_2 := \sqrt{\langle x, x \rangle}$.

Let A be a linear operator, $\|A\|_2 := \max_{\{x, |x|_2 \leq 1\}} |Ax|_2$.

A^* is the complex transpose of A . Note that $\|A^*\|_2 = \|A\|_2$.

$\nabla : \mathbb{R}^n \rightarrow \mathbb{R}^{2n}$ is any constant linear approximation of the gradient (see for instance [5]).

$-div : \mathbb{R}^{2n} \rightarrow \mathbb{R}^n$ is the dual (transpose) of ∇ , uniquely defined by the relation

$$\langle \nabla u, g \rangle = - \langle u, div g \rangle \quad \forall u \in \mathbb{R}^n \text{ and } \forall g \in \mathbb{R}^{2n} \quad (8)$$

Finally, let us denote $J'(u)$ the gradient of J at point u . In the case of the functional (6) - for differentiable ϕ - classical calculus of variations gives

$$J'(u) = -div(\phi'(|\nabla u|) \frac{\nabla u}{|\nabla u|}) \quad (9)$$

Let us turn to the optimization problem (7)

$$\inf_{u \in \tilde{K} = \{u \in \mathbb{R}^n, \alpha_i \leq (Au)_i - (Af)_i \leq \beta_i\}} (J(u)) \quad (10)$$

As the projection on the set \tilde{K} might be cumbersome to compute for non orthonormal transforms A , we use the change of variable $y = Au$. This leads to the problem

$$\inf_{y \in \mathbb{R}^n, \alpha_i \leq y_i - (Af)_i \leq \beta_i} (J(A^{-1}y)) \quad (11)$$

The interest of that formulation is that the set

$$\tilde{K} = \{y \in \mathbb{R}^n, \alpha_i \leq y_i - (Af)_i \leq \beta_i\} \quad (12)$$

simply is an hyperrectangle. The Euclidean projection on that polytope writes in closed form

$$(\Pi_{\tilde{K}}(x))_i = \begin{cases} (Af)_i - \alpha_i & \text{if } x_i < (Af)_i - \alpha_i \\ x_i & \text{if } \alpha_i \leq x_i - (Af)_i \leq \beta_i \\ (Af)_i + \beta_i & \text{if } x_i > (Af)_i + \beta_i \end{cases} \quad (13)$$

This allows the use of projected gradient descent like algorithms to solve problem (11). In [6], the authors applied a Nesterov algorithm to imaging problems and showed that for both differentiable and non differentiable functionals J , this method outperforms classical first order schemes like gradient or subgradient descents.

After some calculations, we show that if ϕ satisfies $|\phi''|_\infty < \frac{1}{\mu}$ for some $\mu > 0$, applying Nesterov ideas to problem (11) leads to the following algorithm

| |
|---|
| 0 - Set $k = -1$, $G^0 = 0$, $x^0 \in \tilde{K}$, $L = \frac{1}{\mu} \ A^{-1}\ _2^2 \ div\ _2^2$, |
| 1 - Set $k = k + 1$, |
| 2 - Compute $\eta^k = -A^{-*} div(\phi'(\nabla A^{-1}x^k) \frac{\nabla A^{-1}x^k}{ \nabla A^{-1}x^k })$, |
| 3 - Set $y^k = \Pi_{\tilde{K}}(x^k - \frac{\eta^k}{L})$ |
| 4 - Set $G^k = G^{k-1} + \frac{k+1}{2} \eta^k$ |
| 5 - Set $z^k = \Pi_{\tilde{K}}(-\frac{G^k}{L})$ |
| 6 - Set $x^{k+1} = \frac{2}{k+3} z^k + \frac{k+1}{k+2} y^k$, go back to 1 until $k = N$. |

Let $D = \sum_{i=1}^n \Delta_i^2$ (this is the square of the radius of \tilde{K}). Let \bar{y} be a solution of (11). At iteration N this scheme ensures that

$$0 \leq J(y^N) - J(y^*) \leq \frac{2LD}{(N+1)(N+2)} \quad (14)$$

In view of (14), getting a solution of precision ϵ with this scheme requires no more than $O(\frac{1}{\sqrt{\epsilon}})$ iterations. The projected gradient descent can be shown to be an $O(\frac{1}{\epsilon})$ algorithm [7]. We thus gain one order in the convergence rate. Roughly speaking, the variable G at step 4 aggregates the information brought by the gradients computed at the previous iterations. This avoids exploring useless directions and explains the efficiency of this scheme.

A surprising remark (see [6] for further details) is that even in the case of total variation which is non differentiable, this scheme applied to a smooth approximation of the l^1 -norm remains more efficient than classical first order algorithms. For total variation, this scheme can be shown to require $O(\frac{1}{\epsilon})$ iterations to get a solution of precision ϵ , while to our knowledge, all first order schemes in the literature require at least $O(\frac{1}{\epsilon^2})$ iterations. Both for differentiable and non differentiable regularizing criterions, this scheme is very efficient.

In all coming experiments, we use the regularization function ϕ defined as

$$\phi(x) = \begin{cases} |x| & \text{if } |x| \geq \mu \\ \frac{x^2}{2\mu} + \frac{\mu}{2} & \text{otherwise} \end{cases} \quad (15)$$

which is edge preserving and can be shown to be a good approximation of the total variation. It satisfies

$$|\phi''|_\infty \leq \frac{1}{\mu} \quad (16)$$

The parameter μ controls the degree of smoothing in flat regions. For a discussion on the choice of this parameter, we refer the reader to section 5.

4. AN ALGORITHM FOR CONTRAST ENHANCEMENT

Looking at Figure 3, we see that image (b) has a good contrast but oscillates, while image (d) is fewly contrasted with no spurious oscillations. It would be enjoyable to create an image that shares the best of both images.

We propose the following approach: find the image v closest (in the Euclidean metric) to f (the compressed image) which has the same level lines as u (the restored image). For this procedure to be efficient, it is necessary that the compression algorithm preserves locally the mean of the original image. This hypothesis holds in the case of wavelet compression. Low frequency bands are fewly quantized which ensures contrast preservation.

The algorithm we use on a discrete image is described as follows:

1. Set $u_Q = \lfloor \frac{u}{\Delta} \rfloor \Delta$ (uniform quantization). In the experiments, we use $\Delta = \frac{\max(u) - \min(u)}{N}$ with $N = 256$.
2. For each level $i\Delta$ ($i \in \mathbb{Z}$):

(a) Separate the connected components $\Omega_{i,j}$ of the set :

$$\Omega_i = \{x \in \mathbb{R}^n, u_Q(x) = i\Delta\}$$

In the experiments, we use the 8-neighbourhood to define the notion of connected component.

(b) On each component $\Omega_{i,j}$, set $v|_{\Omega_{i,j}} = \text{mean}(f|_{\Omega_{i,j}})$.

This kind of algorithm has already been used with a different motivation in [8]. The authors prove that in the continuous setting, this algorithm converges as Δ goes to 0 to the projection of f on the set of functions that have the same topographic map as u . We let the reader refer to [8] for more details.

This is a fast algorithm. Our C implementation takes less than 0.1 second a $2GHz$ computer.

5. RESULTS

In all the following experiments, we use a simple image coder which consists of a 5-levels $2D$ wavelet transform using the 9/7 filters [9], a uniform quantizer which optimizes the mean square error of the reconstructed image, and an arithmetic coder. This coder has about the same efficiency as JPEG2000.

5.1. Number of iterations and smoothing parameter μ

The error estimate (14) brings a good information on how to chose the number of iterations for a Nesterov scheme to converge. This number should grow proportionnaly to D and to $\frac{1}{\mu}$. On Fig. 1, we compare the efficiency of this scheme with a classical projected gradient descent with optimal step. Computational effort is greatly reduced.

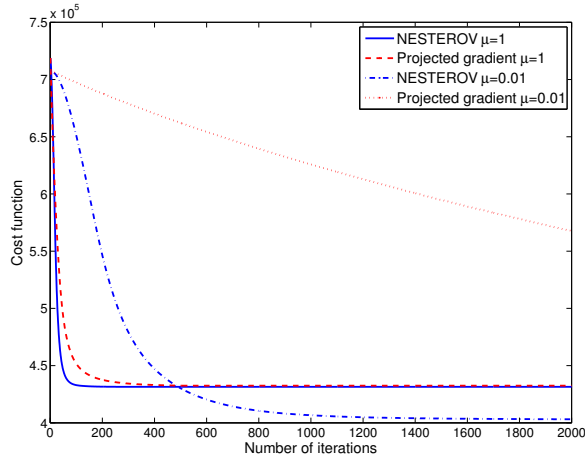


Fig. 1. Evolution of the cost function w.r.t. the number of iterations for a Nesterov algorithm VS a Projected Gradient Descent (PGD) with optimal constant step. For $\mu = 1$, Nesterov's scheme stabilizes after 80 iterations while PGD requires 400 iterations.

On figure (2), we analyse the effect of the regularization parameter μ . The curve shows the mean PSNR using 3 different images. It indicates that the best results are obtained for $\mu = 1.5$. A perceptual analysis of the results gives the same conclusion.

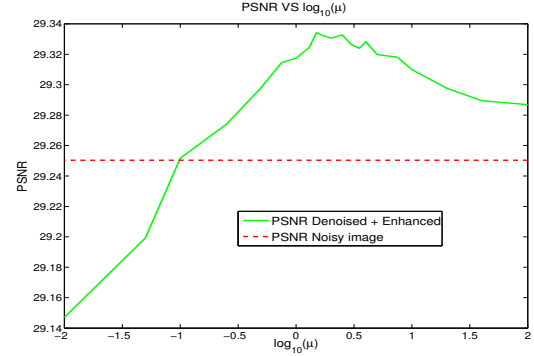


Fig. 2. PSNR w.r.t μ at a fixed compression rate (0.13 bpp)

For $\mu = 0.01$, we obtain a solution that is perceptually identical to the solution of the minimization of the total variation. The oscillations are removed, thin details are preserved. The main problems are staircase effect and contrast reduction (notably on small details). Moreover, it can be shown that the number of iterations before convergence is proportional to $\frac{1}{\mu}$. For such a parameter, we need 1000 iterations to get a satisfactory accuracy.

With $\mu = 1.5$ we get more satisfactory results. Staircase effect disappears and the Nesterov algorithm only requires 60 iterations to converge. With a greater μ , the model tends to Tikhonov regularization, oscillations are not removed, and the image is blurred. In all further experiments we take $\mu = 1.5$ which shows to be a good compromise for images of amplitude 255. However even with that parameter, the contrast of small objects is reduced.

5.2. What can be expected from our algorithm?

In document [10], we lead a series of decompression experiments with various images and various compression rates. It appears that for the tested coder, the proposed algorithm improves image quality at low bitrates, and is not useful at high bitrates. For smooth by piece images, we gain up to $1dB$ at low bitrates. For oscillating and textured images, the image quality remains the same with or without post-processing.

We assess the image quality using 3 different measures. The classical $PSNR$, the $SSIM$ measure (Structural Similarity Index) described in [11] and the NQM measure (Noise Quality Measure), described in [12]. For instance, with image Lena (see Fig. 3) the image quality is neatly improved and we get the following results

| | Decoded | Denoised | Denoised + Enhanced |
|------|---------|----------|---------------------|
| PSNR | 27.787 | 27.915 | 27.942 |
| SSIM | 0.870 | 0.870 | 0.879 |
| NQM | 22.965 | 22.185 | 23.301 |

Note that the thin details like hair are preserved, that the oscillations are removed, and that the blur is reduced.

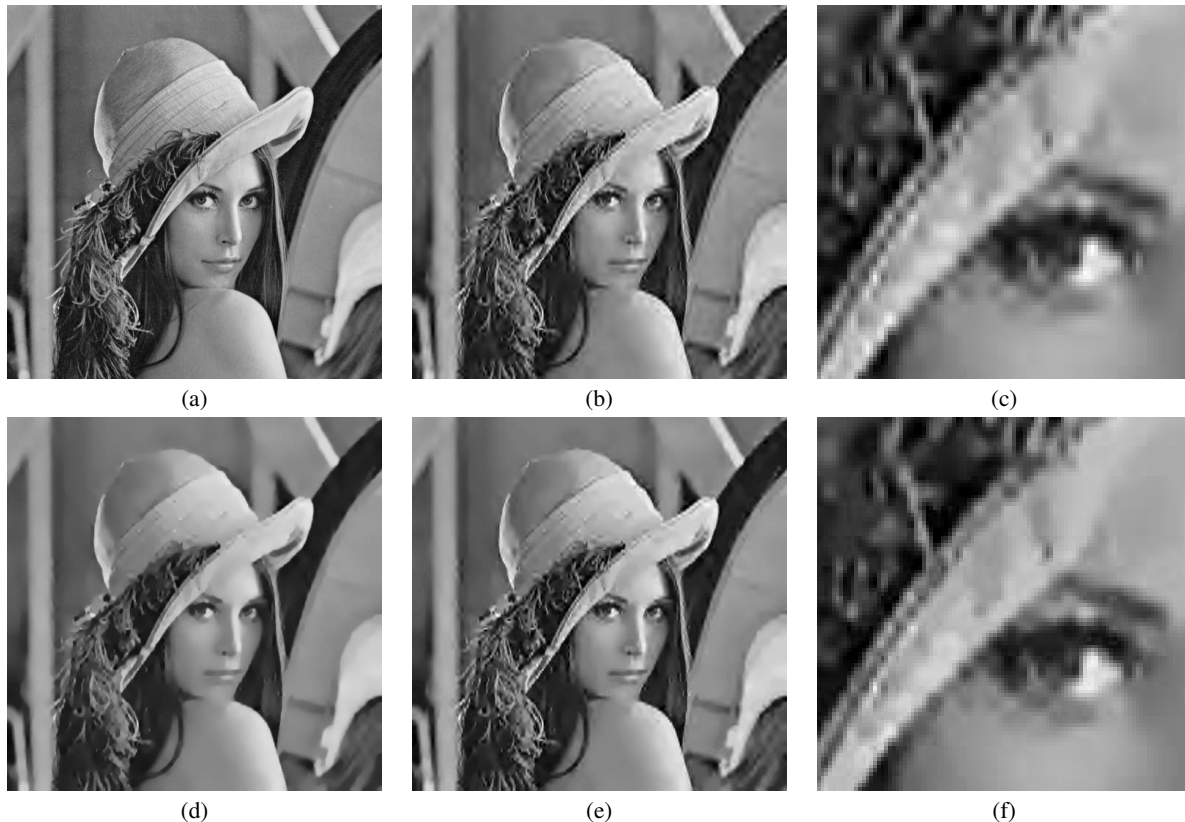


Fig. 3. a: original image - b: decoded image (0.085bpp) - c: detail of b - d: denoised using $\mu = 1.5$ - e: denoised and enhanced - f: detail of e

5.3. Conclusion

In this work we analysed the efficiency of variational methods to remove compression artifacts. From a computational aspect the algorithm we proposed is quite fast (comparable to 100 wavelet transforms), and completely automated. Numerical quality measures as well as perceptual inspection show that its efficiency strongly depends on the compression rates and on the image contents. The algorithm is very efficient at low bitrates on piecewise smooth images.

6. REFERENCES

- [1] S. Tramini, M. Antonini, M. Barlaud, and G. Aubert, "Quantization Noise Removal for Optimal Transform Decoding," *International Conference on Image Processing*, pp. 381–385, 1998.
- [2] S. Lintner and F. Malgouyres, "Solving a variational image restoration model which involves L^∞ constraints," *Inverse Problems*, vol. 20, pp. 815–831, 2004.
- [3] S. Durand and J. Froment, "Reconstruction of wavelet coefficients using total variation minimization," *SIAM Journal of Scientific Computing*, vol. 24, no. 5, pp. 1754–1767, 2003.
- [4] Y. Nesterov, "Smooth minimization of non-smooth functions," *Mathematic Programming, Ser. A*, vol. 103, pp. 127–152, 2005.
- [5] A. Chambolle, "An Algorithm for Total Variation Minimization and Applications," *JMIV*, vol. 20, pp. 89–97, 2004.
- [6] P. Weiss, L. Blanc-Féraud, and G. Aubert, "Efficient schemes for total variation minimization under constraints in image processing," Research Report 6260, INRIA, 2007.
- [7] Y. Nesterov, *Introductory Lectures on Convex Optimization: A Basic Course*, Kluwer Academic Publishers, 2004.
- [8] V. Caselles, B. Coll, and J.M. Morel, "Geometry and color in natural images," *J. Math. Imaging Vis.*, vol. 16, no. 2, pp. 89–105, 2002.
- [9] M. Antonini, M. Barlaud, P. Mathieu, and I. Daubechies, "Image Coding Using Wavelet Transform," *IEEE Transactions on Image Processing*, vol. 1, no. 2, pp. 205–220, avril 1992.
- [10] Weiss, P. and Blanc-Féraud, L. and André, T. and Antonini, M., "Some decompression results with variational methods," http://www-sop.ariana/RESULTS/Pierre_Weiss_Decompression.pdf.
- [11] Z. Wang, A.C. Bovik, H.R. Sheikh, and E.P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, avril 2004.
- [12] N. Damera-Venkata, T.D. Kite, W.S. Geisler, B.L. Evans, and A.C. Bovik, "Image quality assessment based on a degradation model," *IEEE Transactions on Image Processing*, vol. 9, no. 4, pp. 636–650, avril 2000.